COMPARATIVE ANALYSIS OF SINGLE AND MULTI-AGENT LARGE LANGUAGE MODEL ARCHITECTURES FOR DOMAIN-SPECIFIC TASKS IN WELL CONSTRUCTION

Vitor Brandão Sabbagh

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientador: Geraldo Bonorino Xexéo

Rio de Janeiro
Julho de 2025

COMPARATIVE ANALYSIS OF SINGLE AND MULTI-AGENT LARGE LANGUAGE MODEL ARCHITECTURES FOR DOMAIN-SPECIFIC TASKS IN WELL CONSTRUCTION

Vitor Brandão Sabbagh

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Orientador: Geraldo Bonorino Xexéo

Aprovada por: Prof. Nome do Primeiro Examinador Sobrenome
Prof. Nome do Segundo Examinador Sobrenome
Prof. Nome do Terceiro Examinador Sobrenome
Prof. Nome do Quarto Examinador Sobrenome
Prof. Nome do Quinto Examinador Sobrenome

RIO DE JANEIRO, RJ – BRASIL
JULHO DE 2025

Brandão Sabbagh, Vitor

Comparative Analysis of Single and Multi-Agent Large Language Model Architectures for Domain-Specific Tasks in Well Construction/Vitor Brandão Sabbagh. – Rio de Janeiro: UFRJ/COPPE, 2025.

XII, 54 p.: il.; 29, 7cm.

Orientador: Geraldo Bonorino Xexéo

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2025.

Referências Bibliográficas: p. 48 – 52.

1. Large Language Models. 2. Agentes. 3. Construção de Poços de Petroleo. I. Bonorino Xexéo, Geraldo. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Para Carolina, minha
companheira de vida.

Agradecimentos

Gostaria de agradecer a todos. [família, amigos, etc]

Estendo minha gratidão aos especialistas em engenharia de poços, Marcelo Grimberg, Rafael Peralta e Lorenzo Simonassi, cuja expertise e dedicação contribuíram significativamente para esta pesquisa.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

COMPARATIVE ANALYSIS OF SINGLE AND MULTI-AGENT LARGE LANGUAGE MODEL ARCHITECTURES FOR DOMAIN-SPECIFIC TASKS IN WELL CONSTRUCTION

Vitor Brandão Sabbagh

Julho/2025

Orientador: Geraldo Bonorino Xexéo

Programa: Engenharia de Sistemas e Computação

Apresenta-se, nesta tese, a aplicação de grandes modelos de linguagem (LLM) no setor de petróleo e gás, especificamente em tarefas de construção e manutenção de poços. O estudo avalia o desempenho de uma arquitetura baseada em LLM de agente único e de múltiplos agentes no processamento de diferentes tarefas, oferecendo uma perspectiva comparativa sobre sua precisão e as implicações de custo de sua implementação. Os resultados indicam que sistemas multiagentes oferecem desempenho melhorado em tarefas de perguntas e respostas, com uma medida de veracidade 28% maior do que os sistemas de agente único, mas a um custo financeiro mais alto. Especificamente, a arquitetura multiagente incorre em custos que são, em média, 3,7 vezes maiores do que os da configuração de agente único, devido ao aumento do número de tokens processados. Por outro lado, os sistemas de agente único se destacam em tarefas de texto para SQL (Linguagem de Consulta Estruturada), especialmente ao usar o Transformador Pré-Treinado Generativo 4 (GPT-4), alcançando uma pontuação 15% maior em comparação com as configurações multiagentes, sugerindo que arquiteturas mais simples podem, às vezes, superar a complexidade. A novidade deste trabalho reside em seu exame original dos desafios específicos apresentados pelos dados complexos, técnicos e não estruturados inerentes às operações de construção de poços, contribuindo para o planejamento estratégico da adoção de aplicações de IA generativa, fornecendo uma base para otimizar soluções contra parâmetros econômicos e tecnológicos.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

COMPARATIVE ANALYSIS OF SINGLE AND MULTI-AGENT LARGE LANGUAGE MODEL ARCHITECTURES FOR DOMAIN-SPECIFIC TASKS IN WELL CONSTRUCTION

Vitor Brandão Sabbagh

July/2025

Advisor: Geraldo Bonorino Xexéo

Department: Systems Engineering and Computer Science

This article explores the application of large language models (LLM) in the oil and gas sector, specifically within well construction and maintenance tasks. The study evaluates the performances of a single-agent and a multi-agent LLM-based architecture in processing different tasks, offering a comparative perspective on their accuracy and the cost implications of their implementation. The results indicate that multi-agent systems offer improved performance in question and answer tasks, with a truthfulness measure 28% higher than single-agent systems, but at a higher financial cost. Specifically, the multi-agent architecture incurs costs that are, on average, 3.7 times higher than those of the single-agent setup due to the increased number of tokens processed. Conversely, single-agent systems excel in text-to-SQL (Structured Query Language) tasks, particularly when using Generative Pre-Trained Transformer 4 (GPT-4), achieving a 15% higher score compared to multi-agent configurations, suggesting that simpler architectures can sometimes outpace complexity. The novelty of this work lies in its original examination of the specific challenges presented by the complex, technical, unstructured data inherent in well construction operations, contributing to strategic planning for adopting generative AI applications, providing a basis for optimizing solutions against economic and technological parameters.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

1.1 Contextualização

Na dinâmica em constante mudança da indústria de petróleo e gás (O&G), a transformação digital emergiu como um elemento chave para alcançar eficiência operacional, sustentabilidade e competitividade. Na vanguarda dessa transformação estão os Modelos de Linguagem de Grande Escala (LLMs), que têm o potencial de processar consultas não estruturadas, mapear alternativas e aconselhar os usuários sobre possíveis ações KAR e VARSHA (2023). Também observamos a vantagem do aumento do engajamento, cooperação, acessibilidade e, em última análise, lucratividade. Esses modelos redefinem paradigmas em gestão do conhecimento e recuperação de informações e impactam uma variedade de outras áreas ECKROTH e GIPSON (2023), tornando crucial a adoção dessas tecnologias para permanecer competitivo.

Um estudo conduzido por DELLACQUA *et al.* (2023), em colaboração com o Boston Consulting Group, demonstra que em tarefas intensivas em conhecimento, consultores equipados com acesso a LLMs como o GPT-4 não apenas completaram as tarefas mais eficientemente (25,1% mais rapidamente em média), mas também com qualidade substancialmente maior, alcançando resultados mais de 40% melhores em comparação com aqueles sem assistência de IA DELLACQUA *et al.* (2023). O aumento da produtividade dos trabalhadores do conhecimento foi de 12% em média. Para ilustrar, se considerarmos os $2,8 bilhões gastos em remuneração por uma grande empresa de petróleo, dos quais 60% vão para trabalhadores do conhecimento ($1,6 bilhões), um aumento de 12% na produtividade poderia ser visto como gerando um valor adicional de $204 milhões em produção da mesma força de trabalho. Indicadores econômicos mais amplos preveem transformações significativas devido à IA generativa (Gen-AI) em vários setores. Um relatório do Goldman Sachs HATZIUS *et al.* (2023) destaca que a Gen-AI está prestes a aumentar o PIB global em quase 7%, aumentando o crescimento da produtividade em 1,5 pontos percentuais

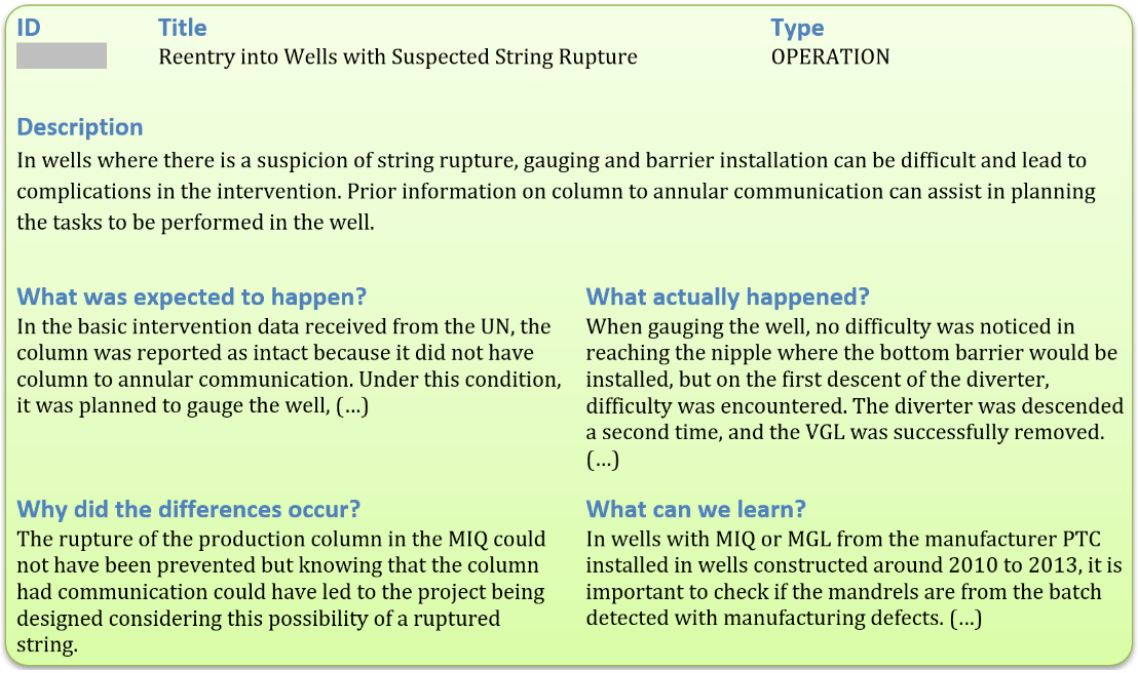
na próxima década.

1.2 Motivação

Expandindo a discussão mais ampla sobre a utilização de dados nas organizações, um problema importante é o desafio de extrair informações relevantes de extensas bases de dados SINGH *et al.* (2023). Inicialmente, o desafio de conhecer, encontrar e acessar dados representa um obstáculo significativo para os processos de tomada de decisão. Colaboradores em empresas de O&G frequentemente enfrentam a tarefa intensiva de buscar manualmente em grandes repositórios de dados para encontrar informações úteis.

Focando especificamente nas atividades de perfuração e completação de poços offshore e onshore, um grande desafio reside na natureza inerentemente complexa e técnica dos dados envolvidos, que podem ser de vários tipos: operações, projetos, tecnologias, cadeias de suprimentos e outros. A ineficiência em aproveitar grandes volumes de dados não estruturados agrava esses desafios, como observado por SINGH *et al.* (2023). Uma quantidade significativa dos dados gerados e coletados neste setor é não estruturada, variando de relatórios de texto e e-mails a imagens e vídeos de atividades de exploração e produção. As empresas de O&G enfrentam desafios na extração de informações relevantes de vastos dados não estruturados, impactando a tomada de decisões e a inovação SINGH *et al.* (2023). Exemplos incluem centenas de relatórios operacionais diários de sondas de perfuração, projetos de execução de poços, relatórios de tempo não produtivo (NPT) e documentos de lições operacionais aprendidas, conforme ilustrado na Figura 1.1. Como resultado, informações valiosas podem permanecer inexploradas e o potencial para encontrar insights, tomar decisões informadas e inovar é significativamente comprometido. SINGH *et al.* (2023) destaca as capacidades e o potencial de chatbots habilitados por IA Generativa para o setor de O&G, particularmente em melhorar a análise de perfuração e produção para alcançar melhores resultados de negócios. O autor conclui que as empresas que adotarem essas tecnologias nos próximos anos verão vantagens claras.

A implantação de tecnologias de IA enfrenta desafios como dados tendenciosos, alucinações e falta de explicabilidade HADI *et al.* (2023), necessitando de uma abordagem equilibrada. Embora pesquisas anteriores tenham abordado amplamente a IA na indústria, este estudo examina de forma única os desafios e soluções para dados não estruturados complexos em operações de O&G. Este trabalho aborda a lacuna na compreensão do desempenho de arquiteturas LLM de agente único versus multi-agente em tarefas específicas de domínio, como engenharia de poços, oferecendo insights sobre sua eficácia e relação custo-benefício. A adoção de LLM por uma grande empresa de petróleo destaca o potencial dessas tecnologias para trans-



ID [redacted]

Title
Reentry into Wells with Suspected String Rupture

Type
OPERATION

Description
In wells where there is a suspicion of string rupture, gauging and barrier installation can be difficult and lead to complications in the intervention. Prior information on column to annular communication can assist in planning the tasks to be performed in the well.

What was expected to happen?
In the basic intervention data received from the UN, the column was reported as intact because it did not have column to annular communication. Under this condition, it was planned to gauge the well, (...)

What actually happened?
When gauging the well, no difficulty was noticed in reaching the nipple where the bottom barrier would be installed, but on the first descent of the diverter, difficulty was encountered. The diverter was descended a second time, and the VGL was successfully removed. (...)

Why did the differences occur?
The rupture of the production column in the MIQ could not have been prevented but knowing that the column had communication could have led to the project being designed considering this possibility of a ruptured string.

What can we learn?
In wells with MIQ or MGL from the manufacturer PTC installed in wells constructed around 2010 to 2013, it is important to check if the mandrels are from the batch detected with manufacturing defects. (...)

Figura 1.1: Amostra de lição aprendida de perfuração e completação. Documento parcial de uma grande empresa de petróleo. (traduzido do português)

formar a análise e gestão de dados.

1.3 Objetivos

Esta pesquisa aborda diretamente os desafios enfrentados por grandes empresas de petróleo. Ao investigar as vantagens comparativas e limitações de várias arquiteturas de IA generativa (Gen-AI), incluindo sistemas de agente único e multiagente, este estudo visa identificar as soluções mais eficientes e econômicas. Os objetivos específicos desta pesquisa são avaliar a adequação e eficácia dos sistemas multiagentes baseados em LLMs para tarefas complexas e específicas de domínios na engenharia de poço, com o objetivo de otimizar o acesso à informação e a tomada de decisões. O estudo compara sistemas de IA de agente único e multiagente em termos de sua capacidade de responder a consultas relacionadas à engenharia de poços. Ele também mapeia os possíveis obstáculos e limitações associados à implantação de aplicações de Gen-AI.

Os insights obtidos com esta pesquisa visam contribuir diretamente para os objetivos estratégicos das empresas de O&G, melhorando o acesso a informações sobre engenharia de poços e tarefas de análise de dados automatizadas. Uma compreensão abrangente dos desafios e limitações associados à Gen-AI permitirá decisões informadas sobre sua adoção, maximizando o retorno sobre o investimento.

1.4 Delimitação do Escopo de Negócio

Para contextualizar o escopo deste estudo, é necessário entender o ciclo de vida de um campo de petróleo, que começa com a Exploração, progride para o Desenvolvimento da Produção, segue com a produção efetiva e culmina no descomissionamento BADIRU e OSISANYA (2016). A construção de poços, que envolve a perfuração e completação de poços para extração de hidrocarbonetos, é uma atividade dentro da fase de Desenvolvimento da Produção THOMAS (2004). A Gen-AI tem o potencial de impactar cada uma dessas fases, mas o foco deste trabalho está nas operações das fases de desenvolvimento e manutenção.

A construção de poços é uma atividade altamente especializada que envolve a perfuração e completação de poços para extração de hidrocarbonetos THOMAS (2004). Neste contexto, a Gen-AI pode ser aplicada de várias maneiras. Por exemplo, um chatbot pode gerenciar o conhecimento respondendo a perguntas sobre operações e projetos de poços, recuperando informações dos bancos de dados da organização. Além disso, agentes baseados em LLMs podem ser usados em revisões executivas de projetos para garantir que as operações de perfuração ou completação estejam em conformidade com os padrões da organização e aderem às melhores práticas operacionais. Ademais, a Gen-AI pode realizar inferências em bancos de dados não estruturados para extrair informações específicas de relatórios de texto e obter dados estruturados.

1.5 Estrutura da Dissertação

****SERÁ FEITO POR ÚLTIMO****

Capítulo 2

Revisão de Literatura

2.1 IA na indústria de petroleo

O uso de IA na indústria de Exploração e Produção (E&P) de petróleo tem sido extenso. Nas últimas décadas, a maioria das aplicações de IA na indústria envolve mineração de dados e redes neurais BRAVO *et al.* (2014). Um exemplo é o trabalho de GUDALA *et al.* (2021) sobre a otimização das propriedades de fluxo de óleo pesado, utilizando redes neurais para otimizar parâmetros que influenciam o fluxo.

Uma contribuição significativa vem da integração do conhecimento do domínio com estruturas digitais para aprimorar a tomada de decisões em tratamentos de fraturamento. Essa abordagem, demonstrada por KHAN (2024), utiliza aprendizado de máquina para melhorar a eficiência operacional e reduzir custos.

Outro desenvolvimento foi um fluxo de trabalho de aprendizado profundo proposto por GOHARI *et al.* (2024), com a geração de logs gráficos sintéticos de poços através da aplicação de aprendizado por transferência. Esses desenvolvimentos ilustram o potencial da IA em melhorar processos e a precisão e eficiência da análise de dados RAHMANI *et al.* (2021).

Processamento de Linguagem Natural (PLN) situa-se na interseção da ciência da computação e linguística, representando um domínio dentro da inteligência artificial que visa permitir que computadores compreendam e processem a linguagem humana de maneira significativa e eficaz LIDDY (2001). Este campo integra uma variedade de técnicas computacionais para analisar e representar texto em vários níveis de detalhe linguístico, buscando emular a compreensão da linguagem humana. Como uma área ativa de pesquisa, tradicionalmente o PLN emprega múltiplas camadas de análise linguística, cada uma contribuindo de forma única para a interpretação e geração de linguagem, encontrando aplicações práticas em diversos setores LIDDY (2001).

Na indústria de O&G, a gestão de dados não estruturados, como textos, imagens

e documentos, é crucial, com o Processamento de Linguagem Natural (PLN) e o Aprendizado de Máquina desempenhando papéis chave. Pesquisas de ANTONIAK *et al.* (2016) e CASTIÑEIRA *et al.* (2018) exploraram o uso de PLN para analisar riscos e relatórios de perfuração.

2.2 Modelos de Linguagem de Grande Escala (LLMs).

Os LLMs são modelos avançados baseados em redes neurais projetados para entender e gerar textos semelhantes aos humanos. Eles utilizam a arquitetura Transformer, apresentada no artigo seminal "Attention is All You Need"por VASWANI *et al.* (2017). Esta arquitetura depende de mecanismos de auto-atenção, permitindo que o modelo avalie efetivamente a importância de diferentes palavras em uma sentença.

O surgimento dos LLMs tornou possível compreender e produzir informações textuais. Espera-se que esses sistemas revolucionem várias indústrias ao apoiar processos complexos de tomada de decisão. Os modelos GPT OPENAI *et al.* (2023), em particular, aproveitam seu vasto conjunto de dados de treinamento para fornecer respostas semelhantes às humanas MOSSER *et al.* (2024), o que pode ser altamente benéfico em contextos que exigem compreensão e geração de linguagem natural.

Como destacado por SINGH *et al.* (2023), a integração de soluções baseadas em LLMs, como chatbots conversacionais, oferece uma abordagem para otimizar operações em vários segmentos de negócios, incluindo perfuração, completação e produção. SINGH *et al.* (2023) usa modelos LLMs para extrair, analisar e interpretar conjuntos de dados, permitindo a geração de insights e recomendações.

Apesar de seu impacto generalizado, os modelos de linguagem não estão isentos de limitações. Em muitas aplicações específicas da indústria, as informações críticas necessárias são frequentemente proprietárias, não compartilhadas com terceiros e, portanto, ausentes dos dados de treinamento desses LLMs MOSSER *et al.* (2024). Essa lacuna significa que os modelos GPT podem não ter acesso às informações mais atualizadas ou sensíveis necessárias para certas tarefas. Além disso, devido à sua natureza probabilística, os LLMs podem experimentar alucinações, produzindo respostas confiantes, mas incorretas ou sem sentido, com base na entrada do usuário OPENAI *et al.* (2023).

2.3 Tarefas de Pergunta e Resposta (Q&A).

A tarefa de Pergunta e Resposta (Q&A) representa um método para facilitar a transferência de conhecimento entre indivíduos dentro das organizações ISKE e BO-

ERSMA (2005). Conceitualmente, os sistemas Q&A são projetados para conectar indivíduos que possuem conhecimento específico com aqueles que buscam esse conhecimento por meio de um formato estruturado de pergunta e resposta. O papel do Q&A no cenário da documentação, exemplificado por plataformas como o Stack Overflow, destaca sua importância em disciplinas técnicas TREUDE *et al.* (2011). Esse entendimento pode orientar as organizações a tomarem decisões mais informadas sobre a implementação de tais sistemas para aprimorar a transferência de conhecimento e o aprendizado organizacional ISKE e BOERSMA (2005).

2.4 Tarefas Text-to-SQL

As tarefas de Text-to-SQL no contexto da inteligência artificial envolvem a tradução automática de perguntas ou comandos em linguagem natural para consultas SQL (Structured Query Language) estruturadas QIN *et al.* (2022). Esta é uma área importante no processamento de linguagem natural (NLP), permitindo que os usuários interajam com bancos de dados usando linguagem comum, em vez de precisar saber como escrever consultas SQL complexas.

A chegada de modelos de linguagem avançados como GPT-3 e GPT-4 OPENAI *et al.* (2023) marcou um salto significativo nas aplicações de Text-to-SQL SINGH *et al.* (2023), demonstrando capacidades notáveis no tratamento dessas tarefas. Isso pode ser atribuído ao seu extenso treinamento em conjuntos de dados diversificados DENG *et al.* (2021), que incluem não apenas grandes volumes de texto, mas também dados estruturados como tabelas e código, permitindo ao modelo entender as relações intrincadas entre linguagem e estruturas de dados. O estudo de DENG *et al.* (2023) introduz um framework de pré-treinamento para tradução de texto para SQL, enfatizando o alinhamento entre texto e tabelas nas tarefas de Text-to-SQL.

2.5 Multi-Agent Setup

Conforme demonstrado por XI *et al.* (2023), a busca pela Inteligência Artificial Geral (AGI) tem se beneficiado significativamente do desenvolvimento de agentes baseados em LLM, capazes de percepção, tomada de decisão e ação em diversos cenários. Seu estudo delineia uma estrutura fundamental para tais agentes, composta por componentes de cérebro, percepção e ação, que podem ser personalizados para várias aplicações, incluindo cenários de agente único, sistemas multi-agentes e colaboração humano-agente. A pesquisa abrangente destaca o papel crucial dos LLMs no avanço em direção à AGI, sugerindo um horizonte promissor para a eficiência operacional e os processos de tomada de decisão em configurações organizacionais complexas XI *et al.* (2023). LI *et al.* (2024) demonstrou que, através de um método de amostragem

e votação, o desempenho dos LLMs escala com o número de agentes instanciados. Outro framework de código aberto é o AutoGen WU *et al.* (2023), que permite a criação de aplicações multi-agentes LLM, possibilitando a personalização em vários modos. Ele apoia diversas aplicações em campos como matemática, programação e pesquisa operacional, demonstrando sua eficácia por meio de estudos empíricos WU *et al.* (2023).

Capítulo 3

Experimento 1

3.1 Metodologia

Esta seção descreve a abordagem e as ferramentas empregadas para investigar a eficácia de um agente baseado em modelo de linguagem em responder a consultas específicas no domínio da construção e manutenção de poços. Primeiramente, a preparação, seleção e utilização das fontes de dados são descritas, explicando como cada uma contribui para a base de conhecimento da qual o agente deriva suas respostas.

3.1.1 Preparação de Dados

Este experimento foi conduzido no departamento de construção de poços de uma grande empresa de petróleo. A escolha das tarefas foi focada em gestão de conhecimento técnico e análise de dados. Exemplos de consultas utilizadas no experimento estão listados na Tabela 3.2. As fontes de dados para a execução dessas tarefas foram escolhidas para cobrir uma variedade de cenários operacionais na atividade de construção e manutenção de poços: Itens de Conhecimento Operacional, NPTs Operacionais (Tempo Não Produtivo) e um Localizador de Colaboradores.

Tabela 3.1: Exemplo de Tabela de Números

Coluna 1	**Coluna 2**	**Coluna 3**
1	2	3
4	5	6
7	8	9
10	11	12

Itens de Conhecimento Operacional Durante intervenções de perfuração, completação e workover, documentos chamados Itens de Conhecimento são escritos por especialistas, conforme ilustrado na Fig 1.1. Esses documentos podem ser de 4 tipos: Alerta Técnico, Lição Aprendida, Boa Prática e Observação de Poço. Esta é uma

Tabela 3.2: Exemplos de consultas usados neste estudo.

Categoria	**Exemplo de Consulta**
Q&A	Como a presença de sílica na composição da pasta de cimento afeta sua estabilidade térmica a altas temperaturas?
	Quais são os principais desafios e riscos associados ao tamponamento e abandono através da tubulação em poços altamente desviados?
	O que pode causar a formação de hidratos no conector da Ferramenta de Corrida de Árvore durante a lavagem da mangueira HCR (Alta Resistência ao Colapso) antes de conectar à Árvore de Natal Molhada?
	O que pode causar a válvula de segurança do fundo do poço permanecer aberta devido à formação de hidratos nas linhas de controle?
	O que pode causar danos aos protetores de rosca e áreas de vedação das extremidades dos tubos armazenados no pátio de revestimento?
	O que pode causar alto arrasto e torque fora do fundo durante a perfuração de um poço com alta inclinação?
	Quais precauções devem ser tomadas ao realizar uma verificação superior do tampão de abandono em poços com maior inclinação?
	Quais são os fatores críticos a serem considerados ao escolher um fluido base para fabricar um tampão de suporte viscoso?
	Quais são as melhores práticas para gerenciar os parâmetros de perfuração durante o corte de cimento para evitar desgaste prematuro da broca?
Text-to-SQL	Qual foi o NPT de maior duração na sonda número 05?
	Quantos NPTs ocorreram na sonda número 06 durante agosto de 2023?

ferramenta para gestão do conhecimento, considerando o grande número e variedade de especialistas envolvidos e operações de poço realizadas.

NPTs Operacionais (Tempo Não Produtivo) A segunda fonte de dados refere-se a dados sobre anomalias ocorridas durante intervenções em poços, contendo informações como título, descrição do evento, poço onde ocorreu, tipo de operação, setor responsável, sonda envolvida, tempo perdido em horas, e datas de início e término do evento. Esses dados são críticos para a indústria, pois os NPTs representam períodos em que a operação de perfuração, completação ou manutenção é interrompida devido a algum problema técnico ou logístico. A identificação e análise desses eventos são essenciais para a melhoria contínua do processo, redução de custos e aumento da eficiência operacional. Ao entender as causas e circunstâncias desses incidentes, as organizações podem desenvolver estratégias para preveni-los no futuro, otimizando o tempo de operação.

Localizador de Colaboradores A terceira fonte de dados utilizada no experimento é um localizador de colaboradores, uma ferramenta importante dentro de uma organização para consultar e gerenciar dados de funcionários. Este sistema permite a busca rápida e identificação de colaboradores através de informações como nome, local de trabalho, empresa, matrícula e função. A importância dessa ferramenta para o experimento reside na possibilidade de cruzar dados de funcionários com outras fontes de informação para uma resposta mais completa pelo agente.

Cada uma dessas fontes fornece insumos para que o agente ofereça uma visão mais precisa e atualizada das operações e da estrutura organizacional. Um conjunto de documentos e registros foi selecionado aleatoriamente de cada banco de dados, sobre os quais foram formuladas perguntas. Para cada documento, foram geradas até 3 perguntas, resultando em um conjunto de tarefas do tipo Q&A e Text-to-SQL. Alguns exemplos são descritos na Tabela 3.2.

Neste trabalho, um agente baseado em metas RUSSELL (2020) foi implementado com o objetivo de responder com precisão a várias consultas. O agente opera em um ambiente equipado com múltiplas ferramentas para operações específicas de tarefas, como mostrado na Figura 3.1, e interage com os usuários para receber consultas.

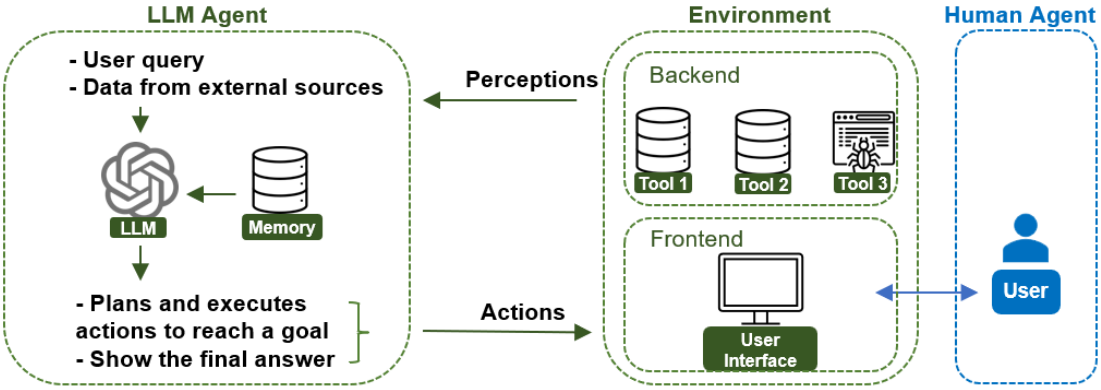


Figura 3.1: Esquemático do agente baseado em LLM interagindo com um ambiente contendo ferramentas para operações específicas de tarefas, e a interface do Agente Humano para interação e feedback do usuário.

Inicialmente, uma configuração de agentes foi implementada conforme descrito na Figura 3.2 usando o Framework AutoGen WU *et al.* (2023) com uma arquitetura que permite a recuperação de informações e interação com o usuário. Este sistema consiste em:

- **User Proxy:** representa a interface com o usuário e com ferramentas para acessar bancos de dados externos. A natureza modular das ferramentas permite que o User Proxy seja personalizado e expandido com base na variedade de fontes de dados e nos requisitos específicos do domínio de aplicação.

- **Agente:** alimentado por LLMs como GPT-4 e GPT-3, é o motor analítico do sistema. Este agente interpreta as consultas recebidas do User Proxy e formula respostas.

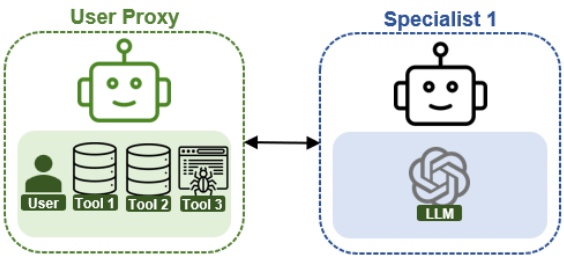


Figura 3.2: Configuração do chat com um User Proxy WU *et al.* (2023) e um Assistente.

Para cada pergunta no conjunto de dados, o processo de tomada de decisão do agente é executado conforme descrito na Figura 3.3, inicialmente selecionando a ferramenta apropriada para responder a uma consulta e, finalmente, compilando as informações recuperadas para fornecer uma resposta final.

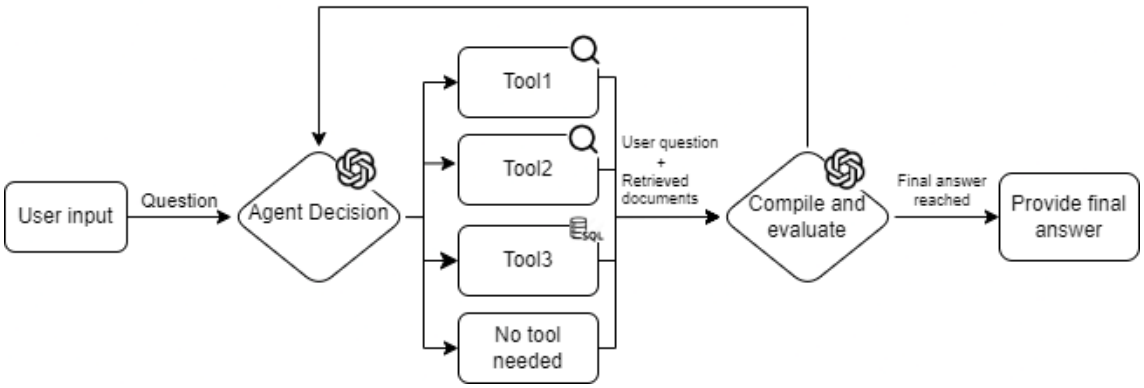


Figura 3.3: Processo de decisão do agente.

Neste experimento, três ferramentas foram consideradas no processo de tomada de decisão:

- **Ferramenta 1 - Pesquisa de Itens de Conhecimento:** uma ferramenta para pesquisar lições aprendidas que podem ser relevantes para a consulta.
- **Ferramenta 2 - Pesquisa de Colaboradores:** funcionalidade que permite a busca de informações relacionadas aos colaboradores de uma organização.
- **Ferramenta 3 - Consulta SQL NPT:** interface para execução de consultas SQL em um banco de dados de NPTs operacionais.

Paralelamente, há um caminho que permite ao Agente LLM fornecer uma resposta direta, sem a necessidade de recorrer a outras ferramentas, presumivelmente usado quando o agente já possui as informações necessárias. Finalmente, o agente apresenta a resposta final ao usuário, que é o produto de um processamento de modelo de linguagem, tomando como entradas a consulta do usuário e informações relevantes recuperadas e incluídas no contexto do prompt.

A Tabela 3.3 fornece uma análise detalhada do desempenho e precisão de diferentes modelos, especificamente GPT-3.5-turbo e GPT-4, quando consultados sobre o impacto da sílica na estabilidade térmica da pasta de cimento em altas temperaturas. A tabela compara configurações de agente único e multiagentes, avaliando as saídas finais com base na veracidade, desempenho e comentários de especialistas. Para cada consulta, a tabela destaca a relevância e precisão das informações fornecidas pelos modelos, incluindo quaisquer seções extraviadas ou não relacionadas observadas pelos especialistas. Esta comparação abrangente permite uma avaliação aprofundada das capacidades dos modelos em gerar respostas precisas e relevantes para perguntas técnicas.

3.1.2 Arquitetura Multi-Agente

Uma segunda arquitetura que emprega múltiplos agentes foi implementada, cada um tendo uma ferramenta distinta para interagir com fontes de dados externas, como ilustrado na Figura 3.4. Esta arquitetura também começa com a entrada do usuário. No entanto, como representado na Figura 3.5, o subsequente processo de 'Seleção de Palestrante' determina o agente especializado avaliado como mais adequado para responder à pergunta do usuário.

Quando uma consulta se enquadra no conhecimento direto do LLM, o caminho 'Nenhuma ferramenta necessária' é selecionado, e o agente correspondente responde sem o engajamento de outras ferramentas. O agente selecionado então 'Compila e avalia' as informações coletadas no contexto da consulta do usuário, garantindo uma resposta que seja tanto precisa quanto contextualizada. A etapa final, 'Fornecer resposta final', é onde o sistema multi-agente converge para entregar a resposta final e coerente ao usuário.

3.1.3 Avaliação

Para o processo de avaliação, em linha com a avaliação conduzida por LI *et al.* (2023), um grupo de 3 engenheiros especialistas analisou 33 perguntas e suas respectivas respostas para cada configuração. Os especialistas avaliaram cada par de perguntas e respostas com base nas seguintes métricas predefinidas:

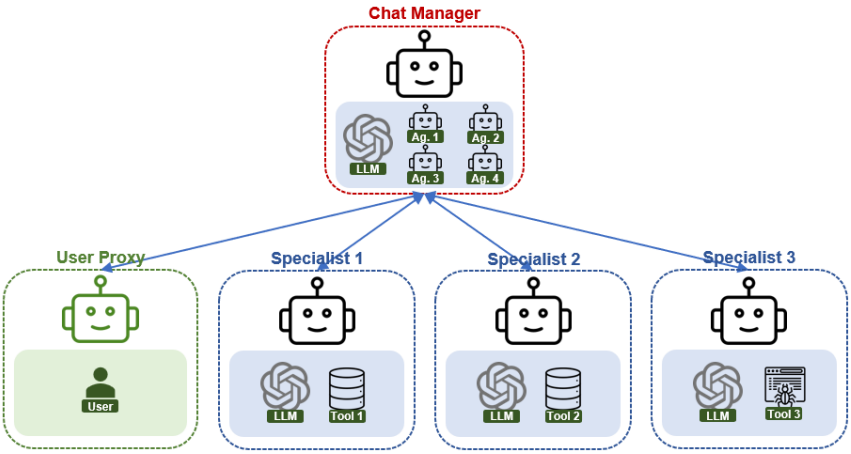


Figura 3.4: Configuração de chat com um Gerenciador de Chat e um grupo de agentes LLM.

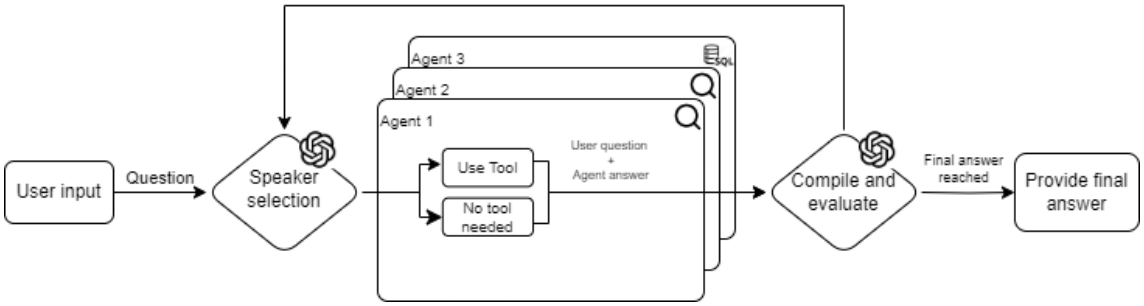


Figura 3.5: Processo de decisão multi-agente.

- **Veracidade:** uma métrica para medir o grau de divergência da precisão factual.
- **Desempenho:** engloba a qualidade geral das respostas, considerando coerência linguística, raciocínio lógico, diversidade e a presença de evidências corroborativas.

A nota final foi determinada pela média das pontuações de todas as entradas para cada configuração. Esta avaliação garantiu uma avaliação abrangente das capacidades dos modelos.

3.2 Resultados

Este capítulo fornece uma análise dos dados coletados e responde às perguntas de pesquisa. Os resultados são apresentados na Tabela 3.4 e estão organizados de acordo com os objetivos do estudo, com cada objetivo sendo abordado em detalhe.

A terceira métrica, Custo do LLM, representa o custo financeiro associado ao uso da API da OpenAI para os modelos de linguagem em cada configuração. Essa

métrica é medida em dólares americanos e reflete os recursos computacionais necessários para cada tarefa.

A análise comparativa entre as configurações de agente único e multi-agente para RAG, utilizando os modelos GPT-3.5-turbo e GPT-4, revelou insights sobre as métricas de veracidade, desempenho e custos do modelo de linguagem.

3.2.1 Veracidade

Na avaliação da métrica de veracidade, foram observadas diferenças significativas entre os cenários de agente único e multiagente nas tarefas de Perguntas e Respostas (Q&A) e Text-to-SQL. Os resultados são ilustrados nas Figuras 3.6 e 3.7. Para as tarefas de Q&A, o GPT-4 em uma configuração multiagente superou significativamente o desempenho do agente único com uma pontuação de veracidade de 4,57 em comparação com 3,88. O modelo GPT-3.5-turbo apresentou resultados distintos entre as duas configurações, com o multiagente superando o agente único com pontuações de 4,09 e 2,94, respectivamente.

Em termos de consultas Text-to-SQL, foi observado um resultado diferente. O GPT-4 agente único alcançou uma pontuação de 4,56, enquanto o mesmo modelo na configuração multiagente obteve 3,20, destacando uma limitação do multiagente nessa tarefa. Por outro lado, o GPT-3.5-turbo manteve um desempenho mais equilibrado entre as configurações, pontuando 4,29 para multiagente e 4,13 para agente único.

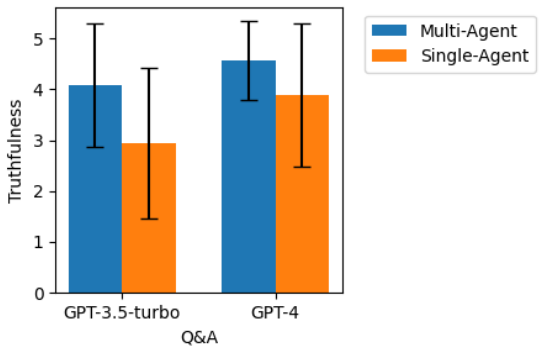


Figura 3.6: Veracidade e desvio padrão em tarefas de Q&A por modelo LLM e configuração de agente.

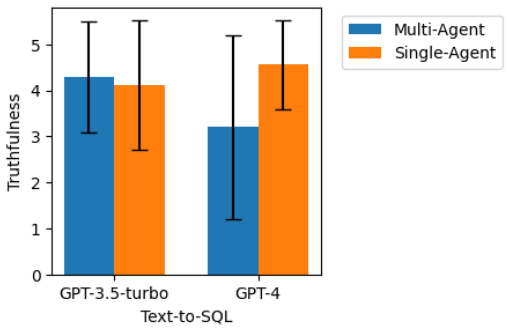


Figura 3.7: Veracidade e desvio padrão em tarefas de Text-to-SQL por modelo LLM e configuração de agente.

3.2.2 Desempenho

A avaliação do desempenho de LLM LI *et al.* (2023) nas tarefas de Q&A e Text-to-SQL revela tendências semelhantes aos resultados de veracidade. Conforme mos-

trado nas Figuras 3.8 e 3.9 e resumido na Tabela 3.4, o desempenho do texto nas configurações de agente único e multiagente foi comparado usando os modelos GPT-3.5-turbo e GPT-4.

Para as tarefas de Q&A, a configuração multiagente mostra um aumento de desempenho em comparação com o agente único. Notavelmente, o GPT-4 multiagente alcança uma pontuação de desempenho de 4,43, que é superior à pontuação de 4,06 do GPT-4 agente único. Esse padrão é consistente com o GPT-3.5-turbo, onde o sistema multiagente também supera o sistema de agente único, pontuando 3,82 e 3,94, respectivamente. Esses achados enfatizam a eficácia da abordagem multiagente em lidar com consultas técnicas dos usuários.

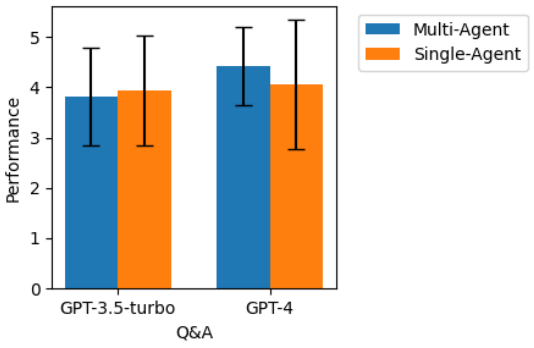


Figura 3.8: Desempenho e desvio padrão em tarefas de Q&A por modelo LLM e configuração de agente.

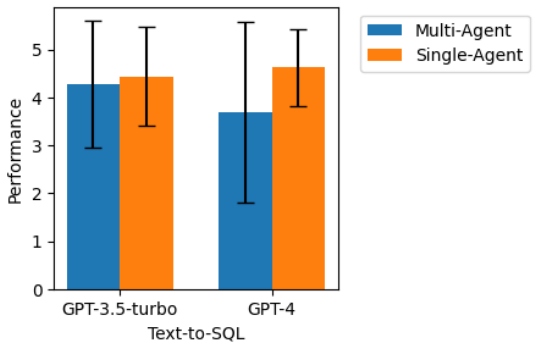


Figura 3.9: Desempenho e desvio padrão em tarefas de Text-to-SQL por modelo LLM e configuração de agente.

3.2.3 Custo de LLM

Os serviços de modelos de linguagem são tipicamente compostos por valores por token. Por exemplo, o modelo GPT-4 custa US$30,00 (entrada) e US$60,00 (saída) por 1 milhão de tokens recebidos e enviados, respectivamente.

A arquitetura de agente único demonstrou custos substancialmente mais baixos para as tarefas de Q&A e Text-to-SQL em comparação com a configuração multiagente, conforme mostrado na Figura 3.10. Por exemplo, o custo médio do modelo GPT-4 OPENAI *et al.* (2023) para uma tarefa de Q&A foi de $0,12 por pergunta processada para o agente único, enquanto o multiagente registrou um custo médio de $0,45. Essa tendência de custos mais altos para a arquitetura multiagente também foi mantida para tarefas de Text-to-SQL, com um custo médio de $0,51 para a arquitetura multiagente em contraste com $0,10 para o agente único.

O maior número de tokens e custo para a configuração multiagente deve-se à inclusão de chamadas intermediárias, por exemplo, quando o "Selector de Agen-

tes"precisa decidir para qual agente passar a vez. Todo o histórico de mensagens é passado para o LLM nesse estágio, aumentando substancialmente o número de tokens submetidos e o tempo de resposta.

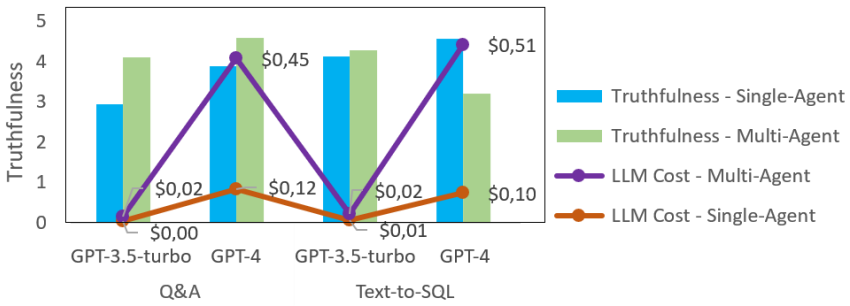


Figura 3.10: Custos médios de LLM e Veracidade por tarefa concluída de acordo com a configuração e modelo.

3.3 Discussão

A comparação entre sistemas de agente único e multiagente revelou diferenças significativas em termos de desempenho e custo:

3.3.1 Desempenho Geral.

Os resultados indicam que, para tarefas de Q&A no contexto de O&G, a medida de veracidade foi 28% maior com a arquitetura multiagente em comparação com a de agente único. No entanto, para tarefas de Text-to-SQL, essa tendência foi invertida, onde o agente único obteve uma pontuação 15% maior.

Esses achados sugerem que, para tarefas de Q&A, a configuração multiagente pode ser mais vantajosa em termos de fornecer informações verídicas, especialmente ao utilizar o modelo mais avançado GPT-4. Em contrapartida, nas tarefas de Text-to-SQL, o modelo GPT-4 em uma configuração de agente único provou ser mais eficaz. Isso pode implicar que a complexidade adicional de gerenciar múltiplos agentes em algumas tarefas não leva necessariamente a um desempenho melhor nas respostas, ressaltando a importância de selecionar cuidadosamente a configuração do agente com base no tipo de tarefa e nas características específicas do modelo de linguagem utilizado.

3.3.2 Análise de Custo-Desempenho.

Embora o sistema multiagente mostre maior veracidade nas tarefas de Q&A, é crucial considerar os custos associados. Para fornecer uma comparação mais clara,

considere as razões pontuação/custo. Para tarefas de Q&A usando GPT-4, a configuração de agente único produz uma razão de 32,33 pontos de veracidade por dólar, comparado a 10,16 para a configuração multiagente. Isso indica que, embora o sistema multiagente mostre uma melhoria de 17,8% na veracidade, isso ocorre com um aumento de custo de 275%.

Com base em nossa análise, recomendamos o uso de um sistema multiagente para tarefas de Q&A quando o orçamento permitir e a precisão for um fator crítico. No entanto, os tomadores de decisão devem considerar a definição de um limite de custo-desempenho para orientar a escolha da configuração do sistema, garantindo que os benefícios justifiquem os gastos envolvidos.

3.3.3 Variações no Desempenho dos Modelos.

Curiosamente, nossos resultados mostram que o GPT-3.5-turbo supera o GPT-4 em certas tarefas, particularmente na configuração multiagente de Text-to-SQL, apesar do maior tamanho e treinamento mais extenso do GPT-4. Esse desempenho inesperado pode ser atribuído a vários fatores. Primeiro, o GPT-3.5-turbo pode ter passado por um ajuste fino mais específico para tarefas de consulta estruturada, permitindo que ele se destaque em cenários de Text-to-SQL. Além disso, os dados de treinamento do GPT-3.5-turbo podem ser mais recentes ou mais relevantes para o domínio específico do nosso estudo. Outra possibilidade é que o menor tamanho do modelo GPT-3.5-turbo permita um processamento mais rápido e um manuseio mais eficiente da configuração multiagente, resultando em melhor desempenho em alguns contextos.

No entanto, é importante notar que o GPT-4, quando usado em uma configuração multiagente, demonstrou uma veracidade e um desempenho mais consistentes, como evidenciado por sua redução no desvio padrão nos resultados. Essa consistência pode ser particularmente vantajosa em aplicações onde confiabilidade e precisão são críticas. Sistemas multiagentes têm a vantagem de manter contextos separados para diferentes aspectos de uma tarefa. Essa compartimentalização pode levar a um melhor manuseio de consultas complexas e multifacetadas, à medida que cada agente pode se concentrar em seu contexto específico sem ser sobrecarregado por informações irrelevantes. No entanto, essa vantagem pode ser compensada em tarefas como Text-to-SQL, onde manter um contexto unificado do esquema do banco de dados e da estrutura da consulta é crucial, possivelmente explicando o melhor desempenho das configurações de agente único nessa tarefa.

A arquitetura multiagente envolve inerentemente múltiplos estágios de processamento de informações, que podem servir como mecanismos naturais de filtragem. À medida que a informação passa de um agente para outro, dados irrelevantes ou

de baixa qualidade podem ser naturalmente filtrados, levando a saídas finais mais refinadas e precisas. Isso pode explicar o desempenho superior na filtragem de informações irrelevantes observado nas configurações multiagentes.

3.3.4 Eficiência Econômica.

A arquitetura multiagente incorre em custos significativamente mais altos em comparação com o sistema de agente único, principalmente devido a chamadas intermediárias adicionais ao modelo de linguagem e múltiplas iterações entre agentes para o planejamento de ações. As diferenças de custo entre o uso do GPT-4 e do GPT-3.5-turbo são substanciais, com o GPT-4 sendo notavelmente mais caro.

Conforme detalhado na seção Análise de Custo-Desempenho, a razão de veracidade por dólar destaca os trade-offs econômicos entre sistemas de agente único e multiagente. Embora o sistema multiagente ofereça melhorias na veracidade, isso vem com um aumento considerável de custo, impactando a eficiência econômica geral.

Para uma grande empresa com 40.000 trabalhadores do conhecimento, a escolha do modelo e da arquitetura impacta significativamente os custos anuais. Usar o GPT-4 em uma configuração de agente único poderia resultar em um custo anual de aproximadamente $4,38 milhões, enquanto o GPT-3.5 custaria cerca de $438.000. No entanto, ao empregar uma arquitetura multiagente, os custos aumentam substancialmente. A configuração multiagente com GPT-4 elevaria o custo anual para $16,425 milhões, representando um aumento dramático devido ao uso de tokens 3,75 vezes maior. Da mesma forma, o GPT-3.5 em uma configuração multiagente custaria $1,642 milhões. Essas estimativas assumem um padrão médio de uso de 10.000 tokens por trabalhador por dia e ressaltam as implicações financeiras significativas da adoção de um sistema multiagente, que, embora possa oferecer benefícios de desempenho, vem com um aumento considerável nos custos de LLM.

Em resumo, enquanto sistemas multiagentes e modelos mais avançados como o GPT-4 oferecem melhorias no desempenho, a eficiência econômica, medida pela veracidade por dólar, pode favorecer sistemas de agente único e modelos menos custosos como o GPT-3.5-turbo, dependendo da aplicação específica e das restrições orçamentárias.

3.3.5 Desafios e Limitações.

Durante a avaliação dos agentes, vários desafios e limitações foram identificados.

- **Contextualização e Interpretação**: Em muitos casos, a solução de agente único teve dificuldades para entender o contexto da pergunta. Por exemplo,

uma pergunta sobre cimentação foi interpretada no contexto da indústria da construção, um tema ao qual os modelos de linguagem foram mais expostos durante a fase de treinamento. No entanto, a estrutura multiagente, com seus papéis bem definidos, compreendeu melhor as perguntas e mostrou desempenho superior nas tarefas de Q&A, corroborando os achados de LI *et al.* (2024).

- **Filtragem de Informações Irrelevantes**:

  O agente frequentemente recebe documentos irrelevantes junto com os importantes no contexto do prompt, e cabe ao LLM ignorá-los. Por exemplo, quando perguntado sobre alternativas para acelerar o tempo de cura da pasta de cimento sem comprometer sua integridade em altas temperaturas, o sistema RAG recuperou um documento que incluía informações sobre cimentação em lote para garantir homogeneidade durante a fabricação e bombeamento. Embora essa informação seja verdadeira, não era relevante para a pergunta específica feita. Nesse aspecto, a solução multiagente teve um desempenho melhor ao descartar tais informações irrelevantes, focando mais precisamente na tarefa em questão. Outras possíveis soluções incluem melhorar a precisão da busca semântica ajustando um limite mínimo para medidas de similaridade ou por meio de técnicas de reclassificação, como as propostas por CARRARO (2024) e SUN *et al.* (2023).

- **Alucinação**:

  Durante a avaliação do nosso sistema, encontramos instâncias em que o agente produziu informações alucinadas em vez de utilizar a ferramenta apropriada para recuperar dados precisos, como em BILBAO *et al.* (2023). Por exemplo, quando perguntado "Quantas anomalias ocorreram na sonda número 05 durante agosto de 2023?", esperava-se que o agente usasse a ferramenta Text-to-SQL para consultar o banco de dados. No entanto, ele ignorou essa ferramenta e gerou uma resposta fabricada, afirmando que ocorreram 5 anomalias, juntamente com descrições detalhadas de eventos fictícios. A resposta correta, conforme recuperada do banco de dados, foi que ocorreram 7 anomalias. Essa alucinação provavelmente resultou da dependência do agente em seu conhecimento interno em vez da recuperação de dados externos.

- Em termos de estatísticas de alucinação, nossa análise revelou que para tarefas de Q&A, as alucinações ocorreram em 9,6% dos casos e 3,8% foram parcialmente alucinadas. Em contraste, as tarefas de Text-to-SQL exibiram uma taxa de alucinação menor, com apenas 3,6% das respostas contendo informações alucinadas e 96,4% sendo precisas. Esses achados destacam a suscetibilidade variável à alucinação entre diferentes tipos de tarefas, enfatizando a

necessidade de estratégias direcionadas para mitigar esse problema.

- **Jargão da Indústria**:

  Analisando especificamente a atividade de perfuração e completação de poços offshore, o principal desafio é a natureza inerentemente complexa e técnica dos dados envolvidos. Houve instâncias de interpretação incorreta da informação, provavelmente devido ao uso de termos, expressões e temas específicos da construção de poços, aos quais o modelo de linguagem teve pouca ou nenhuma exposição durante a fase de treinamento. Uma possível solução é a implementação de modelos especializados, que tem sido apontada na literatura cinza como uma tendência para os próximos anos SHAH (2024); MEENA (2023); GHOSH (2023).

- **Ferramentas vs. Desempenho**:

  Foi identificado durante os experimentos que agentes com uma alta quantidade de ferramentas mostraram um declínio no desempenho geral. Isso pode ser atribuído ao contexto adicionado aos prompts. À medida que o comprimento do contexto aumenta, a capacidade do modelo de interpretar e responder com precisão diminui. Esta é uma limitação dos modelos de linguagem atuais, onde contextos mais longos podem levar a uma diluição de informações relevantes e aumentar a dificuldade em manter a coerência e a precisão. Esta conclusão é atualmente qualitativa, pois essas métricas não foram abordadas neste experimento.

- **Consultas Envolvendo Nomes Próprios**:

  Em consultas envolvendo nomes de pessoas, não foi possível recuperar documentos relevantes usando a busca semântica. Por exemplo, quando solicitado a identificar o funcionário associado a uma chave específica e listar os itens de conhecimento que registraram no sistema, o sistema RAG atribuiu incorretamente itens de conhecimento ao autor errado. Isso destaca a dificuldade em recuperar informações com precisão baseadas em nomes próprios, que podem ser complicadas por variações em acentuação, abreviação e formatação. Uma solução potencial a ser explorada é o uso do Self-Query Retriever LANGCHAIN (2023), implementando uma busca híbrida com filtros de metadados (incluindo nomes próprios) e recuperação semântica do restante da consulta. Também é sugerido, nesses casos, usar a distância de LEVENSHTEIN (1966) para lidar com possíveis variações na grafia dos nomes. Essa abordagem poderia melhorar a precisão na recuperação de documentos relacionados a indivíduos específicos, garantindo que as informações corretas sejam associadas à pessoa certa.

3.3.6 Implicações Práticas.

Os achados do nosso estudo têm implicações práticas significativas para o setor de O&G e, potencialmente, para outras indústrias caracterizadas por ambientes de dados complexos e técnicos:

- **Apoio Aprimorado à Tomada de Decisões:** Nossos resultados indicam que sistemas multiagentes fornecem uma medida de veracidade 28% maior em tarefas de Q&A. Isso pode ser particularmente benéfico para a tomada de decisões em engenharia de poços, onde informações precisas e verídicas são críticas. Implementar sistemas multiagentes nos processos de tomada de decisão pode levar a decisões mais confiáveis e informadas, reduzindo o risco de erros e aumentando a segurança e eficiência operacional.

- **Equilíbrio entre Desempenho e Eficiência Econômica:** Embora sistemas multiagentes ofereçam desempenho superior em termos de veracidade, eles vêm com um custo que é, em média, 3,7 vezes maior em comparação com sistemas de agente único. Isso destaca a importância de uma abordagem estratégica na seleção de configurações de agentes com base em tarefas específicas e restrições orçamentárias. Uma análise detalhada de custo-benefício revela que, para tarefas de Q&A usando GPT-4, a configuração de agente único produz uma razão de 32,33 pontos de veracidade por dólar, comparado a 10,16 para a configuração multiagente. Enquanto o sistema multiagente mostra uma melhoria de 17,8% na veracidade, isso vem com um aumento de custo de 275%. A eficiência varia significativamente por tipo de tarefa; em tarefas de Text-to-SQL, o GPT-4 agente único supera o multiagente em 42,5% na veracidade enquanto custa 80,4% menos.

- **Agentes de Reflexão e Crítica:** Uma abordagem promissora para melhorar o desempenho desses agentes é o uso da reflexão SHINN *et al.* (2023), um método onde os agentes refletem verbalmente sobre sinais de feedback da tarefa e mantêm esse texto reflexivo em um buffer de memória episódica para melhorar a tomada de decisões em tentativas subsequentes. Agentes críticos são uma forma de implementar a reflexão em uma configuração multiagente. Esse tipo de agente é desafiador de aplicar em tarefas de Q&A sobre dados técnicos privados, pois LLMs comerciais (OpenAI, Google Bard e outros) não foram profundamente treinados no domínio e têm dificuldade em fornecer críticas relevantes e precisas, reforçando a tendência de uso crescente de modelos específicos de domínio SHAH (2024); MEENA (2023); GHOSH (2023).

- **Configuração de Agentes Específica para a Tarefa:** O estudo destaca que a complexidade de gerenciar múltiplos agentes nem sempre leva a um

desempenho melhor. Em alguns casos, uma configuração de agente único pode ser mais eficaz. Essa percepção pode orientar o desenvolvimento e a implantação de sistemas de IA, garantindo que a configuração dos agentes seja adaptada aos requisitos específicos da tarefa, otimizando tanto o desempenho quanto o custo.

- **Potencial para Aplicação Ampla:** As percepções obtidas deste estudo não se limitam ao setor de O&G, mas podem ser aplicadas a outras indústrias com complexidades técnicas similares, como aeroespacial, farmacêutica e energia renovável. Ao adotar sistemas multiagentes nessas indústrias, as organizações podem melhorar a tomada de decisões, a gestão do conhecimento e a eficiência operacional, impulsionando a inovação e a competitividade.

3.3.7 Futuras Direções

Este trabalho indica possíveis caminhos para aprimorar as arquiteturas RAG no setor de O&G.

- **Aprimoramento das Técnicas Semânticas de IR:** Há uma necessidade crítica de desenvolver tecnologias de busca semântica mais sofisticadas. Esforços futuros devem se concentrar em aumentar a precisão da recuperação de informações, filtrando conteúdos irrelevantes de maneira mais eficaz. Isso garantirá que os agentes possam fornecer respostas mais precisas e contextualmente adequadas, crucial para domínios técnicos como O&G.

- **Desenvolvimento de Modelos Específicos de Domínio:** Modelos especializados, feitos especificamente para O&G e outros domínios, como engenharia biomédica PAL *et al.* (2024), poderiam melhorar significativamente o manuseio de jargões específicos e dados técnicos complexos, ao mesmo tempo que reduzem os custos de LLM AREFEEN *et al.* (2024). Pesquisas futuras devem visar desenvolver e treinar esses modelos para entender e interpretar melhor a linguagem e os tipos de dados únicos encontrados em O&G, melhorando a precisão geral das respostas dos agentes.

- **Otimização do Uso de Ferramentas no Desempenho dos Agentes:** A relação entre a quantidade de ferramentas disponíveis para um agente e seu desempenho precisa de mais exploração. Estudos futuros devem quantificar o impacto da disponibilidade de ferramentas na eficácia e eficiência do agente, visando otimizar o uso das ferramentas sem sobrecarregar o agente ou diluir a qualidade do desempenho.

- **Integração de Técnicas Avançadas de Reconhecimento de Nomes:** Consultas que envolvem nomes próprios representam um desafio significativo na busca semântica. A integração de técnicas avançadas de recuperação, como Self-Query Retrievers LANGCHAIN (2023) e algoritmos de distância LEVENSHTEIN (1966), poderia melhorar o tratamento dessas consultas. Pesquisas futuras devem se concentrar em aprimorar as capacidades de reconhecimento de nomes para garantir que os agentes possam recuperar e utilizar informações corretas com precisão, especialmente em cenários onde a precisão é fundamental.

- **Extensão para Outros Domínios Complexos:** As potenciais aplicações de sistemas multiagentes não se limitam ao setor de O&G. Pesquisas futuras devem explorar a adaptação e implementação desses sistemas em outros domínios complexos e técnicos, como aeroespacial, farmacêutico e energia renovável. Investigar como esses sistemas podem apoiar a tomada de decisões nessas áreas fornecerá insights valiosos sobre sua versatilidade e adaptabilidade.

- **Experimentação com Modelos Híbridos:** Combinar as forças de sistemas de agente único e multiagente pode trazer benefícios significativos. As direções futuras devem incluir a experimentação com modelos híbridos que integrem a robustez e profundidade das interações multiagentes com a simplicidade e eficiência dos sistemas de agente único. Essa abordagem híbrida poderia potencialmente oferecer uma solução equilibrada, maximizando o desempenho enquanto gerencia custos e complexidade.

Ao seguir essas direções, a pesquisa futura pode avançar significativamente no desenvolvimento de sistemas multiagentes, não apenas aprimorando sua aplicação no setor de O&G, mas também expandindo sua utilidade em várias atividades tecnologicamente intensivas.

Tabela 3.3: Amostra de consulta com entradas, saídas e avaliações.

Consulta	**Modelo**	**Config.**	**Saída**	**Veracidade**	**Desempenho**	**Comentários**
Como a presença de sílica na composição da pasta de cimento afeta sua estabilidade térmica em altas temperaturas?	GPT-3.5-turbo	Agente único	A presença de sílica na composição da pasta de cimento pode afetar sua estabilidade térmica em altas temperaturas devido a algumas características da sílica. A sílica é um material inorgânico que possui uma estrutura cristalina estável em altas temperaturas. Quando exposta a altas temperaturas, a sílica presente na pasta de cimento pode sofrer transformações estruturais, resultando em mudanças nas propriedades da pasta. [...]	4	5	Criou uma seção que é irrelevante para a questão: "Além disso, a presença de sílica na composição da pasta de cimento pode afetar a resistência térmica do material. [...]"
		Multiagente	A presença de sílica na composição da pasta de cimento tem um efeito positivo na estabilidade térmica em altas temperaturas. A sílica é adicionada ao cimento para mitigar a retrogressão da resistência do cimento quando exposto a temperaturas acima de 230°F (110°C). Poços profundos ou aqueles com alto gradiente geotérmico podem exigir o uso de cimento com sílica para garantir a estabilidade térmica da pasta. [...]	2	2	Recuperou uma Lições Aprendidas não relacionada ao assunto (sobre plugs).
	GPT-4	Agente único	A presença de sílica na composição da pasta de cimento aumenta a estabilidade térmica, especialmente em altas temperaturas. A sílica é usada para transformar o cimento em um material refratário, capaz de suportar condições extremas. [...]	3	4	Embora seja verdade, o item ao lado não é relevante para a questão: "No entanto, é importante notar que [...]."
		Multiagente	A sílica é adicionada ao cimento em uma proporção de 35% em peso de cimento para mitigar a retrogressão (perda de resistência à compressão) do cimento no poço quando submetido a temperaturas acima de 230ºF (110ºC). [...]	5	4	Citou uma Lições Aprendidas não relacionada (mas verdadeira) sobre plugs, Lições Aprendidas LA7005.

Tabela 3.4: Resultados nas tarefas de Q&A e Text-to-SQL, incluindo desvio padrão (Std). As melhores métricas estão destacadas em **negrito e sublinhado**. As segundas melhores estão destacadas em **negrito**.

Tarefa	**Agente Único**					**Multi-Agente**				
Modelo	**Custo LLM**	**Verdade**	**Std**	**Desempenho**	**Std**	**Custo LLM**	**Verdade**	**Std**	**Desempenho**	**Std**
Q&A										
GPT-3.5-turbo	0.005	2.94	1.48	3.94	1.09	0.02	4.09	1.22	3.82	0.98
GPT-4	0.12	**3.88**	1.41	**4.06**	1.30	0.45	**4.57**	0.79	**4.43**	0.79
Text-to-SQL										
GPT-3.5-turbo	0.009	4.13	1.41	4.44	1.03	0.02	**4.29**	1.20	**4.29**	1.33
GPT-4	0.10	**4.56**	0.96	**4.63**	0.81	0.51	3.20	1.99	3.70	1.89

Capítulo 4

Experimento 2

4.1 Metodologia 2

...

Algorithm 1 Experiment Execution Loop

Require: questions, setups, models
Ensure: results

```
function RunExperiment
    results ← {}
    for all question ∈ questions do
        ground_truth ← question.ground_truth
        for all setup ∈ setups do
            for all model ∈ models do
                agent ← InitializeAgent(setup, model)
                response ← agent.ProcessQuestion(question)
                metrics ← EvaluateResponse(response, ground_truth)
                results[question, setup, model] ← {
                        "response" : response,
                        "metrics" : metrics,
                        "execution_trace" : agent.trace
                }
            end for
        end for
    end for
    return AggregateResults(results)
end function
```

4.1.1 Dataset de Q&A

...

4.1.2 Setups

Linear-Flow with Router

...

Single-Agent

...

Multi-Agent

...

4.1.3 Frameworks utilizados

...

4.1.4 Avaliação de desempenho

[Falar brevemente sobre métricas, prompts, citar ragas, etc]

4.2 Methodology (generated)

4.2.1 1. Overview

This chapter describes the experimental methodology used to evaluate large language model (LLM) agents and workflows for answering questions in the oil well operations domain. The experiment integrates multiple LLM configurations, agent architectures, and retrieval-augmented generation (RAG) tools, leveraging Petrobras datasets.

<insert brief summary of research objectives and hypotheses here>

4.2.2 2. Experimental Workflow (Expanded)

2.1 Dataset Preparation

The experimental workflow was designed to provide a thorough and reproducible evaluation of language model agents within oil well operations. The process begins with the careful preparation of the dataset, which is composed of questions and corresponding ground truth answers derived from a diverse range of operational records, incident reports, and lessons learned. To ensure the quality and relevance of the data, questions undergo a filtering and preprocessing phase where clarity,

diversity, and alignment with real-world scenarios are prioritized. This includes removing duplicates, standardizing terminology, and confirming that each question is properly paired with an accurate answer. The dataset is further validated for completeness and consistency, ensuring it represents the full spectrum of operational challenges, such as safety, cementing, and intervention scenarios.

2.2 Model and Setup Selection

Following dataset preparation, the experimental design incorporates a variety of agent architectures. These include approaches where questions are routed to specialized agents, single-agent systems that centralize all reasoning and retrieval, and multi-agent frameworks that leverage collaboration among specialized agents under a supervisory structure. Each of these configurations is evaluated using different language models, allowing for a comprehensive assessment of how model choice and agent setup influence performance. The agents are also provided with access to advanced retrieval tools and domain-specific knowledge bases, enabling them to draw on a broad foundation of operational expertise.

2.3 Execution Loop

The core of the experimental workflow is an automated execution loop. For each combination of question, agent setup, and language model, the system systematically loads the relevant data, configures the agent, and executes the workflow. Throughout this process, all responses and intermediate reasoning steps are meticulously logged. This approach not only ensures systematic coverage of all experimental conditions but also provides full traceability for subsequent analysis. The automation of these procedures guarantees consistency and reproducibility, while the comprehensive logging facilitates in-depth evaluation and comparison of agent performance across a range of operational scenarios.

2.4 Evaluation and Metrics

Following the execution of all experimental combinations, a comprehensive evaluation framework is applied to assess agent performance. The system calculates a suite of quantitative metrics for each question, setup, and model combination by comparing the generated answers against the established ground truth. These metrics include standard performance indicators such as accuracy, precision, recall, and F1 score, which provide a multifaceted view of response quality. For open-ended questions where binary correctness measures are insufficient, a confusion matrix approach is implemented to capture nuances in answer quality and content coverage. Additionally, the system measures answer size ratio relative to ground truth, offering

insights into model verbosity and conciseness. These metrics are then aggregated across different dimensions to enable meaningful comparisons between agent architectures and language models, revealing patterns in performance across various operational scenarios and question types.

2.5 Reproducibility and Quality Control

To ensure scientific rigor and reproducibility, the experimental methodology incorporates robust tracking of all environmental variables and configuration parameters. The system maintains detailed logs of the computational environment, including software versions, dependency specifications, and hardware characteristics that might influence results. All experimental parameters, from model identifiers to dataset specifications, are systematically recorded alongside the results they generate. Throughout the experimental process, periodic validation checks are performed to maintain data integrity and result consistency, with anomalies flagged for investigation. This comprehensive approach to reproducibility not only facilitates verification of findings but also enables future extensions of the research with comparable baselines. The quality control measures embedded in the workflow ensure that conclusions drawn from the experiments rest on a foundation of methodological soundness and data reliability.

<insert workflow diagram or pseudocode here to illustrate the above stages>

4.2.3 3. Data Sources

The experimental evaluation relies on a carefully curated collection of data sources that represent the diverse knowledge domains relevant to oil well operations. At the core of the experiment is a comprehensive questions dataset containing structured entries that simulate real-world queries an operator might encounter. This dataset was developed through extensive collaboration with domain experts and analysis of historical operational records. Each entry in the dataset contains a question formulated in natural language, a unique identifier, categorical metadata to facilitate analysis, and a corresponding ground truth answer validated by subject matter experts. The questions span various complexity levels, from factual inquiries to complex reasoning scenarios that require integration of multiple knowledge sources.

To provide the language models with the necessary domain knowledge, the experiment incorporates several specialized knowledge bases that reflect different aspects of oil well operations:

- **Knowledge Bases and Tools**:

- **Lessons**: A repository of knowledge items capturing insights, best practices, and technical know-how from past oil well operations. These lessons represent institutional memory and expertise accumulated over years of operational experience.
- **Alertas SMS**: A collection of safety alerts and incident reports documenting past events, near-misses, and accidents, providing critical safety information and preventative measures.
- **Cronoweb**: A comprehensive database of scheduling information and intervention records, detailing maintenance activities, equipment deployments, and operational timelines.
- **SITOP**: Detailed daily operational logs from drilling rigs, containing technical parameters, operational decisions, and situational reports from active drilling operations.

These knowledge sources were preprocessed to ensure consistency, remove sensitive information, and optimize retrieval performance. The integration of these diverse data sources enables a holistic evaluation of how language model agents navigate the complex informational landscape of oil well operations, from technical specifications to safety protocols and historical precedents.

<insert table or image summarizing datasets and tools here>

4.2.4 4. System Architecture

The experimental system was implemented using modern Python frameworks specialized for language model orchestration and agent workflows. The architecture leverages the LangChain and LangGraph ecosystems, which provide robust foundations for building complex language model applications with multiple components and state management. This subsection details the modular design of the system, highlighting how different components interact to enable systematic evaluation of language model agents in oil well operations.

4.1 Experiment Orchestration

At the core of the system architecture is an experiment orchestration layer responsible for coordinating the entire evaluation process. This component manages the loading of questions from the dataset, systematically iterates through different model and setup combinations, and ensures proper logging of results. The orchestrator maintains experiment state across multiple runs, handles error recovery, and implements checkpointing to allow for resumption of long-running experiments. By centralizing control flow, this component ensures that all experimental conditions

are tested consistently and that results are captured in a standardized format for subsequent analysis.

4.2 Agent Workflow Frameworks

The system implements multiple agent workflow frameworks to evaluate different approaches to question answering in the oil well domain. These frameworks define the flow of information and decision-making processes within and between language model agents. The implemented workflows include a Linear-Flow with Router (CORTEX) that directs questions to specialized processing paths, a Single-Agent approach that centralizes all reasoning and tool use, and a Multi-Agent Supervisor framework that coordinates multiple specialized agents. Each workflow is defined declaratively, specifying the sequence of operations, decision points, and information exchange patterns that govern agent behavior during question processing.

4.3 Nodes and Tool Integration

The system architecture includes specialized nodes that implement specific reasoning steps and tool-calling logic. These nodes serve as the building blocks of agent workflows, encapsulating discrete functionality such as question analysis, knowledge retrieval, and answer synthesis. The tool integration layer provides agents with access to external knowledge sources through a standardized interface, enabling semantic search over domain-specific corpora, structured data queries, and other specialized operations. This modular approach to tool integration allows for consistent evaluation of how different agent architectures leverage available tools and knowledge sources.

4.4 Prompt Engineering and System Messages

A critical component of the architecture is the prompt engineering layer, which defines the instructions and context provided to language models. This includes carefully crafted system messages that establish the role and capabilities of each agent, prompt templates that structure inputs consistently across experimental conditions, and few-shot examples that guide model behavior. The system maintains a library of prompt variants optimized for different tasks within the question-answering workflow, ensuring that each agent receives appropriate guidance while maintaining experimental control.

4.5 State Management and Metrics

The architecture incorporates a comprehensive state management system that tracks the progress of experiments, maintains contextual information across agent interac-

tions, and captures intermediate reasoning steps. This component is tightly integrated with the metrics calculation subsystem, which computes performance indicators in real-time as experiments progress. The metrics framework implements various evaluation approaches, from simple accuracy measures to sophisticated semantic similarity calculations, providing multi-dimensional assessment of agent performance. All experimental data, including intermediate states and final results, is persisted in structured formats to enable both immediate feedback and in-depth post-experiment analysis.

<insert system architecture diagram here>

4.2.5 Experimental Setups

To comprehensively evaluate language model performance in well construction operations, the experiment employed multiple agent architectures and model configurations. This subsection details the different experimental setups, highlighting their design principles, operational characteristics, and the rationale behind their selection. The experimental design deliberately incorporates contrasting approaches to agent architecture, enabling comparative analysis of different strategies for complex question answering in specialized domains.

Linear-Flow

The Linear-Flow architecture represents the simplest RAG design, where user input is processed in a strictly sequential manner.

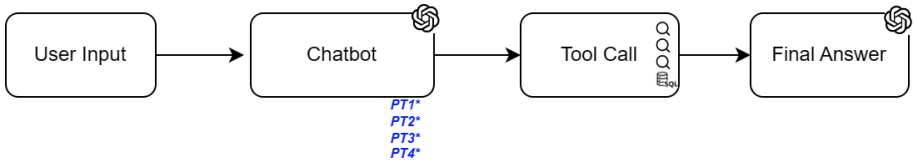


Figura 4.1: Linear-Flow architecture. PT1 indicates Prompt for Tool 1 and so on.

Linear-Flow with Router

The Linear-Flow with Router paradigm extends the basic linear flow by introducing a routing mechanism that enables the distribution of tool instruction prompts. As illustrated in Figure 4.2, instead of a single chatbot generating one query and invoking a single tool, the router decomposes the user input into multiple sub-queries. Each sub-query is then processed independently, often in parallel, by separate tool invocations.

This approach offers several advantages:

- **Increased Throughput:** By distributing sub-tasks across multiple tools, the system can handle more complex or multi-faceted user requests efficiently.
- **Specialization:** Each tool can be tailored to address a specific aspect of the user's query, allowing for more accurate and relevant results.
- **Scalability:** The architecture naturally supports scaling, as additional tools can be added to handle more sub-queries or specialized tasks.

In practice, the router acts as an orchestrator, analyzing the user input and generating multiple targeted queries (PT1*, PT2*, PT3*, PT4* in the figure). These queries are dispatched to their respective tools, and the results are aggregated to form the final answer. This method is particularly effective for tasks that can be decomposed into independent components, such as multi-part questions or workflows requiring different types of expertise.

Compared to the standard linear flow, the use of a router introduces additional complexity in query generation and result aggregation but enables a significant boost in system flexibility and performance.

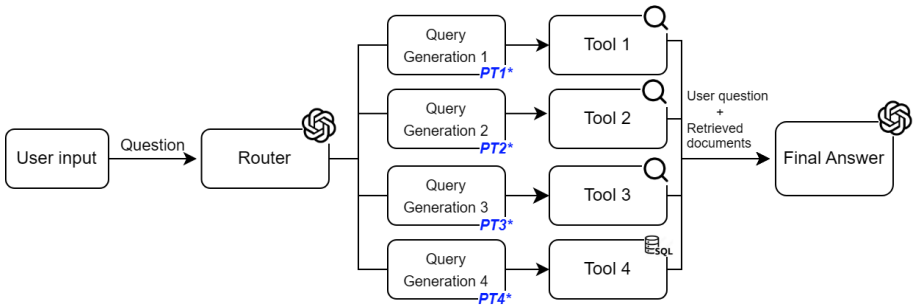


Figura 4.2: Linear-Flow with Router architecture.

Single-Agent

The Single-Agent approach represents a centralized architecture where a single language model agent handles the entire question-answering process. This agent has access to the full suite of retrieval tools and knowledge sources, making independent decisions about which tools to invoke and how to synthesize information into coherent answers. The design emphasizes end-to-end reasoning within a unified context, allowing the model to maintain a consistent understanding throughout the process. This approach tests the capability of language models to manage complex workflows autonomously, balancing between exploration of different knowledge sources and focused answer generation without the overhead of inter-agent communication.

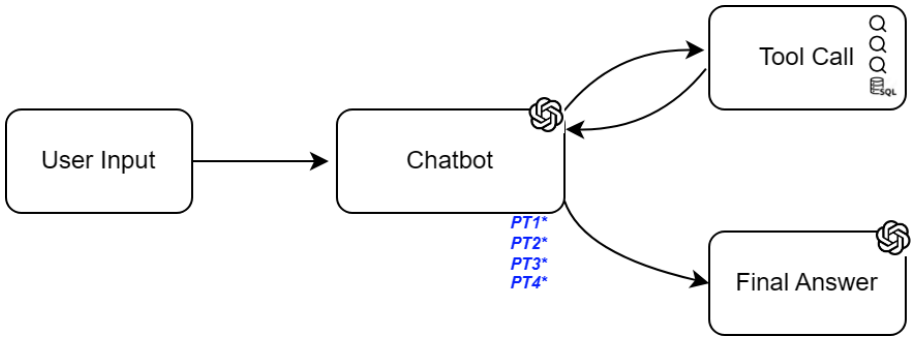


Figura 4.3: Single-Agent architecture

Multi-Agent Supervisor

The Multi-Agent Supervisor setup implements a collaborative approach where multiple specialized agents work together under the coordination of a supervisor agent. Each specialized agent focuses on a specific domain of knowledge or reasoning skill, such as retrieval, analysis, or explanation generation. The supervisor agent orchestrates the collaboration, delegating subtasks to appropriate specialized agents, integrating their contributions, and ensuring coherence in the final answer. This architecture explores the potential benefits of distributed cognition, where complex reasoning is decomposed into manageable components handled by purpose-built agents. The framework includes mechanisms for resolving conflicts between agents and synthesizing potentially divergent perspectives into unified responses.

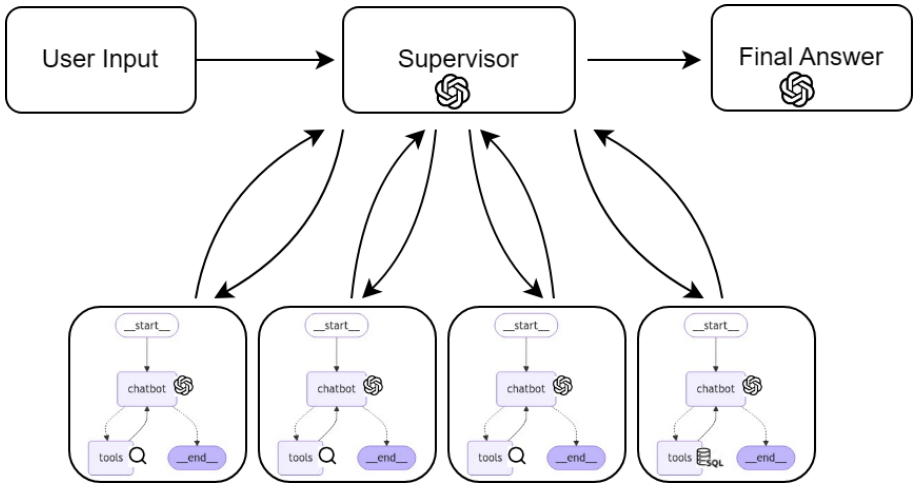


Figura 4.4: Multi-Agent setup with one supervisor and 4 specialist agents.

<insert table summarizing experimental setups and models here>

4.2.6 6. Execution Details

The experiment was driven by a script without manual intervention during the evaluation process. A main execution loop systematically iterated through all combi-

nations of questions, agent setups, and language models defined in the experimental design.

6.1 Tool Integration and Knowledge Access

During execution, the agent systems accessed domain-specific knowledge through a standardized tool interface layer. This layer provided consistent access patterns across all experimental configurations, ensuring that differences in performance could be attributed to agent architecture rather than variations in knowledge availability. The tool integration framework supported a diverse range of knowledge access methods, including semantic search over unstructured text corpora, structured queries against relational databases, and specialized information extraction routines tailored to the oil well operations domain. Each tool invocation was executed within a controlled environment that captured performance metrics such as latency and resource utilization, providing additional dimensions for analysis beyond answer correctness. The standardization of tool interfaces across agent architectures was a critical design decision that enabled fair comparison while still allowing each architecture to implement its own strategy for tool selection and result interpretation.

6.2 Comprehensive Logging and Observability

A cornerstone of the experimental methodology was the implementation of comprehensive logging throughout the execution process. The system captured detailed records of each step in the question-answering workflow, from initial question parsing to final answer generation. These logs included intermediate reasoning steps, tool invocations with their inputs and outputs, and internal state transitions within the agent systems. All experimental artifacts were persisted in structured formats that facilitated both automated analysis and manual inspection. The logging system implemented a hierarchical organization that linked high-level metrics to the detailed execution traces that produced them, enabling root cause analysis of performance patterns. This observability infrastructure was essential for understanding not just what results were produced, but how and why different agent architectures arrived at their answers, providing insights into their reasoning processes and failure modes.

<insert code snippet or pseudocode of main execution loop here>

4.2.7 7. Evaluation Metrics

The evaluation of each experimental run is grounded in a comprehensive set of metrics designed to capture both the correctness and the quality of the system's responses. Standard quantitative measures such as accuracy, precision, recall, and F1 score are calculated by comparing the answers generated by the agent systems to

the established ground truth for each question. These metrics provide a multifaceted view of performance, indicating not only how often the system produces correct answers but also how well it balances false positives and false negatives.

For questions that are open-ended or less amenable to binary correctness, the evaluation framework employs a confusion matrix approach. This allows for a more nuanced assessment, capturing partial correctness and the degree to which the system's response overlaps with the expected content. Additionally, the methodology includes the calculation of the answer size ratio, which measures the verbosity of the generated answer relative to the ground truth. This metric helps to identify tendencies toward overly concise or excessively verbose responses, offering further insight into the models' behavior and suitability for practical deployment.

4.2.8 8. Reproducibility

Ensuring reproducibility is a cornerstone of the experimental methodology. To this end, every aspect of the computational environment is meticulously documented. This includes recording the exact Python version used, as well as all package dependencies and their respective versions. Hardware specifications, such as processor type and available memory, are also logged to account for any potential influence on experimental outcomes.

Beyond the environment, the system systematically records all configuration parameters relevant to each experimental run. This encompasses model names, hyperparameters, and dataset paths, as well as any other settings that might affect the results. By maintaining this comprehensive record, the methodology enables other researchers to replicate the experiments precisely or to build upon them with confidence that baseline conditions are well understood and controlled.

4.2.9 9. Limitations

While the experimental methodology strives for rigor and comprehensiveness, several limitations must be acknowledged. One key limitation concerns the coverage of the dataset: although the question set is carefully curated to represent a broad range of operational scenarios, it may not capture the full diversity of real-world challenges encountered in oil well operations. Similarly, the models and agent architectures evaluated are constrained by the available computational resources and the current state of language modeling technology, which may limit their ability to generalize beyond the scenarios tested.

Another limitation arises from the reliance on ground truth answers, which, despite expert validation, may still reflect subjective judgments or incomplete information in certain cases. Furthermore, the evaluation metrics, while robust, may

not fully capture qualitative aspects of answer usefulness or clarity, especially in highly technical or ambiguous situations. Recognizing these limitations is essential for interpreting the results and for guiding future research aimed at addressing these gaps.

4.2.10 10. Summary

This methodology provides a systematic framework for comparing different agent architectures and large language models in the context of complex question answering for oil well operations. By integrating rigorous evaluation metrics, robust reproducibility practices, and a clear acknowledgment of limitations, the approach enables meaningful insights into the strengths and weaknesses of various system designs. The findings derived from this methodology can inform both the deployment of language model agents in operational settings and the ongoing development of more capable and reliable AI systems for specialized industrial domains.

4.3 Resultados e Discussão

4.3.1 Performance

[GERAR TEXTO AQUI]

...

...

...

[GERAR GRÁFICO NOVO AQUI onde config são as cores e X os modelos]

...

...

...

F1 Score

[GERAR TEXTO AQUI]

...

...

...

[GERAR TEXTO AQUI]

...

...

...

[GERAR TEXTO AQUI]

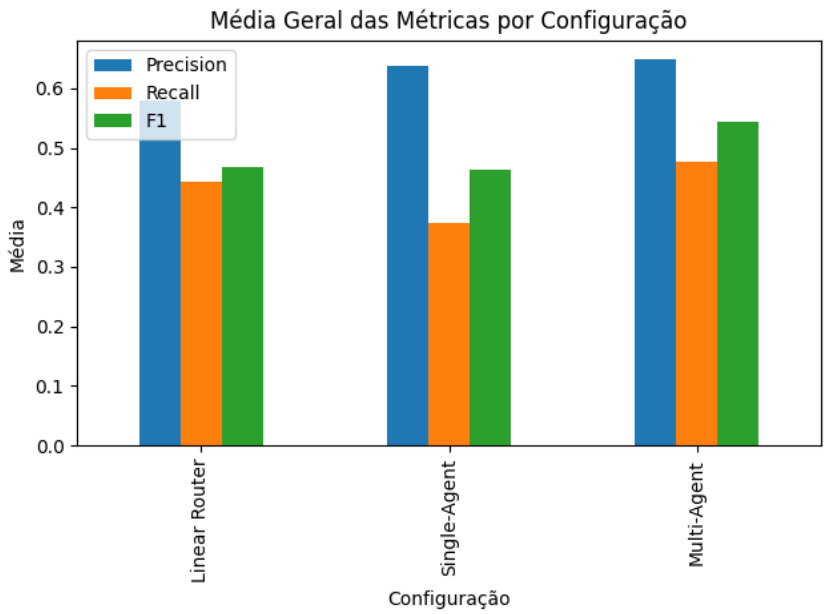


Figura 4.5: Precisão, recall e f1 por configuração. [GERAR GRÁFICO NOVO AQUI]

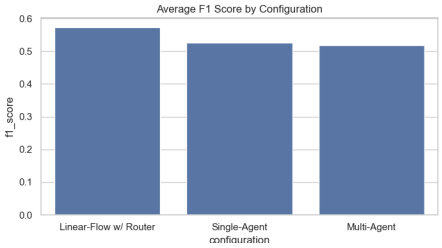


Figura 4.6: Image 1 caption

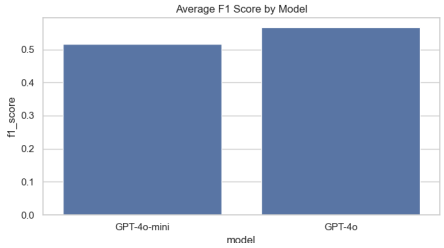


Figura 4.7: Image 2 caption

...

...

...

[GERAR TEXTO AQUI]

...

...

...

[GERAR TEXTO AQUI]

...

...

...

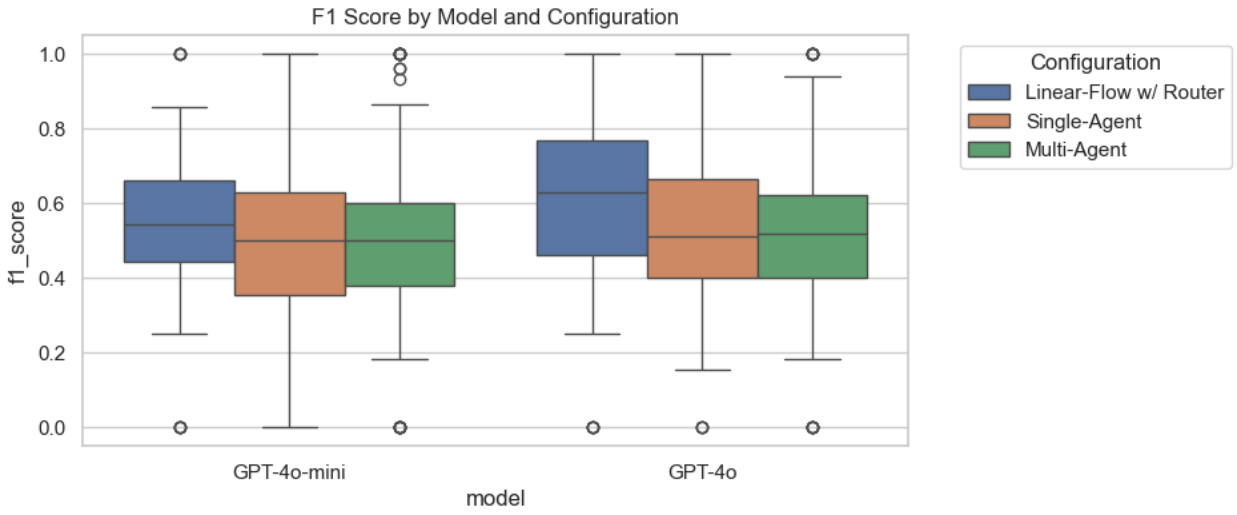


Figura 4.8: F1 Score distribution by model and configuration of agents

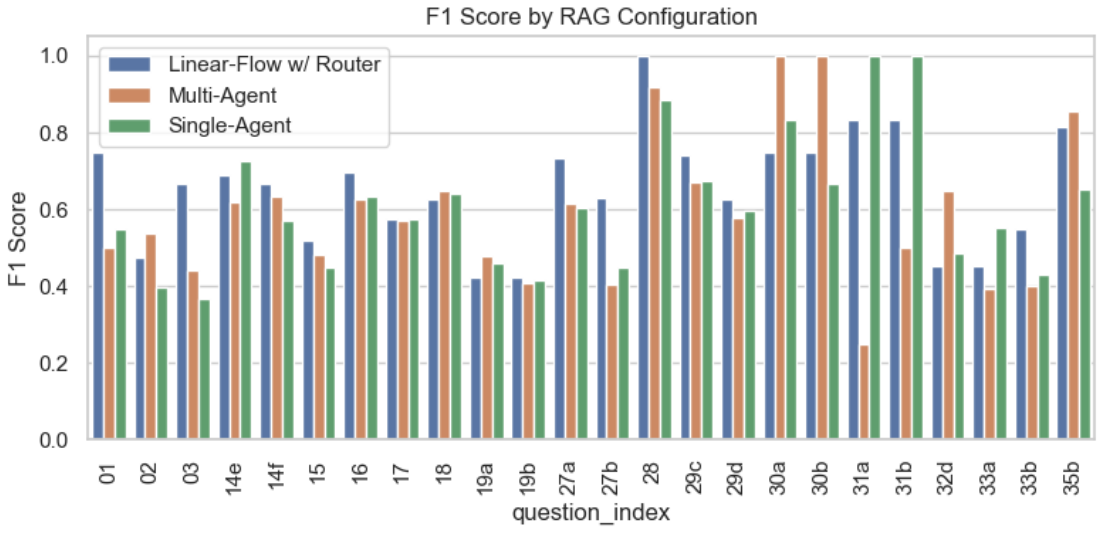


Figura 4.9: Enter Caption

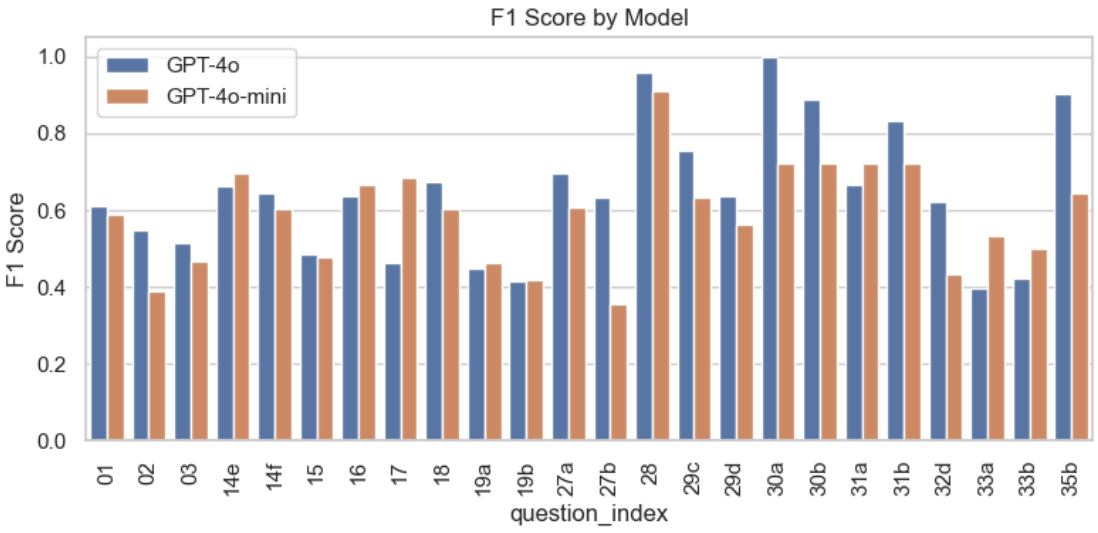


Figura 4.10: Enter Caption

Precisão

[GERAR TEXTO AQUI]

...

...

...

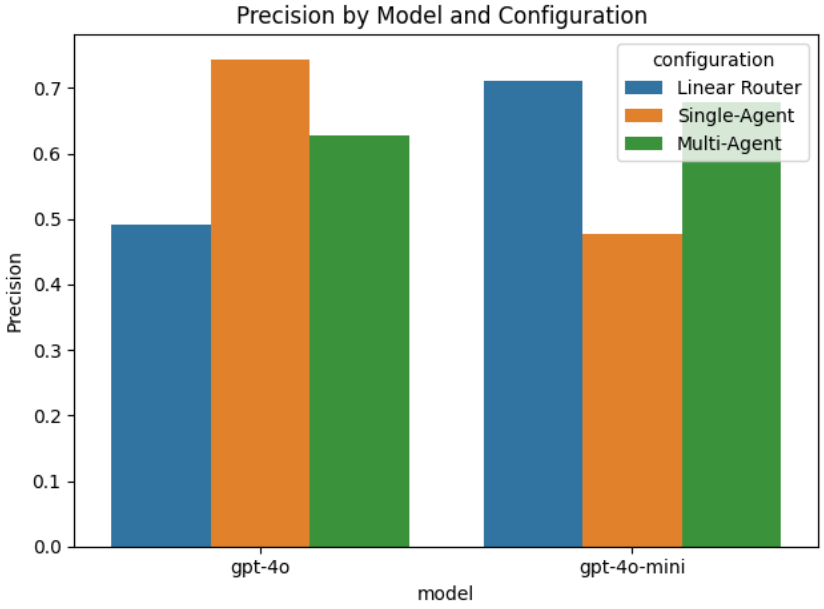


Figura 4.11: Precisão por modelo e configuração.

[GERAR TEXTO AQUI]

...

...

...

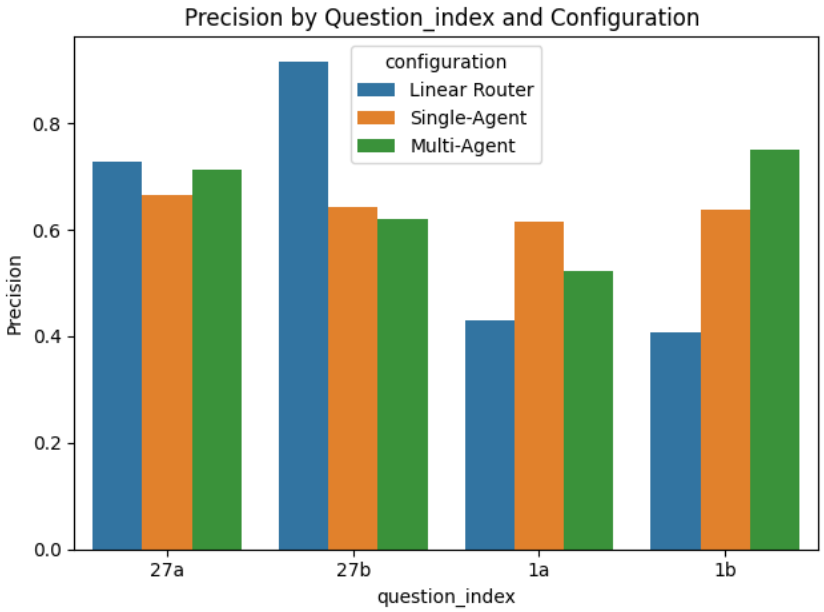


Figura 4.12: Precisão por pergunta e configuração.

[GERAR TEXTO AQUI]

...

...

Recall

[GERAR TEXTO AQUI]

...

...

[GERAR TEXTO AQUI]

...

...

...

[GERAR TEXTO AQUI]

...

...

4.3.2 Linear-Flow

In this setup, the user's query is handled by a single LLM step, which carries all the instructions (PT1, PT2, PT3 and PT4, as depicted in 4.1) required for the generation of various types of search queries. These instruction prompts are often

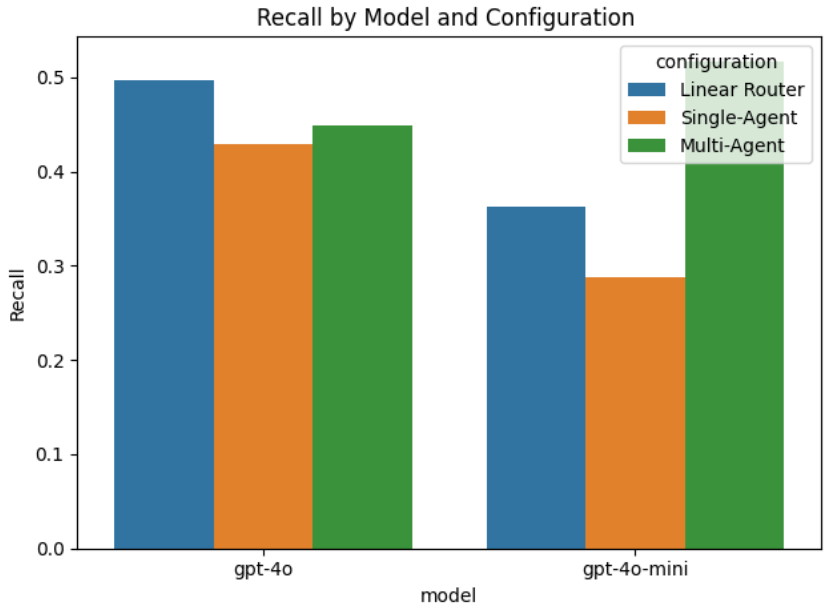


Figura 4.13: Recall por modelo e configuração.

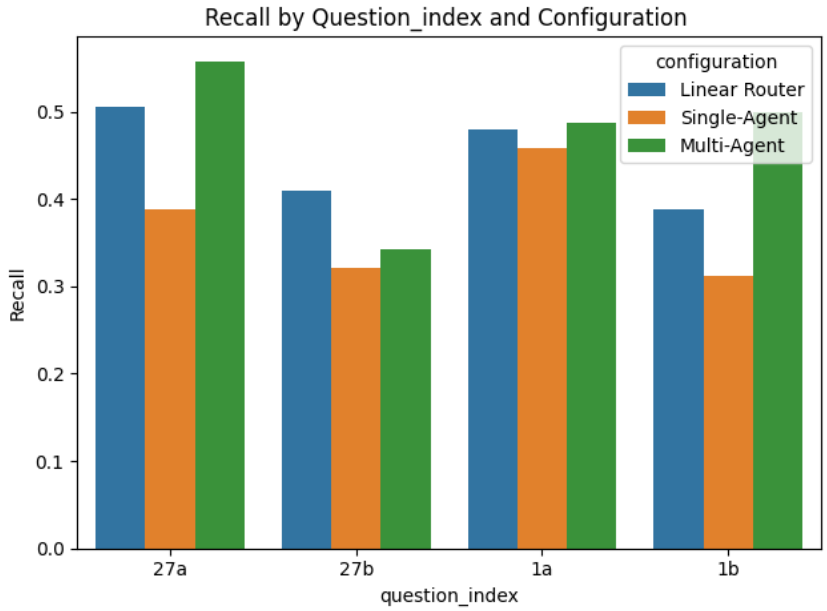


Figura 4.14: Recall por pergunta e configuração.

quite long, as they are carefully crafted to produce high-quality queries for the vector store. Due to the aggregation of all instruction prompts within a single LLM invocation, the resulting context becomes notably extensive. This can lead to performance degradation as the context length increases [O QUE PODE SER VISTO NO GRAFICO TAL EM CONTRASTE COM O SETUP TAL QUE DIVIDE OS PROMPTS EM PARTES].

...

...
...

4.3.3 Linear-Flow with Router

[GERAR TEXTO AQUI]

...
...
...

4.3.4 Single-Agent

[GERAR TEXTO AQUI]

...
...
...

4.3.5 Multi-Agent

[GERAR TEXTO AQUI]

...
...
...

Capítulo 5

Conclusões 1

Os resultados deste estudo destacam o potencial das arquiteturas multiagente baseadas em LLMs no setor de O&G, especialmente no domínio da engenharia de poços. A capacidade de processar e responder a consultas complexas abre caminho para uma transformação digital significativa na área.

Nossa análise comparativa de arquiteturas de agente único e multiagente, utilizando GPT-3.5-turbo e GPT-4, revela um panorama detalhado de trade-offs entre desempenho e eficiência econômica. Os sistemas multiagente demonstram uma veracidade 28% maior em tarefas de perguntas e respostas (Q&A), especialmente com GPT-4, em comparação com sistemas de agente único. No entanto, eles incorrem em custos de LLM que são, em média, 3,7 vezes maiores devido às complexidades da comunicação entre agentes. Em contraste, os sistemas de agente único se destacam em tarefas de Text-to-SQL, apresentando um desempenho 15% melhor do que as configurações multiagente. Essa dinâmica de custo-benefício exige uma consideração cuidadosa ao implementar RAG em cenários do mundo real, onde precisão e restrições financeiras devem ser equilibradas.

Destacamos vários desafios encontrados durante nossos experimentos, incluindo questões de contextualização, necessidade de filtragem de informações mais refinada e a persistência de alucinações. Esses desafios sublinham a necessidade de pesquisas contínuas em áreas como modelos especializados em domínios específicos, técnicas avançadas de busca semântica e arquiteturas híbridas que combinem as forças dos sistemas de agente único e multiagente.

As implicações práticas deste estudo vão além do setor de O&G. Os insights alcançados aqui são aplicáveis a qualquer domínio intensivo em conhecimento que lide com grandes volumes de dados técnicos. Ao focar em aprimorar os mecanismos de recuperação, desenvolver LLMs específicos de domínio e otimizar as interações entre agentes e ferramentas, pavimentamos o caminho para soluções RAG mais eficazes, confiáveis e econômicas em diversos setores.

Os principais pontos do estudo são os seguintes: sistemas multiagente ofere-

cem superior veracidade em tarefas de Q&A, embora a um custo significativamente maior. Arquiteturas de agente único, por outro lado, se destacam em tarefas de Text-to-SQL. Apesar das vantagens, persistem vários desafios, incluindo questões de contextualização, filtragem, alucinação e vocabulário específico de domínio.

Pesquisas futuras devem focar no desenvolvimento de modelos especializados, no avanço das técnicas de recuperação e na exploração de arquiteturas híbridas. As lições aprendidas deste estudo têm implicações mais amplas e podem se estender a outros domínios técnicos complexos. Ao abordar as limitações identificadas neste estudo e abraçar as tendências emergentes em sistemas multiagente e tecnologia RAG, podemos desbloquear seu potencial total, revolucionando a tomada de decisões, a gestão do conhecimento e a eficiência operacional em indústrias complexas em todo o mundo.

Capítulo 6

Conclusões 2 AAA

...

Referências Bibliográficas

WU, Q., BANSAL, G., ZHANG, J., et al. "AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation", 2023. doi: 10.48550/arXiv.2308.08155.

KAR, A. K., VARSHA, P. S. "Unravelling the Impact of Generative Artificial Intelligence (GAI) in Industrial Applications: A Review of Scientific and Grey Literature", *Global Journal of Flexible Systems Management*, v. 24, pp. 659–689, 12 2023. ISSN: 09740198. doi: 10.1007/s40171-023-00356-x.

ECKROTH, J., GIPSON, M. "Answering Natural Language Questions with OpenAI's GPT in the Petroleum Industry", pp. 16–18, 2023. doi: 10.2118/214888-MS. Disponível em: <http://onepetro.org/SPEATCE/proceedings-pdf/23ATCE/3-23ATCE/D031S032R005/3301837/spe-214888-ms.pdf/1>.

DELLACQUA, F., SARAN, A., MCFOWLAND, R. E., et al. "Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality", *Harvard Business School: Technology and Operations Management Unit Working Paper Series*, 2023. doi: 10.2139/ssrn.4573321. Disponível em: <https://ssrn.com/abstract=4573321>.

HATZIUS, J., BRIGGS, J., KODNANI, D., et al. "The Potentially Large Effects of Artificial Intelligence on Economic Growth". 2023.

SINGH, A., JIA, T., NALAGATLA, V. "Generative AI Enabled Conversational Chatbot for Drilling and Production Analytics". SPE, 10 2023. doi: 10.2118/216267-MS. Disponível em: <https://onepetro.org/SPEADIP/proceedings/23ADIP/2-23ADIP/D021S065R002/534485>.

HADI, M. U., QASEM AL TASHI, QURESHI, R., et al. "A Survey on Large Language Models: Applications, Challenges, Limitations, and Practical Usage", 2023. doi: 10.36227/techrxiv.23589741.v1. Disponível em: <https://doi.org/10.36227/techrxiv.23589741.v1>.

BADIRU, A. B., OSISANYA, S. O. *Project Management for the Oil and Gas Industry: A World System Approach*. Boca Raton, FL 33487-2742, CRC Press, 1 2016. ISBN: 9781420094268. doi: 10.1201/b13755.

THOMAS, J. E. *Fundamentos de Engenharia de Petróleo*. 2nd ed. Av. Presidente Vargas 435, Rio de Janeiro, RJ - 20.077-900, Editora Interciência, 2004.

BRAVO, C., SAPUTELLI, L., RIVAS, F., et al. "State of the art of artificial intelligence and predictive analytics in the E&P industry: A technology survey", *SPE Journal*, v. 19, n. 4, pp. 547–563, 2014. ISSN: 1086055X. doi: 10.2118/150314-pa. Disponível em: <http://onepetro.org/SJ/article-pdf/19/04/547/2099035/spe-150314-pa.pdf/1>.

GUDALA, M., NAIYA, T. K., GOVINDARAJAN, S. K. "Remediation of heavy oil transportation problems via pipelines using biodegradable additives: An experimental and artificial intelligence approach", *SPE Journal*, v. 26, n. 2, pp. 1050–1071, apr 2021. ISSN: 1086055X. doi: 10.2118/203824-PA.

KHAN, A. M. "Digital Integration Scope in Fracturing: Leveraging Domain Knowledge for Intelligent Advisors-Part I". International Petroleum Technology Conference (IPTC), 2024. ISBN: 9781959025184. doi: 10.2523/IPTC-24228-MS.

GOHARI, M. S. J., NIRI, M. E., SADEGHNEJAD, S., et al. "Synthetic Graphic Well Log Generation Using an Enhanced Deep Learning Workflow: Imbalanced Multiclass Data, Sample Size, and Scalability Challenges", *SPE Journal*, v. 29, pp. 1–20, 2024. ISSN: 1086055X. doi: 10.2118/217466-PA. Disponível em: <http://onepetro.org/SJ/article-pdf/29/01/1/3358626/spe-217466-pa.pdf/1>.

RAHMANI, A. M., AZHIR, E., ALI, S., et al. "Artificial intelligence approaches and mechanisms for big data analytics: a systematic study", *PeerJ Computer Science*, v. 7, pp. 1–28, 4 2021. ISSN: 23765992. doi: 10.7717/peerj-cs.488.

LIDDY, E. *Natural Language Processing*. Encyclopedia of Library and Information Science, 2001. Disponível em: <https://surface.syr.edu/istpub>.

ANTONIAK, M., DALGLIESH, J., VERKRUYSE, M., et al. "Natural language processing techniques on oil and gas drilling data", *Society of Petroleum Engineers - SPE Intelligent Energy International Conference and Exhibition*, 2016. doi: 10.2118/181015-MS.

CASTIÑEIRA, D., TORONYI, R., SALERI, N. "Machine Learning and Natural Language Processing for Automated Analysis of Drilling and Completion Data", pp. 23–26, 2018. doi: 10.2118/192280-MS. Disponível em: <http://onepetro.org/SPESATS/proceedings-pdf/18SATS/All-18SATS/SPE-192280-MS/1246545/spe-192280-ms.pdf/1>.

VASWANI, A., BRAIN, G., SHAZEER, N., et al. "Attention Is All You Need". 2017. Disponível em: <https://arxiv.org/abs/1706.03762>.

OPENAI, ACHIAM, J., ADLER, S., et al. "GPT-4 Technical Report", v. 4, pp. 1–100, 2023. doi: 10.48550/arXiv.2303.08774. Disponível em: <http://arxiv.org/abs/2303.08774>.

MOSSER, L., AURSAND, P., BRAKSTAD, K. S., et al. "Exploration Robot Chat: Uncovering Decades of Exploration Knowledge and Data with Conversational Large Language Models". In: *Day 1 Wed, April 17, 2024*. SPE, apr 2024. doi: 10.2118/218439-MS. Disponível em: <https://onepetro.org/SPEBERG/proceedings/24BERG/1-24BERG/D011S002R006/544177>.

ISKE, P., BOERSMA, W. "Connected brains. Question and answer systems for knowledge sharing: Concepts, implementation and return on investment". 2005. ISSN: 13673270.

TREUDE, C., BARZILAY, O., STOREY, M.-A. "How do programmers ask and answer questions on the web? (NIER track)". In: *Proceedings of the 33rd International Conference on Software Engineering*, ICSE '11, p. 804–807, New York, NY, USA, 2011. Association for Computing Machinery. ISBN: 9781450304450. doi: 10.1145/1985793.1985907. Disponível em: <https://doi.org/10.1145/1985793.1985907>.

QIN, B., HUI, B., WANG, L., et al. "A Survey on Text-to-SQL Parsing: Concepts, Methods, and Future Directions", 8 2022. doi: 10.48550/arXiv.2208.13629. Disponível em: <http://arxiv.org/abs/2208.13629>.

DENG, X., AWADALLAH, A. H., MEEK, C., et al. "Structure-Grounded Pretraining for Text-to-SQL", pp. 1337–1350, 2021. doi: 10.18653/v1/2021.naacl-main.105. Disponível em: <http://dx.doi.org/10.18653/v1/2021.naacl-main.105>.

DENG, X., GU, Y., ZHENG, B., et al. "MIND2WEB: Towards a Generalist Agent for the Web", 2023. Disponível em: <https://osu-nlp-group.github.io/Mind2Web>.

XI, Z., CHEN, W., GUO, X., et al. "The Rise and Potential of Large Language Model Based Agents: A Survey". 2023.

LI, J., ZHANG, Q., YU, Y., et al. "More Agents Is All You Need", 2024. doi: 10.48550/arXiv.2402.05120.

RUSSELL, S. *Artificial intelligence: a modern approach*. Pearson, 2020. ISBN: 0134610997.

LI, C., WANG, J., ZHANG, Y., et al. "Large Language Models Understand and Can Be Enhanced by Emotional Stimuli", 2023. doi: 10.48550/arXiv.2307.11760.

CARRARO, D. "Enhancing Recommendation Diversity by Re-ranking with Large Language Models", 2024. doi: 10.48550/arXiv.2401.11506.

SUN, W., YAN, L., MA, X., et al. "Is ChatGPT Good at Search? Investigating Large Language Models as Re-Ranking Agents", 2023. doi: 10.48550/arXiv.2304.09542. Disponível em: <www.github.com/sunnweiwei/RankGPT.>.

BILBAO, D., GELBUKH, A., RODRIGO, A., et al. "A Mathematical Investigation of Hallucination and Creativity in GPT Models", *Mathematics 2023*, v. 11, pp. 2320, 5 2023. ISSN: 2227-7390. doi: 10.3390/MATH11102320. Disponível em: <https://www.mdpi.com/2227-7390/11/10/2320/htmhttps://www.mdpi.com/2227-7390/11/10/2320>.

SHAH, B. "Large Learning Models: The Rising Demand of Specialized LLM's". 2024. Disponível em: <https://blogs.infosys.com/emerging-technology-solutions/artificial-intelligence/large-learning-models-the-rising-demand-of-specialized-llms.html>.

MEENA, S. "The Future of Large Language Models: Evolution, Specialization, and Market Dynamics". 2023. Disponível em: <https://www.linkedin.com/pulse/future-large-language-models-evolution-specialization-shekhar-meena/>.

GHOSH, B. "Emerging Trends in LLM Architecture,". 2023. Disponível em: <https://medium.com/@bijit211987/emerging-trends-in-llm-architecture-a8897d9d987b>.

LANGCHAIN. "Self-query Retriever". 2023. Disponível em: <https://python.langchain.com/docs/modules/data_connection/retrievers/self_query/>.

LEVENSHTEIN, V. "Binary codes capable of correcting deletions, insertions, and reversals", *Cybernetics and Control Theory*, v. 10, 1966.

SHINN, N., CASSANO, F., BERMAN, E., et al. "Reflexion: Language Agents with Verbal Reinforcement Learning", 3 2023. Disponível em: <http://arxiv.org/abs/2303.11366>.

PAL, S., BHATTACHARYA, M., LEE, S. S., et al. "A Domain-Specific Next-Generation Large Language Model (LLM) or ChatGPT is Required for Biomedical Engineering and Research". 3 2024. ISSN: 15739686.

AREFEEN, A., DEBNATH, B., CHAKRADHAR, S. "LeanContext: Cost-efficient domain-specific question answering using LLMs", *Natural Language Processing Journal*, v. 7, pp. 100065, 2024. doi: 10.1016/j.nlp.2024.100065. Disponível em: <https://doi.org/10.1016/j.nlp.2024.100065>.

Apêndice A

Um apêndice

Segundo a norma da ABNT (Associação Brasileira de Normas Técnicas), a definição e utilização de apêndices e anexos seguem critérios específicos para a organização de documentos acadêmicos e técnicos.

Apêndice: O apêndice é um texto ou documento elaborado pelo autor do trabalho com o objetivo de complementar sua argumentação, sem que seja essencial para a compreensão do conteúdo principal do documento. O uso de apêndices é indicado para incluir dados detalhados como questionários, modelos de formulários utilizados na pesquisa, descrições extensas de métodos ou técnicas, entre outros. Os apêndices são identificados por letras maiúsculas consecutivas, travessão e pelos respectivos títulos. A inclusão de apêndices visa a fornecer informações adicionais que possam ajudar na compreensão do estudo, mas cuja presença no texto principal poderia distrair ou desviar a atenção do leitor dos argumentos principais.

Anexo A

Um Anexo

Segundo a norma da ABNT (Associação Brasileira de Normas Técnicas), a definição e utilização de apêndices e anexos seguem critérios específicos para a organização de documentos acadêmicos e técnicos.

Anexo: O anexo, por sua vez, consiste em um texto ou documento não elaborado pelo autor, que serve de fundamentação, comprovação e ilustração. O uso de anexos é apropriado para materiais como cópias de artigos, legislação, documentos históricos, fotografias, mapas, entre outros, que tenham relevância para o entendimento do trabalho do autor. Assim como os apêndices, os anexos são identificados por letras maiúsculas consecutivas, travessão e pelos respectivos títulos. Eles são utilizados para enriquecer o trabalho com informações de suporte, garantindo que o leitor tenha acesso a documentos complementares importantes para a validação dos argumentos apresentados no texto principal.