
Desenvolvimento e Avaliação de Chatbot Pedagógico Open Source com RAG para Educação Online

Jonathan Martins Borges



UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE COMPUTAÇÃO
BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Monte Carmelo - MG
2025

Jonathan Martins Borges

**Desenvolvimento e Avaliação de Chatbot
Pedagógico Open Source com RAG para
Educação Online**

Trabalho de Conclusão de Curso apresentado
à Faculdade de Computação da Universidade
Federal de Uberlândia, Minas Gerais, como
requisito exigido parcial à obtenção do grau de
Bacharel em Sistemas de Informação.

Área de concentração: Sistemas de Informação

Orientador: Rafael Dias Araújo

Monte Carmelo - MG

2025

Dedico este trabalho àquele cujos olhos me viram a substância ainda informe, e no teu livro foram escritos todos os meus dias, cada um deles escrito e determinado, quando nem um deles havia ainda. E, de igual modo, dedico à minha preciosa família, cujo valor é inestimável: Cairo, Ana e Miguel, minhas fontes de inspiração, amor e esperança.

Agradecimentos

Agradeço grandemente ao singular Prof. Dr. Rafael Dias Araújo, pelo comprometimento, pelas lições acadêmicas e pela orientação, não só no trabalho vigente, mas também na publicação da pesquisa científica que apresentamos em Brasília. Foi uma experiência única e memorável que vivenciei como seu orientando. Sou grato, também, pelo exemplo de seriedade, sabedoria e humildade.

Agradeço ao PETSIMC que contribuiu para meu crescimento acadêmico e pessoal de diversas formas. Portanto, meu obrigado ao orientador Prof. Dr. Adriano Mendonça Rocha, e a todos os petianos.

Agradeço de forma pessoal aos amigos e colegas que estiveram comigo nesta odisseia universitária, com quem dividi momentos, a mesa, a sala de aula, o tempo e conversas. Especialmente, aos moradores da República Reinicia e seus agregados.

Devo agradecer aos trabalhadores dos Restaurantes Universitários (RU), cozinheiros, atendentes e demais colaboradores.

Agradeço em geral à Universidade Federal de Uberlândia, ao PETSIMC, à Faculdade de Computação (FACOM) e à FAPEMIG.

“e conhecereis a verdade, e a verdade vos libertará.”
(João 8:32)

Resumo

Este trabalho detalha uma metodologia abrangente para o desenvolvimento e avaliação de um chatbot pedagógico em plataformas educacionais, priorizando a acessibilidade e reprodutibilidade com ferramentas *open source*. O processo iniciou-se com a construção de um *dataset* detalhado do curso CS50, empregando técnicas de *web scraping* e processamento de texto para estruturar a base de conhecimento.

Com base neste *dataset*, geraram-se dados sintéticos de Perguntas e Respostas (QA) ricos e diversificados, formulados segundo os níveis cognitivos da Taxonomia de Bloom. A qualidade destes dados foi rigorosamente avaliada por meio de métricas automáticas do framework RAGAS, tais como *Faithfulness*, *Question Quality* (via *Simple Criteria Score*) e *Response Relevancy*. Esta abordagem foi complementada por supervisão humana, e a análise subsequente revelou correlações moderadas, mas estatisticamente significativas, entre as métricas automáticas e as avaliações humanas, validando a utilidade do RAGAS.

A etapa seguinte consistiu na avaliação do chatbot pedagógico, implementado com o software OpenWebUI e alimentado pela base de conhecimento do CS50. Compararam-se os modelos Gemini-2.0-Flash (proprietário) e GPT-OSS-20B (*open-weight*) na geração de respostas. Os resultados não apontaram diferença estatisticamente significativa na métrica *Factual Correctness* entre o modelo proprietário e o *open-weight*. Este achado reforça a viabilidade técnica de soluções open source para fornecer suporte pedagógico factualmente correto, mitigando custos e promovendo um cenário educacional mais inclusivo e equitativo.

Palavras-chave: Inteligência Artificial na Educação, LLMs, Chatbots Pedagógicos, RAG, Open Source.

Lista de ilustrações

Figura 1	– Fluxo de Dados do Pipeline de Preparação e Indexação para o Sistema Recuperação Aumentada por Geração (RAG)	28
Figura 2	– Aplicações de IA na Educação Online sugeridas no Brainstorming . . .	39
Figura 3	– Arquitetura Conceitual do Chatbot Pedagógico com RAG.	41
Figura 4	– Interface de interação do usuário no OpenWebUI , exibindo o chatbot com uma resposta gerada por LLM.	42
Figura 5	– Interface de administração do OpenWebUI , destacando as configurações de modelos de LLM, atribuição de acesso a grupos de usuários e associação a bases de conhecimento RAG.	43
Figura 6	– Visão Geral dos Componentes Tecnológicos Utilizados.	45
Figura 7	– Análise da distribuição dos documentos por tipo, tamanho em <i>tokens</i> e distribuição ao longo das semanas do curso.	48
Figura 8	– Fluxo do pipeline de geração de dados sintéticos de QA. O grafo de conhecimento (<i>Knowledge Graph</i>) é composto por nós de documentos e seus respectivos fragmentos (<i>chunks</i>), enriquecidos com propriedades como <i>headlines</i> e <i>keyphrases</i> . A partir desse grafo, são selecionados contextos e amostras semânticas para compor o <i>sampling pool</i> . O modelo de linguagem (<i>LLM</i>) recebe como entrada o contexto, parâmetros de estilo, comprimento e persona, gerando como saída pares de questão, contexto e resposta. Cada componente do diagrama representa uma etapa fundamental do processo de geração automática de conjuntos de dados para avaliação de sistemas de QA.	51
Figura 9	– Exemplo resumido do código Python e Prompts para definição de Personas	53
Figura 10	– Visualização de um grafo de conhecimento. Nota: A imagem apresenta uma versão ilustrativa e reduzida do grafo para facilitar a visualização.	56
Figura 11	– Tela do label Studio usado pelo especialista	62
Figura 12	– Cypress executando o experimento	68

Figura 13 – Estudo de caso visual para uma amostra do <i>dataset</i> de avaliação. A figura exibe: (a) a resposta de referência do gabarito e suas métricas de qualidade, (b) a resposta factualmente correta do GPT-OSS com RAG (<i>Factual Correctness</i> de 100%), e (c) a resposta do Gemini 2.5 Flash sem RAG, que demonstra o fenômeno da alucinação ao gerar informações incorretas e não relacionadas ao contexto solicitado.	75
Figura 14 – Análise de Correlação: RAGAS vs Avaliação Humana. Cada subplot mostra a relação entre métricas RAGAS (eixo-x) e humanas (eixo-y), incluindo linha de regressão (vermelha), correlação perfeita (cinza tracejada) e coloração baseada na distância da linha de regressão.	80
Figura 15 – Análise de Contingência e Matriz de Correlação. Esquerda: Análise de concordância entre exclusão humana e detecção RAGAS, mostrando distribuição de amostras e taxa de acordo. Direita: Matriz de correlação Spearman entre todas as métricas RAGAS e humanas.	81
Figura 16 – Distribuição das pontuações para Factual Correctness e Semantic Similarity por modelo, visualizada por meio de boxplots.	83
Figura 17 – Relação entre as pontuações de Factual Correctness e Semantic Similarity para os modelos Gemini e GPT.	84

Lista de tabelas

Tabela 1	–	Análise Comparativa de Plataformas Open-Source para Chatbots . . .	42
Tabela 2	–	Componentes Tecnológicos Utilizados e Sua Aplicação.	46
Tabela 3	–	Distribuição das Variáveis no Dataset de Avaliação Estratificado (N=120)	76
Tabela 4	–	Validação Estatística dos Dados de Avaliação	77
Tabela 5	–	Correlações Spearman entre Métricas RAGAS e Humanas	78
Tabela 6	–	Tabela de Contingência: Exclusão Humana vs Detecção RAGAS . . .	79
Tabela 7	–	Estatísticas Descritivas das Métricas de Avaliação por Modelo	83
Tabela 8	–	Links e descrições dos repositórios do projeto.	95

Lista de siglas

IA Inteligência Artificial

RAG Recuperação Aumentada por Geração - *Retrieval Augmented Generation*

API Interface de Programação de Aplicações - *Application Programming Interface*

LGPD Lei Geral de Proteção de Dados

LLM Modelo de Linguagem de Grande Escala - *Large Language Model*

SLM Modelo de Linguagem Pequeno - *Small Language Model*

CoT Cadeia de Pensamento - *Chain of Thought*

EP Engenharia de Prompts - *Prompt Engineering*

PNL Processamento de Linguagem Natural - *Natural Language Processing*

TCC Trabalho de Conclusão de Curso

GenAI Inteligência Artificial Generativa - *Generative Artificial Intelligence*

DPO Otimização por Preferência Direta - *Direct Preference Optimization*

ReAct Racióonio e Ação - *Reason and Act*

AGI Inteligência Artificial Geral - *Artificial General Intelligence*

ANN Vizinhos Mais Próximos Aproximados - *Approximate Nearest Neighbor*

QA Perguntas e Respostas - *Question Answering*

MoE Mistura de Especialistas - *Mixture-of-Experts*

JSON Notação de Objeto JavaScript - *JavaScript Object Notation*

JWT Token Web JSON - *JSON Web Token*

HTML Linguagem de Marcação de Hipertexto - *HyperText Markup Language*

RBAC Controle de Acesso Baseado em Função - *Role-Based Access Control*

Sumário

1	INTRODUÇÃO	14
1.1	Objetivos	16
1.1.1	Objetivos Específicos	16
1.2	Organização da Monografia	17
2	FUNDAMENTAÇÃO TEÓRICA	18
2.1	Modelos de Linguagem de Grande Escala (LLMs) e a Ascensão dos Modelos Pequenos (SLMs)	18
2.1.1	Avanços e Capacidades dos LLMs Proprietários de Larga Escala	18
2.1.2	O Paradigma Open Source em LLMs: Inovação, Transparência e Desafios	19
2.1.3	Modelos de Linguagem Pequenos (SLMs)	20
2.1.4	Implicações e Aplicações Educacionais dos LLMs e SLMs	21
2.1.5	Modelos de IA Utilizados no Estudo	21
2.2	Engenharia de Prompts	22
2.2.1	Técnicas de Engenharia de Prompts	23
2.3	Engenharia de Prompts para Interação com LLMs em Contextos Pedagógicos	23
2.4	Feedback, Avaliação e Aprendizagem	26
2.5	Considerações Éticas	27
2.6	Recuperação Aumentada por Geração (RAG) para Contextualização em Chatbots Educacionais	27
2.6.1	Funcionamento Detalhado e Componentes Chave do RAG	28
2.6.2	Chatbots Educacionais e sua Relevância	29
2.6.3	RAG versus Fine-tuning para Incorporação de Conhecimento Factual .	30
2.7	Trabalhos Relacionados	31
2.7.1	Panorama dos Chatbots na Educação e a Emergência do RAG	31
2.7.2	Chatbots Educacionais Baseados em Retrieval-Augmented Generation (RAG)	32

2.7.3	Outras Aplicações e Abordagens de IA Generativa na Educação	34
2.7.4	Considerações Técnicas, Infraestrutura Experimental e Modelos de Lin- guagem Menores	36
3	METODOLOGIA	38
3.1	Brainstorming e Levantamento de Requisitos	38
3.2	Ambiente de Implementação e Ferramentas Tecnológicas	40
3.2.1	Arquitetura Conceitual do Sistema	40
3.2.2	Ambiente de Desenvolvimento e Infraestrutura de Experimento	40
3.2.3	Componentes Tecnológicos do Sistema	41
3.2.4	Mapa Geral dos Componentes Tecnológicos Usados	44
3.3	Coleta e Preparação do Dataset CS50	46
3.3.1	Justificativa da Escolha da Fonte de Dados (CS50)	46
3.3.2	Processo de Coleta e Pré-processamento dos Dados	47
3.4	Geração de Dados Sintéticos de Perguntas e Respostas (QA)	49
3.4.1	Fundamentação Metodológica para Geração de QA	50
3.4.2	Pipeline de Geração de Dados Sintéticos de QA	50
3.4.3	Estratégias de Diversidade e Amostragem	53
3.4.4	Validação da Qualidade dos Dados Sintéticos de QA	55
3.4.5	Modelagem do Conhecimento via Grafos	55
3.4.6	Síntese de Perguntas e Respostas	56
3.4.7	Reprodutibilidade e Flexibilidade	57
3.4.8	Considerações de Qualidade e Diversidade	57
3.4.9	Importância para Avaliação de Sistemas RAG	57
3.5	Metodologia de Avaliação de Dados Sintéticos de QA	58
3.5.1	Métrica de Fidedignidade (Faithfulness)	58
3.5.2	Métrica de Critério Simples (Simple Criteria Score)	58
3.5.3	Métrica de Relevância da Resposta (Response Relevancy)	60
3.5.4	Procedimento do Experimento	61
3.5.5	Reprodutibilidade	61
3.5.6	Definição dos Critérios de Avaliação Humana	61
3.5.7	Detalhamento Formal das Métricas Utilizadas	63
3.6	Implementação e Avaliação do Chatbot Pedagógico com RAG	64
3.6.1	Configuração do Chatbot com RAG no OpenWebUI	64
3.6.2	Procedimento de Experimento Automatizado	68
3.6.3	Metodologia de Avaliação Automática do RAG com Ragas	69
3.6.4	Considerações Éticas	71
3.6.5	Escopo Ético e Público-Alvo do Sistema	71
3.7	Considerações Éticas	72
3.7.1	Privacidade e Soberania de Dados	72

3.7.2	Transparência e Reprodutibilidade	72
3.7.3	Mitigação do Risco de Dependência: Design e Governança	73
3.7.4	Vieses em Modelos <i>Open Source</i> e Curadoria Pedagógica	73
4	RESULTADOS E DISCUSSÃO	74
4.1	Análise Qualitativa de Exemplo: A Importância do RAG	74
4.2	Caracterização do Dataset de Avaliação	74
4.3	Processamento e Análise de Dados de Avaliação Humana	76
4.3.1	Metodologia de Processamento	76
4.3.2	Resultados Estatísticos	77
4.3.3	Análise de Correlação	78
4.3.4	Análise de Contingência	79
4.3.5	Visualizações e Análise Gráfica	79
4.3.6	Análise de Amostras com Baixa Correlação	80
4.3.7	Implicações Científicas	81
4.3.8	Conclusões	82
4.4	Resultados da Avaliação do Chatbot	82
4.4.1	Análise Estatística Descritiva e Visualização	83
4.4.2	Análise Estatística Inferencial	84
4.4.3	Discussão dos Resultados	85
5	CONCLUSÃO	86
5.1	Principais Contribuições	86
5.2	Trabalhos Futuros	87
	REFERÊNCIAS	89
	 APÊNDICES	 94
	APÊNDICE A – REPOSITÓRIOS DA PESQUISA	95

Introdução

A Inteligência Artificial (IA) tem avançado de maneira acelerada, impactando diversas áreas do conhecimento, especialmente a educação. Como afirmado, "Os homens agem sobre o mundo, modificam-no e são modificados pelas consequências de suas ações" (SKINNER, 1957). Este princípio é particularmente pertinente no contexto de tecnologias atuais como o GPT-4, um Modelo de Linguagem de Grande Escala (LLM) (OPENAI, 2023), esses modelos oferecem novas possibilidades para apoiar e transformar os processos educacionais. Essas ferramentas beneficiam tanto professores quanto alunos, uma vez que são fundamentadas em arquiteturas de *Transformer*, como o BERT, projetadas para compreender e gerar linguagem natural (DEVLIN et al., 2019). Isso abre um leque de aplicações inovadoras no ambiente educacional, incluindo sistemas capazes de fornecer suporte e *feedback* interativo.

Uma pesquisa global recente, conduzida pelo Digital Education Council (2024), revelou que 86% dos 3.839 estudantes de 16 países já utilizam IA em suas atividades acadêmicas, empregando ferramentas para pesquisa de informações, verificação gramatical e auxílio na escrita. No entanto, a pesquisa também destacou que mais de 50% dos alunos expressaram preocupações sobre a possibilidade de se tornarem excessivamente dependentes dessas tecnologias. Esse cenário ressalta a necessidade de integrar a IA na educação de forma responsável, garantindo que as ferramentas atuem como um complemento ao processo de aprendizagem, e não como um substituto que possa comprometer o desenvolvimento de habilidades essenciais, como a escrita ou o raciocínio crítico (MEYER et al., 2024).

As preocupações dos estudantes indicam que, embora reconheçam o potencial da IA, muitos buscam um equilíbrio entre o apoio que essas ferramentas oferecem e o desenvolvimento de suas próprias habilidades, evitando uma dependência excessiva da tecnologia. Portanto, é crucial orientar os alunos a utilizarem a IA como um suporte ao aprendizado, como uma ferramenta para obter *feedback* formativo e esclarecer dúvidas, e não como um substituto para o esforço individual e o desenvolvimento de competências essenciais.

A proficiência em diversas competências, incluindo a escrita, é vital para o êxito acadêmico e profissional, e o *feedback* eficaz desempenha um papel fundamental em seu

desenvolvimento (MEYER et al., 2024). Contudo, fornecer *feedback* individualizado, contextualizado e de alta qualidade para as diversas tarefas e dúvidas dos alunos representa um desafio significativo para os educadores, devido à complexidade dessas interações e às limitações de tempo (MEYER et al., 2024). Com a crescente demanda por uma aprendizagem personalizada e adaptativa em plataformas online, que requer atenção às necessidades individuais de cada aluno, esses desafios se tornam ainda mais intensos (FALCÃO et al., 2023).

A integração de tecnologias digitais na educação abre novas oportunidades para aprimorar o processo de *feedback* e suporte. À medida que essas tecnologias se tornam mais presentes no ensino e na aprendizagem, surgem possibilidades para o desenvolvimento de ferramentas de software que podem fornecer *feedback* imediato e interativo (FALCÃO et al., 2023). Nesse contexto, os LLMs emergem como ferramentas promissoras para automatizar e personalizar o suporte e o *feedback* em larga escala (MAITY; DERROY, 2024). Com sua capacidade de compreender dúvidas, analisar respostas (como as de exercícios) e gerar texto com características próximas à linguagem humana, esses modelos têm o potencial de transformar a forma como os alunos recebem orientação e *feedback* sobre seu aprendizado (MAITY; DERROY, 2024). Uma I.A, por exemplo, pode atuar como um tutor disponível a qualquer momento, respondendo perguntas sobre o conteúdo, auxiliando na resolução de problemas e fornecendo *feedback* sobre as tentativas do aluno, além de que os LLMs podem aliviar a sobrecarga dos educadores.

Entretanto, a implementação de LLMs para fins educacionais, especialmente através de soluções proprietárias, enfrenta barreiras significativas. Custos elevados de Interface de Programação de Aplicações (API), dependência de plataformas onerosas, falta de transparência e controle sobre os modelos, além de questões de privacidade de dados, podem limitar sua adoção. Essas barreiras são particularmente desafiadoras para instituições educacionais com recursos financeiros limitados, como muitas no Brasil e em outros países emergentes, criando uma potencial divisão digital no acesso a ferramentas de IA de apoio.

Como alternativa, o ecossistema de IA de código aberto (*open source*) oferece um caminho promissor para a democratização dessas tecnologias. Modelos de linguagem, bibliotecas e ferramentas abertas permitem o desenvolvimento de soluções mais acessíveis, transparentes e adaptáveis. A utilização de LLMs *open source* mais leves (e.g., <10B parâmetros) torna viável a execução em hardware mais modesto, reduzindo custos e permitindo implementações locais que garantem maior controle sobre os dados e a tecnologia. Esta abordagem alinha-se com a necessidade de soluções sustentáveis e equitativas para a educação em contextos com restrições orçamentárias.

Apesar das vantagens do *open source*, a implementação da IA em contextos educacionais ainda demanda atenção a desafios cruciais. É essencial que os modelos, mesmo os abertos, sejam configurados e guiados (via engenharia de prompt) para atuar de forma

pedagogicamente sólida. Um chatbot educacional não deve apenas fornecer respostas, mas também estimular o raciocínio, guiar a descoberta e oferecer feedback construtivo que promova o desenvolvimento contínuo de competências (RODRIGUES et al., 2024). A qualidade e a precisão das interações, a gestão de possíveis vieses nos modelos e as implicações éticas do uso de IA no suporte ao aluno exigem análise e monitoramento cuidadosos (RODRIGUES et al., 2024). A proteção da privacidade dos dados e a conformidade com regulamentações como a Lei Geral de Proteção de Dados (LGPD) também são fundamentais.

1.1 Objetivos

O objetivo geral deste estudo é propor, desenvolver e avaliar uma arquitetura e metodologia baseada em ferramentas *open source* para a criação de chatbots pedagógicos com RAG. Busca-se demonstrar a viabilidade técnica e a eficácia pedagógica preliminar desta abordagem, através de implementação e testes em ambiente local controlável, visando sua potencial aplicação em plataformas educacionais online de forma acessível, replicável e democrática. Portanto, este trabalho detalha a construção e avaliação de um protótipo funcional de chatbot pedagógico RAG, capaz de fornecer suporte e feedback contextualizado, utilizando modelos e ferramentas *open source*, aplicado a diferentes contextos de cursos, e discute suas implicações para um cenário educacional mais inclusivo e equitativo.

1.1.1 Objetivos Específicos

Esta pesquisa propõe a execução dos seguintes objetivos específicos, que visam aprofundar e concretizar o objetivo geral da pesquisa:

1. Propor, desenvolver e documentar uma arquitetura de sistema para *chatbots* pedagógicos com RAG utilizando uma pilha tecnológica baseada em ferramentas e modelos de código aberto.
2. Desenvolver um *pipeline* metodológico robusto para a coleta, curadoria e preparação de um conjunto de dados educacionais, com foco no curso CS50 da Universidade de Harvard, garantindo a sua adequação e confiabilidade como base de conhecimento para o sistema RAG.
3. Implementar e otimizar a engenharia de *prompts* com fundamentos pedagógicos, adaptando-a especificamente para o LLM *open source* selecionado, com o intuito de guiar suas interações e respostas de forma a estimular o raciocínio crítico e a aprendizagem autodirigida.
4. Propor e validar uma metodologia híbrida para a criação de um dataset sintético de Perguntas e Respostas (QA), combinando geração automática baseada na Ta-

xonomia de Bloom com supervisão humana, visando estabelecer um referencial de avaliação confiável para sistemas RAG em contextos educacionais.

5. Realizar a avaliação do desempenho dos resultados do *chatbot* pedagógico em factua-
lidade e similaridade semântica com dois LLMs distintos (proprietário e opensource)
para fins de comparação.

1.2 Organização da Monografia

A presente monografia está estruturada em capítulos ordenados, organizados para apresentar de forma sequencial e lógica o desenvolvimento e as contribuições do estudo.

Capítulo 2: Fundamentação Teórica: descreve os conceitos essenciais da pesquisa com o devido arcabouço científico na literatura atual, abrangendo a evolução e as características dos LLMs e Modelo de Linguagem Pequenos (SLMs), a teoria da Recuperação Aumentada por Geração (RAG), a relevância do *feedback* e da avaliação na educação, e as implicações éticas no uso de sistemas de IA.

Capítulo 3: Metodologia: Detalha a arquitetura do sistema proposta, as ferramentas tecnológicas e os modelos *open source* selecionados, o processo de curadoria do conjunto de dados do curso CS50, e a metodologia de geração de dados sintéticos de perguntas e respostas para avaliação. Adicionalmente, apresenta a configuração experimental e os critérios de avaliação para a análise de desempenho do *chatbot*.

Capítulo 4: Resultados e Discussão: Inicialmente, são detalhados os resultados da geração e processamento dos dados sintéticos (QA). Esta parte aborda a avaliação da qualidade dos dados, comparando as métricas automáticas do RAGAS com a supervisão humana. Em seguida, são apresentados os resultados da avaliação comparativa entre o modelo de linguagem de código aberto (GPT-OSS) e o modelo proprietário (Gemini-2.0-Flash) usados no chatbot com RAG, analisando as métricas de fidedignidade e relevância semântica. A seção também discute as implicações desses resultados para a viabilidade e a eficácia de soluções de IA acessíveis, além de abordar as limitações identificadas no processo.

Por fim, o **Capítulo 5: Conclusão:** sintetiza as principais contribuições do trabalho, reafirmando o potencial de soluções *open source* para a democratização da IA na educação. O capítulo também discute as limitações do estudo e sugere direções para futuras pesquisas, incluindo a expansão da metodologia de avaliação e a exploração de novas técnicas e modelos.

Fundamentação Teórica

Esta seção apresenta a fundamentação teórica que embasa o desenvolvimento e a implementação do *chatbot* com RAG proposta, abordando LLMs, SLM, *feedback*, avaliação e aprendizagem, bem como metodologias e considerações éticas.

2.1 Modelos de Linguagem de Grande Escala (LLMs) e a Ascensão dos Modelos Pequenos (SLMs)

Os Modelos de Linguagem de Grande Escala (LLMs) constituem uma categoria avançada de sistemas de IA, projetados para processar, compreender e gerar texto de forma coerente e contextualmente relevante. Estes modelos são tipicamente treinados em vastos corpus textuais utilizando arquiteturas de aprendizado profundo, como os Transformers (VASWANI et al., 2023), que lhes conferem notáveis capacidades em Processamento de Linguagem Natural (PNL) (DEVLIN et al., 2019; BROWN et al., 2020). Seu impacto tem sido significativo em diversas áreas, com aplicações particularmente promissoras na educação, onde habilitam a automação e o aprimoramento de tarefas como a geração de materiais didáticos, a criação de sistemas de perguntas e respostas, e o fornecimento de *feedback* individualizado (RODRIGUES et al., 2024).

2.1.1 Avanços e Capacidades dos LLMs Proprietários de Larga Escala

A evolução dos LLMs tem sido marcada por avanços significativos impulsionados por modelos proprietários de grande escala. Capacidades multimodais, por exemplo, expandiram a funcionalidade para além do texto, ao integrar o processamento de informações visuais e auditivas. Modelos como o GPT-4 da OpenAI (OPENAI, 2023) e o Gemini 1.5 do Google (Gemini Team, 2024), exemplificam essa tendência, demonstrando habilidade em interpretar e gerar conteúdo a partir de diversos tipos de dados sensoriais. Tais

capacidades são valiosas para a compreensão de materiais didáticos complexos e para a geração de explicações detalhadas em ambientes educacionais (YANG et al., 2023). O Gemini 1.5, por exemplo, destaca-se pela sua capacidade de processar extensos contextos (milhões de *tokens*), sendo eficiente em tarefas de recuperação de informação e Perguntas e Respostas (QA) - do inglês *Questions and Answers* - em domínios educacionais (Gemini Team, 2024). Esses modelos de fronteira, frequentemente validados em plataformas como o *LMSys Chatbot Arena* (CHIANG et al., 2024), demonstram o estado da arte em termos de performance, mas geralmente operam sob um paradigma de código fechado, com as implicações que serão discutidas posteriormente.

2.1.2 O Paradigma Open Source em LLMs: Inovação, Transparência e Desafios

Paralelamente ao desenvolvimento de modelos proprietários, um vibrante e crescente ecossistema de LLMs *open source* tem emergido, impulsionado pela filosofia de democratizar o acesso à tecnologia, promover a colaboração científica e permitir maior escrutínio e adaptação dos modelos (MANCHANDA et al., 2025). Modelos como a série LLaMA da Meta (TOUVRON et al., 2023) tornaram-se referências de arquitetura e desempenho para a comunidade aberta, inspirando uma onda de desenvolvimentos seguintes como o DeepSeek (DEEPSEEK-AI et al., 2024). Iniciativas como Mistral (JIANG et al., 2023) e Qwen (BAI et al., 2023) também contribuíram significativamente para o avanço e a diversificação dos LLMs abertos.

Este movimento *open source* é fundamental, pois, conforme argumentado por Manchanda et al. (MANCHANDA et al., 2025), ele representa o caminho mais robusto para o avanço da pesquisa e para a implementação ética dos LLMs. A disponibilização pública de arquiteturas, e em muitos casos de pesos e dados de treinamento (ou suas metodologias de curadoria), fomenta um ecossistema colaborativo. Este ambiente facilita o desenvolvimento iterativo, a validação externa rigorosa e a mitigação de vieses através do escrutínio coletivo, conforme destacado por Manchanda et al. (MANCHANDA et al., 2025), que consolidam argumentos sobre os benefícios da transparência algorítmica e do acesso a dados para o avanço da pesquisa colaborativa e a identificação de vieses. Além disso, a abordagem *open source* prioriza inovações adaptativas e focadas em eficiência, promovendo modularidade arquitetônica e acessibilidade computacional (MANCHANDA et al., 2025).

Contudo, o desenvolvimento de LLMs *open source* de alta performance, capazes de rivalizar com os modelos proprietários, não é trivial. Exige não apenas expertise em arquitetura e treinamento, mas também um entendimento profundo das leis de escalonamento (*scaling laws*) que regem a relação entre tamanho do modelo, volume de dados de treinamento e orçamento computacional. O projeto DeepSeek LLM, por exemplo, dedi-

cou um esforço considerável à investigação dessas leis de escalonamento para otimizar seus modelos abertos, reconhecendo que trabalhos anteriores apresentavam conclusões variadas e informações sobre hiperparâmetros (DEEPSEEK-AI et al., 2024). Eles destacam que a qualidade dos dados de pré-treinamento impacta significativamente a estratégia ótima de escalonamento, sugerindo que dados de maior qualidade podem justificar um maior investimento no aumento do tamanho do modelo para um dado orçamento computacional (DEEPSEEK-AI et al., 2024).

2.1.3 Modelos de Linguagem Pequenos (SLMs)

Dentro da trajetória de evolução dos modelos de linguagem, uma vertente de crescente importância é a dos Modelos de Linguagem Pequenos (SLMs). Conforme destacado por Lu et al. (LU et al., 2025), enquanto a pesquisa em LLMs de grande escala frequentemente visa os limites da Inteligência Artificial Geral (AGI), os SLMs buscam tornar a inteligência de máquina mais acessível, financeiramente viável e eficiente para tarefas cotidianas, especialmente em dispositivos com recursos limitados. A visão por trás dos SLMs é a de democratizar o acesso à inteligência artificial, tornando-a ubíqua e prática (LU et al., 2025). SLMs, tipicamente definidos na faixa de 100 milhões a 5 bilhões de parâmetros (LU et al., 2025), são cruciais para viabilizar a execução em hardware de consumidor, desktops padrão ou servidores locais, mitigando custos e preocupações com privacidade.

O projeto DeepSeek LLM, embora tenha desenvolvido modelos de até 67 bilhões de parâmetros, também contribui para o entendimento e avanço de modelos que, em comparação com os gigantes de centenas de bilhões ou trilhões de parâmetros, são mais acessíveis e são desenvolvidos sob a égide do *open source*. Seu modelo de 7 bilhões de parâmetros (DeepSeek LLM 7B) é um exemplo relevante nesta categoria. O trabalho da equipe DeepSeek demonstra um esforço sistemático para construir modelos *open source* “do zero” (*from scratch*), com foco em um extenso pré-treinamento (2 trilhões de tokens, bilíngue em inglês e chinês) e seguindo, em grande parte, a arquitetura eficiente da série LLaMA, mas com otimizações próprias, como um escalonador de taxa de aprendizado multi-etapas para facilitar o treinamento contínuo (DEEPSEEK-AI et al., 2024).

A existência e o sucesso de projetos como DeepSeek e outros, que investem na criação de modelos *open source* robustos e compartilham parte de sua metodologia, são essenciais para o avanço de LLMs e SLMs que podem ser efetivamente utilizados em aplicações como as propostas neste trabalho, onde a acessibilidade, o controle e a capacidade de execução em infraestrutura mais modesta são prioritários.

2.1.4 Implicações e Aplicações Educacionais dos LLMs e SLMs

A combinação da capacidade de adaptação dos LLMs com a acessibilidade crescente proporcionada pelos SLMs *open source* abre um vasto leque de aplicações educacionais. A escalabilidade para atender a um grande número de alunos em plataformas online é uma vantagem inerente (MAITY; DERROY, 2024). No contexto específico de chatbots pedagógicos, que é o cerne deste trabalho, tanto LLMs de porte moderado quanto SLMs podem ser empregados para criar assistentes virtuais capazes de:

- ❑ Fornecer explicações contextuais sobre o material do curso, utilizando RAG para garantir fidelidade.
- ❑ Auxiliar na resolução de dúvidas pontuais de forma interativa.
- ❑ Oferecer feedback formativo e imediato sobre as tentativas dos alunos em exercícios ou na compreensão de conceitos (MEYER et al., 2024).
- ❑ Estimular o engajamento e a aprendizagem ativa através de interações dialógicas, adaptadas ao nível do aluno.

A contínua evolução e diversificação dos LLMs, com um destaque especial para a ascensão dos SLMs *open source*, reforçam o potencial transformador dessas tecnologias. Elas pavimentam o caminho para experiências de aprendizado mais adaptativas, personalizadas e, crucialmente, acessíveis e equitativas, alinhando o avanço tecnológico com as necessidades práticas de diversos contextos educacionais.

2.1.5 Modelos de IA Utilizados no Estudo

A seleção dos modelos de linguagem e de *embedding* foi um componente crítico deste trabalho, orientada pela necessidade de avaliar diferentes arquiteturas e paradigmas de acesso (proprietário vs. *open source*). A seguir, detalham-se os modelos empregados.

2.1.5.1 Família Gemini (Google)

Os modelos da família Gemini, desenvolvidos pelo Google, são sistemas multimodais com arquitetura baseada em transformadores do tipo Mistura de Especialistas (MoE) *do ingles Mixture-of-Experts* (COMANICI et al., 2025). Essa arquitetura permite desacoplar a capacidade total do modelo do custo computacional por token, ativando apenas um subconjunto de parâmetros (experts) para cada entrada. Neste estudo, foram utilizados os modelos **gemini-2.0-flash** e **gemini-2.0-flash-lite**. Ambos são otimizados para tarefas que exigem baixa latência e alto volume de processamento, tornando-os adequados para a geração de dados sintéticos em larga escala e para aplicações interativas como chatbots. O modelo **gemini-2.0-flash**, em particular, foi utilizado como um dos agentes

no chatbot RAG para servir como um ponto de referência de desempenho de um modelo proprietário de última geração, além de ser utilizado também para geração de dados sintéticos.

2.1.5.2 GPT-OSS (OpenAI)

O `gpt-oss-20b` é um modelo de linguagem *open-weight* com 20.9 bilhões de parâmetros, disponibilizado pela OpenAI sob a licença Apache 2.0 (OPENAI et al., 2025). Sua arquitetura também é baseada em transformadores do tipo MoE, construída sobre os fundamentos das arquiteturas da família de LLMs GPTs da OpenAI. O modelo foi projetado com foco em capacidades de raciocínio, uso de ferramentas (*tool use*) e seguimento de instruções complexas, características desejáveis para um agente de chatbot pedagógico. De acordo com seu *model card*, o `gpt-oss-20b` demonstra desempenho competitivo em diversos *benchmarks* de raciocínio e codificação (OPENAI et al., 2025). No contexto deste trabalho, ele representa a vertente *open source* na avaliação comparativa do chatbot, permitindo analisar a viabilidade de uma solução baseada em modelos abertos, mesmo quando acessado via API.

2.1.5.3 Qwen3-Embedding (Alibaba Group)

Para a tarefa de representação vetorial do texto (*embedding*), essencial para o funcionamento do sistema RAG, foi selecionado o modelo `Qwen3-Embedding-0.6B`. Este modelo faz parte da série Qwen3 Embedding, desenvolvida pelo Alibaba Group, e é construído sobre os modelos de fundação Qwen3 (ZHANG et al., 2025). Com apenas 0.6 bilhões de parâmetros, o modelo é notavelmente eficiente, permitindo sua execução em hardware local com recursos limitados. Apesar de seu tamanho reduzido, o `Qwen3-Embedding` alcança resultados de estado da arte em diversos *benchmarks* de *embedding*, incluindo o MTEB (*Massive Text Embedding Benchmark*), superando modelos proprietários em tarefas de recuperação de código e recuperação multilíngue (ZHANG et al., 2025). Sua natureza *open source* e seu alto desempenho em relação ao custo computacional o tornaram a escolha ideal para o componente de *retrieval* deste projeto.

2.2 Engenharia de Prompts

A engenharia de *prompts* desempenha um papel crucial no uso de LLMs para tarefas educacionais. Essa técnica envolve a formulação cuidadosa de *prompts* que direcionam os LLMs a produzir respostas que não apenas atendam às necessidades pedagógicas específicas, mas também promovam a compreensão profunda e o desenvolvimento de habilidades cognitivas nos estudantes.

Para potencializar as respostas dos LLMs, empregam-se técnicas de engenharia de *prompts*, incluindo Cadeia de Pensamento (CoT), do inglês (*Chain of Thought*, CoT), *Progressive-Hint Prompting* (PHP) e o método socrático (ZHENG et al., 2023; CHANG, 2023), com o intuito de aprimorar o raciocínio dos modelos, especialmente em atividades que requerem explicações pedagógicas profundas e contextualizadas.

2.2.1 Técnicas de Engenharia de Prompts

A **Metodologia Socrática** é implementada como uma técnica de engenharia de *prompts* que promove um diálogo estruturado, incentivando o modelo a engajar-se em raciocínio mais profundo e educativo (ZHENG et al., 2023). Essa técnica utiliza uma série de perguntas que desafiam o modelo a explorar aspectos específicos do conteúdo, como os fundamentos teóricos e exemplos práticos, permitindo que o modelo forneça explicações detalhadas que facilitam o aprendizado. A aplicação do método socrático ajuda os estudantes a desenvolver uma compreensão mais crítica e aprofundada, à medida que o modelo guia o aprendizado através de perguntas reflexivas e fundamentadas.

A **Cadeia de Pensamento (CoT)** é uma técnica que organiza o *prompt* de modo a conduzir o modelo de IA a desenvolver uma linha de raciocínio passo a passo até chegar a uma conclusão ou resposta. Esse processo permite que o LLM articule seu raciocínio de maneira sequencial e detalhada, promovendo uma análise clara e estruturada do conteúdo educacional. Essa técnica é particularmente valiosa em cenários em que o aluno necessita de explicações que detalhem o processo de resolução de problemas, facilitando a transparência e a identificação de possíveis pontos de dúvida ou falhas de entendimento.

O ***Progressive-Hint Prompting (PHP)*** envolve fornecer ao modelo dicas progressivas que ajudam a guiar suas respostas em direção a uma compreensão mais precisa do conteúdo educacional. Com base nas respostas iniciais do modelo, são oferecidas dicas adicionais que aprimoram a resposta, incentivando o modelo a refinar suas explicações e a corrigir erros conceituais. Este método é complementar ao CoT e extremamente útil para problemas complexos ou quando é necessário ajustar e detalhar as respostas do modelo ao longo do processo, facilitando uma abordagem adaptativa que se ajusta ao nível de compreensão dos modelos (CHANG, 2023).

2.3 Engenharia de Prompts para Interação com LLMs em Contextos Pedagógicos

A interação eficaz com Modelos de Linguagem de Grande Escala (LLMs) depende fundamentalmente da qualidade das instruções fornecidas a eles. A Engenharia de Prompts (EP) do inglês *Prompt Engineering*) emergiu como uma disciplina crucial, focada no de-

sign estratégico e na otimização de *prompts* — as entradas textuais (ou multimodais) que guiam o comportamento e a saída dos LLMs (BOONSTRA, 2025; LO, 2023). Conforme destacado por Boonstra (BOONSTRA, 2025), a EP é um processo iterativo de ajuste fino para encontrar o melhor prompt, otimizando seu tamanho e avaliando seu estilo de escrita e estrutura em relação à tarefa. O objetivo é configurar o LLM para prever a sequência correta de tokens que leve a uma resposta precisa e útil, sem modificar os parâmetros internos do modelo (SAHOO et al., 2025).

A significância da EP é particularmente acentuada no campo educacional, onde a clareza, a precisão e a relevância pedagógica das respostas de um chatbot são primordiais. Uma instrução mal formulada pode levar a respostas ambíguas, incorretas ou pedagogicamente inadequadas, comprometendo o potencial da ferramenta como suporte ao aprendizado (BOONSTRA, 2025). O framework CLEAR (Concise, Logical, Explicit, Adaptive, Reflective) proposto por Lo (LO, 2023) oferece um guia para otimizar interações com LLMs, enfatizando princípios que são diretamente aplicáveis ao desenvolvimento de chatbots pedagógicos eficazes. A capacidade de direcionar as respostas do modelo através de prompts cuidadosamente elaborados é, portanto, essencial para o desenvolvimento de aplicações de IA educacionais que sejam verdadeiramente úteis e confiáveis.

Diversas técnicas de engenharia de prompt foram desenvolvidas e catalogadas, variando em complexidade e aplicação. Surveys sistemáticos como os de Sahoo et al. (SAHOO et al., 2025) e Schulhoff et al. (SCHULHOFF et al., 2025) oferecem taxonomias abrangentes dessas técnicas. Para o contexto de chatbots pedagógicos, algumas das abordagens mais relevantes incluem:

- ❑ **Prompting de Tiro Único (Zero-Shot Prompting):** Esta é a forma mais básica, onde o LLM recebe uma descrição da tarefa ou uma pergunta direta sem exemplos prévios de como responder (BOONSTRA, 2025; SAHOO et al., 2025). Por exemplo, um prompt como "Explique o conceito de fotossíntese para um aluno do ensino fundamental" depende inteiramente do conhecimento pré-treinado do modelo. Embora simples, a eficácia do zero-shot pode ser limitada para tarefas mais complexas ou que exigem um formato de resposta muito específico.
- ❑ **Prompting de Poucos Tiros (Few-Shot Prompting):** Para melhorar a previsibilidade e a qualidade da resposta, esta técnica fornece ao LLM um ou mais exemplos (demonstrações) de entrada-saída desejada antes da pergunta ou instrução real (BOONSTRA, 2025; SAHOO et al., 2025). Os exemplos ajudam o modelo a "aprender" o padrão ou o formato esperado, sendo particularmente útil para guiar o LLM a produzir saídas estruturadas ou a adotar um certo estilo (BOONSTRA, 2025). A seleção e a qualidade desses exemplos são críticas para o sucesso da técnica (SCHULHOFF et al., 2025).

- ❑ **Prompting de Cadeia de Pensamento (Chain-of-Thought - CoT):** Introduzida para aprimorar as capacidades de raciocínio dos LLMs, a CoT envolve instruir o modelo a “pensar passo a passo” ou a externalizar sua linha de raciocínio antes de fornecer a resposta final (BOONSTRA, 2025; SAHOO et al., 2025). Esta técnica tem se mostrado eficaz em tarefas que exigem raciocínio lógico, matemático ou de senso comum, pois decompõe problemas complexos em etapas intermediárias gerenciáveis (SAHOO et al., 2025). Schulhoff et al. (SCHULHOFF et al., 2025) também destacam a CoT como uma técnica fundamental de geração de pensamento.
- ❑ **Prompting de Raciocínio e Ação (ReAct):** Indo além do raciocínio puramente interno, a técnica ReAct propõe uma sinergia entre raciocínio e ação, onde o LLM é guiado a gerar tanto traços de raciocínio (pensamentos) quanto ações específicas de tarefas de forma intercalada (YAO et al., 2023). Por meio de prompts estruturados que exemplificam esse ciclo de "Thought -> Action -> Observation", o LLM aprende a decompor tarefas, formular planos, interagir com ferramentas externas (como APIs de busca para coletar informações) e ajustar seu raciocínio com base nas observações recebidas. Esta abordagem permite que o LLM supere limitações de conhecimento estático e resolva problemas que requerem informações atualizadas ou interações com o ambiente, demonstrando melhorias significativas em tarefas de resposta a perguntas e verificação de fatos ao interagir com bases de conhecimento externas (YAO et al., 2023). O ReAct é um exemplo poderoso de como a engenharia de prompt pode orquestrar fluxos complexos de processamento, transformando o LLM em um agente mais autônomo e capaz.
- ❑ **Prompting Baseado em Persona/Papel (Role Prompting):** Nesta abordagem, atribui-se um papel ou persona específica ao LLM (e.g., “Você é um tutor de física paciente e encorajador”). Isso ajuda a moldar o tom, o estilo e o nível de detalhe da resposta, tornando-a mais adequada ao contexto pedagógico desejado (BOONSTRA, 2025). A definição de uma persona pode melhorar significativamente a qualidade, relevância e eficácia da interação (SCHULHOFF et al., 2025).
- ❑ **Prompting Contextual e de Sistema (System and Contextual Prompting):** O prompting de sistema define o contexto geral e o propósito do LLM (e.g., “Você é um chatbot de apoio ao curso CS50”), enquanto o prompting contextual fornece detalhes específicos ou informações de background relevantes para a conversa ou tarefa atual (BOONSTRA, 2025). Ambos são cruciais para manter a coerência e a relevância das respostas do chatbot.
- ❑ **Técnicas para Reduzir Alucinações e Melhorar a Fidelidade:** Dada a importância da precisão factual na educação, técnicas que mitigam alucinações são essenciais. O artigo de Sahoo et al. (SAHOO et al., 2025) discute abordagens

como a Geração Aumentada por Recuperação (RAG), ReAct e Chain-of-Verification (CoVe), que integram recuperação de conhecimento externo ou processos de verificação para aumentar a confiabilidade das respostas. Schulhoff et al. (SCHULHOFF et al., 2025) também abordam RAG extensivamente no contexto de agentes.

Além dessas técnicas, aspectos como a clareza da instrução, a especificação do formato de saída desejado (e.g., JSON), e o controle de parâmetros de inferência como temperatura e top-p (discutidos por Lo (LO, 2023) e Boonstra (BOONSTRA, 2025)) são componentes integrais da engenharia de prompts eficaz. A escolha e a combinação criteriosa dessas técnicas, juntamente com um processo iterativo de teste e refinamento, são fundamentais para o desenvolvimento de chatbots pedagógicos que atendam aos objetivos educacionais e ofereçam interações de alta qualidade e valor para os estudantes. A necessidade de uma abordagem sistemática e bem documentada para a engenharia de prompt é, portanto, essencial para o avanço de aplicações de IA na educação (SAHOO et al., 2025).

2.4 Feedback, Avaliação e Aprendizagem

O feedback é um elemento central no processo de aprendizagem, orientando os alunos na identificação de seus pontos fortes e fracos para o alcance dos objetivos educacionais. Características como a descrição detalhada, conselhos corretivos específicos e o estabelecimento de um diálogo entre instrutores e alunos são essenciais para um feedback de qualidade (FALCÃO et al., 2023). A personalização do feedback, além de aprimorar sua eficácia, impulsiona a motivação e o engajamento dos estudantes.

Estudos indicam que o feedback gerado por LLMs pode influenciar positivamente a motivação para tarefas futuras e promover emoções positivas, reforçando a integração de aspectos afetivos e motivacionais na avaliação (MEYER et al., 2024). Além disso, a adoção de ferramentas de IA pelos estudantes tem sido significativa. Segundo o *Digital Education Council Global AI Student Survey 2024*, 86% dos estudantes afirmam utilizar IA em seus estudos, com 24% utilizando diariamente (Digital Education Council, 2024). Essa alta taxa de adoção destaca a importância de integrar efetivamente as tecnologias de IA nos processos de feedback e avaliação para atender às expectativas e necessidades dos alunos de maneira eficaz e responsável.

A avaliação, por sua vez, requer alinhamento com os objetivos de aprendizagem e a consideração dos diferentes níveis de complexidade cognitiva. A Taxonomia de Bloom, frequentemente utilizada para categorizar perguntas, oferece uma estrutura para classificar as questões de acordo com a demanda cognitiva, da memorização à criação (RODRIGUES et al., 2024). Essa estrutura permite uma avaliação e análise mais abrangente das habilidades dos alunos e dos feedbacks dos LLMs. Ademais, a personalização da aprendizagem, facilitada pelos LLMs, permite que as experiências de aprendizagem sejam adaptadas às necessidades e ao progresso individual de cada aluno (MAITY; DERROY, 2024).

2.5 Considerações Éticas

A ética no uso de LLMs na educação é primordial. A garantia de justiça, imparcialidade, transparência e privacidade dos dados são cruciais para o desenvolvimento e implementação responsáveis de sistemas baseados em LLMs (MAITY; DERROY, 2024). A validação empírica, por meio da comparação entre o feedback gerado por LLMs e a ausência de feedback, é essencial para assegurar a contribuição efetiva da tecnologia para a aprendizagem (MEYER et al., 2024).

Adicionalmente, é importante considerar as preocupações e expectativas dos estudantes em relação ao uso de IA na educação. Conforme o *Digital Education Council Global AI Student Survey 2024*, 71% dos estudantes desejam estar envolvidos no processo de decisão sobre quais ferramentas de IA são implementadas em suas instituições (Digital Education Council, 2024). Além disso, 72% esperam que as universidades ofereçam treinamento sobre o uso efetivo de ferramentas de IA. Essas percepções ressaltam a necessidade de envolver os alunos na implementação de tecnologias de IA e fornecer orientações claras para promover a adoção responsável e eficaz.

2.6 Recuperação Aumentada por Geração (RAG) para Contextualização em Chatbots Educacionais

Apesar das capacidades impressionantes dos Modelos de Linguagem de Grande Escala (LLMs) em diversas tarefas de Processamento de Linguagem Natural (PNL), eles enfrentam desafios intrínsecos, como a propensão a gerar “alucinações” - informações factualmente incorretas ou não fundamentadas - e a limitação imposta pelo seu conhecimento estático, que reflete apenas os dados sobre os quais foram treinados e não eventos ou informações recentes (GAO et al., 2024; CHURCH et al., 2024). Essas limitações são particularmente problemáticas em contextos educacionais, onde a precisão factual e a relevância contextual do conteúdo são primordiais. A técnica de Recuperação Aumentada por Geração (RAG, do inglês Retrieval-Augmented Generation) emergiu como uma solução paradigmática robusta para mitigar esses problemas, integrando dinamicamente conhecimento externo aos LLMs no momento da inferência (LEWIS et al., 2021).

Conforme introduzido por Lewis et al. (2021), o RAG combina um modelo de recuperação de informação (*retriever*) com um modelo de geração de texto (*generator*, tipicamente um LLM). O objetivo é fundamentar a geração do LLM em trechos de informação relevantes recuperados de uma base de conhecimento externa, como os materiais de um curso específico. Esta abordagem, como descrito por Church et al. (2024), permite que chatbots, por exemplo, transcendam os dados públicos de seu treinamento original e utilizem informações privadas ou recentes “*just-in-time*” para responder a consultas específicas,

como questões sobre uma fatura particular em um cenário de atendimento ao cliente ou, no contexto educacional, dúvidas sobre o conteúdo específico de uma aula.

2.6.1 Funcionamento Detalhado e Componentes Chave do RAG

A implementação de um sistema RAG tipicamente envolve um fluxo de processamento que pode ser dividido em etapas de indexação e de recuperação/geração, conforme detalhado no *survey* de Gao et al. (2024). A Figura 1 apresenta os componentes e processos-chave do RAG.

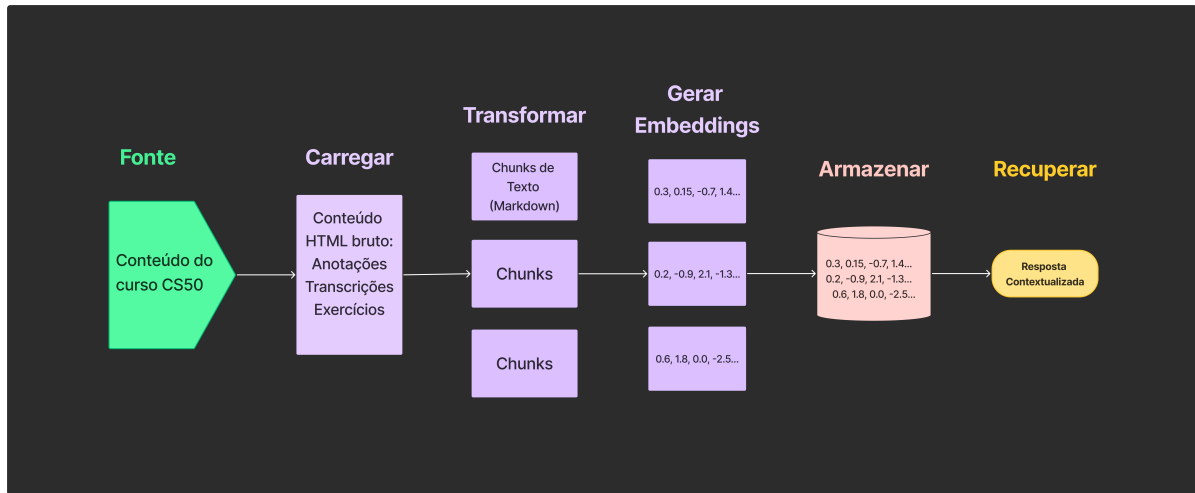


Figura 1 – Fluxo de Dados do Pipeline de Preparação e Indexação para o Sistema RAG

❑ **Base de Conhecimento Externa:** Consiste nos documentos que contêm a informação relevante para o domínio de aplicação. No contexto deste trabalho, é datasets dos curso (CS50) (MALAN; UNIVERSITY, 2024), compreendendo transcrições de aulas, anotações e exercícios (B; PURWAR, 2024). Esses documentos são a fonte da verdade para o sistema RAG.

❑ **Processo de Indexação (Chunking e Embedding):**

- *Chunking (Segmentação):* Os documentos da base de conhecimento são divididos em segmentos menores, ou *chunks*, para otimizar o processo de recuperação. A granularidade desses chunks (e.g., parágrafos, sentenças, ou blocos de tamanho fixo) é um parâmetro importante que pode afetar a qualidade da recuperação (GAO et al., 2024).
- *Embeddings (Representações Vetoriais):* Cada chunk de texto é transformado em uma representação vetorial densa, conhecida como *embedding*. Este processo é realizado por modelos de embedding especializados, como os baseados em arquiteturas Transformer (e.g., Sentence-BERT (REIMERS; GUREVYCH,

2019)), que são treinados para capturar o significado semântico do texto. A escolha de um modelo de embedding *open source* eficaz, como o BAAI/bge-large-en-v1.5 utilizado por B e Purwar (B; PURWAR, 2024), é crucial para a qualidade da busca por similaridade.

- ❑ **Vector Stores (Bancos de Dados Vetoriais):** Os embeddings gerados são armazenados e indexados em um banco de dados vetorial. Essas bases de dados são otimizadas para realizar buscas por similaridade em alta velocidade em espaços vetoriais de alta dimensão, utilizando algoritmos de Busca por Vizinhos Mais Próximos Aproximados (ANN), como os implementados em bibliotecas como FAISS (JOHNSON; DOUZE; JÉGOU, 2017) e utilizados em sistemas RAG (CHURCH et al., 2024). Mesmo que ferramentas como o OpenWebUI abstraíam a complexidade da gestão do vector store, o conceito de indexação para recuperação eficiente é fundamental.
- ❑ **Mecanismo de Retrieval (Recuperação):** Quando um usuário submete uma consulta (e.g., uma pergunta), esta também é convertida em um embedding usando o mesmo modelo utilizado para a base de conhecimento. O sistema de retrieval então busca no vector store os K chunks cujos embeddings são mais similares (e.g., por similaridade de cosseno) ao embedding da consulta. A qualidade e relevância desses chunks recuperados são vitais no sistema.
- ❑ **Geração Contextualizada:** Os chunks de texto recuperados são então concatenados com a consulta original do usuário para formar um prompt enriquecido. Este prompt aumentado é fornecido ao LLM (o gerador), que utiliza tanto seu conhecimento paramétrico interno quanto o contexto fornecido para formular uma resposta. A expectativa é que, ao ter acesso a informações relevantes e específicas do domínio, o LLM gere respostas mais precisas, factuais e menos propensas a alucinações (LEWIS et al., 2021).

2.6.2 Chatbots Educacionais e sua Relevância

A aplicação do RAG em chatbots educacionais oferece vantagens significativas, alinhando-se diretamente com os objetivos pedagógicos de fornecer suporte preciso e confiável aos estudantes:

- ❑ **Fidelidade ao Conteúdo do Curso:** A principal vantagem do RAG é garantir que as respostas do chatbot sejam predominantemente baseadas no material didático específico fornecido (e.g., transcrições de aulas, notas de estudo). Isso é essencial para evitar que o chatbot forneça informações genéricas ou incorretas que possam confundir o aluno. Conforme demonstrado por Lewis et al. (LEWIS et al., 2021), RAG gera linguagem mais específica, diversa e factual.

- ❑ **Redução de Alucinações:** Ao ancorar a geração em documentos recuperados, o RAG diminui substancialmente a tendência dos LLMs a “alucinar” ou inventar informações, um problema crítico identificado por Church et al. (CHURCH et al., 2024) e Gao et al. (GAO et al., 2024).
- ❑ **Conhecimento Dinâmico e Atualizável:** Diferentemente do conhecimento “congelado” no tempo de um LLM pré-treinado, a base de conhecimento de um sistema RAG pode ser facilmente atualizada. Novas aulas, materiais complementares ou erratas podem ser adicionados ao vector store, permitindo que o chatbot forneça informações sempre correntes sem a necessidade de retreinar o LLM subjacente, que é um processo custoso.
- ❑ **Transparência e Rastreabilidade (Potencial):** Em muitas implementações de RAG, é possível (e desejável) que o sistema indique as fontes (os chunks de documentos) que foram utilizadas para gerar uma resposta específica. Essa capacidade de rastreamento é pedagogicamente valiosa, pois permite que alunos e educadores verifiquem a informação e se aprofundem no material original.
- ❑ **Adaptabilidade a Domínios Específicos:** RAG é particularmente eficaz para adaptar LLMs generalistas a domínios de conhecimento específicos, como o conteúdo de um curso técnico, sem a necessidade de fine-tuning extensivo para cada domínio.

2.6.3 RAG versus Fine-tuning para Incorporação de Conhecimento Factual

Enquanto o fine-tuning (ajuste fino) de um LLM pode ser utilizado para adaptar o modelo a um estilo específico, a um conjunto de tarefas ou para incutir certos comportamentos, ele é geralmente menos eficaz e mais custoso para a incorporação de grandes volumes de conhecimento factual específico e dinâmico, quando comparado ao RAG (LEWIS et al., 2021). O fine-tuning para conhecimento factual exigiria a criação de vastos datasets de treinamento (e.g., pares pergunta-resposta sobre todo o conteúdo do curso), um processo que pode ser caro, demorado e computacionalmente intensivo. Além disso, cada vez que o conhecimento factual precisasse ser atualizado (e.g., uma nova aula adicionada), o modelo precisaria ser parcialmente ou totalmente retreinado.

O RAG, por outro lado, separa o “cérebro” linguístico do LLM da base de conhecimento factual. O conhecimento é armazenado externamente e acessado sob demanda. Isso torna a atualização do conhecimento uma simples questão de atualizar os documentos na base de recuperação, uma tarefa significativamente mais leve e barata do que o retreinamento de um LLM. Lewis et al. (LEWIS et al., 2021) demonstram que o RAG pode superar o desempenho de modelos puramente paramétricos (sem recuperação) em tarefas que exigem conhecimento intensivo, e também superar arquiteturas específicas de

tarefa que dependem de recuperação e extração. Embora o fine-tuning possa ter seu papel em adaptar o componente gerador do RAG para melhor utilizar os contextos recuperados ou para seguir instruções específicas, o RAG se apresenta como a abordagem mais eficiente e flexível para prover aos LLMs conhecimento factual atualizado e específico de um domínio.

2.7 Trabalhos Relacionados

A aplicação de assistentes conversacionais baseados em Inteligência Artificial, comumente conhecidos como chatbots, no domínio educacional tem sido objeto de crescente investigação e desenvolvimento. Esta seção visa apresentar uma revisão da literatura pertinente aos principais eixos deste trabalho, situando a presente pesquisa no panorama científico atual. O foco recai sobre a utilização de Modelos de Linguagem de Grande Escala (LLMs) e, em particular, da técnica de Recuperação Aumentada por Geração (RAG), para a criação de chatbots pedagógicos acessíveis e eficazes.

2.7.1 Panorama dos Chatbots na Educação e a Emergência do RAG

A crença de que o diálogo desempenha um papel significativo no processo de aprendizagem é amplamente difundida entre pesquisadores da área de educação (SWACHA; GRACEL, 2025). Contudo, na prática educacional contemporânea, muitos estudantes são privados da oportunidade de interações dialógicas aprofundadas com seus educadores ou pares, seja devido ao grande número de alunos por turma ou às limitações inerentes ao ensino online assíncrono. A introdução de chatbots baseados em LLMs, capazes de gerar respostas conversacionais realistas e convincentes de forma instantânea, como o ChatGPT, representou uma solução potencialmente revolucionária para esta questão (SWACHA; GRACEL, 2025). Uma vantagem importante desses sistemas é a capacidade de fomentar um ambiente de aprendizado mais solidário, onde os estudantes podem expor suas dúvidas e lacunas de conhecimento sem o receio associado à interação com professores ou colegas, mitigando barreiras sociais e de poder (SWACHA; GRACEL, 2025).

Apesar das virtudes dos LLMs, sua aplicação direta em contextos educacionais enfrenta desafios significativos, notadamente a dificuldade em atualizar ou estender seu conhecimento de forma dinâmica e, de maneira crucial, a propensão a “alucinações” — a geração de conteúdo que é factualmente incorreto, sem sentido ou infiel a uma fonte de informação específica (SWACHA; GRACEL, 2025). Tais falhas representam uma barreira considerável para a adoção confiável de chatbots baseados em LLMs na educação.

Nesse contexto, a técnica de Recuperação Aumentada por Geração (RAG), introduzida por (LEWIS et al., 2021) e (SWACHA; GRACEL, 2025), surge como uma solução capaz de

endereçar essas limitações. Conforme detalhado por (SWACHA; GRACEL, 2025), o RAG aprimora o desempenho dos LLMs não através da otimização de seu treinamento, mas sim pela integração de técnicas de recuperação de informação. Essencialmente, o RAG utiliza a entrada do usuário para recuperar documentos ou trechos de texto relevantes de uma base de conhecimento externa e os fornece como contexto adicional ao LLM durante a geração da resposta. Esta abordagem permite reduzir significativamente as alucinações, ao fundamentar as respostas do chatbot em informações verificáveis e específicas do domínio.

O survey conduzido por Swacha e Gracel (SWACHA; GRACEL, 2025) sobre aplicações de chatbots RAG na educação identifica um campo em rápida expansão, embora ainda recente. Os autores analisaram 47 publicações dedicadas ao uso de chatbots RAG para diversos fins educacionais, destacando o potencial dessa arquitetura para atender a necessidades específicas do domínio. Esta pesquisa investiga o caráter dessas aplicações, o público-alvo (estudantes ou professores), o escopo temático do conhecimento acessível via chatbot, os LLMs subjacentes e os métodos de avaliação empregados, fornecendo um panorama valioso do estado da arte. Eles ressaltam que, embora o RAG tenha se estabelecido como um meio prático para desenvolver aplicações LLM especializadas, e seja particularmente útil em áreas onde a confiabilidade do modelo é primordial (como saúde e educação), ainda não havia, até o momento de sua publicação, um survey dedicado a mapear o progresso da aplicação do RAG especificamente no campo educacional (SWACHA; GRACEL, 2025). Este Trabalho de Conclusão de Curso (TCC) se insere nesse contexto, buscando contribuir com a exploração de implementações RAG acessíveis e focadas na qualidade pedagógica, utilizando o ecossistema open source.

2.7.2 Chatbots Educacionais Baseados em Retrieval-Augmented Generation (RAG)

A aplicação da técnica de Recuperação Aumentada por Geração (RAG) tem se mostrado particularmente promissora para o desenvolvimento de chatbots educacionais mais robustos e contextualmente relevantes. A capacidade do RAG de integrar informações de bases de conhecimento externas e específicas do curso diretamente no processo de geração de respostas pelos LLMs permite mitigar problemas como alucinações e conhecimento desatualizado, que são barreiras significativas para a adoção de chatbots em contextos de ensino e aprendizagem (SWACHA; GRACEL, 2025). Diversos trabalhos recentes têm explorado a implementação e avaliação de chatbots RAG em ambientes de educação superior, demonstrando seu potencial e identificando áreas para desenvolvimento futuro.

Neumann et al. (NEUMANN et al., 2025) apresentam o MoodleBot, um chatbot integrado ao LMS Moodle e direcionado a um curso de Bancos de Dados e Sistemas de Informação. Este sistema utiliza uma arquitetura RAG para fornecer suporte aos estudantes, respondendo a questões organizacionais e de conteúdo da disciplina. O estudo

destaca a importância de uma base de conhecimento específica do curso (e.g., slides de aula, notas) e avalia a aceitação do chatbot pelos alunos através do Modelo de Aceitação de Tecnologia (TAM), além de verificar a acurácia das respostas. Os autores concluem que chatbots baseados em LLM, como o MoodleBot, podem melhorar significativamente o processo de ensino-aprendizagem, embora apontem para a necessidade de aprimoramentos no *fact-checking* automatizado e na exploração de LLMs *self-hosted* para otimizar custos e controle (NEUMANN et al., 2025). A capacidade do RAG de fornecer respostas ancoradas no conteúdo do curso é vista como um diferencial chave para a eficácia pedagógica.

Em uma linha similar, Thway et al. (THWAY et al., 2024) introduzem o “Professor Leodar”, um chatbot RAG customizado e com persona própria, projetado para um curso de Introdução à Ciência de Dados e Inteligência Artificial na Nanyang Technological University, Singapura. O objetivo central é não apenas aprimorar a aprendizagem, mas também combater o “Botpoop” — informações de baixa qualidade ou incorretas geradas por chatbots. O estudo emprega uma abordagem de métodos mistos para avaliar o impacto do chatbot na aprendizagem, engajamento e preparo para exames, reportando uma alta taxa de experiências positivas (97.1%) entre os participantes. Fatores como a qualidade da saída do RAG, o uso de estratégias de questionamento escalonado (*scaffolding*) e a persona do chatbot foram identificados como cruciais para sua eficácia (THWAY et al., 2024). O trabalho ressalta o potencial de chatbots customizados de Inteligência Artificial Generativa (GenAI), como o Professor Leodar, e a importância da integração de conhecimento em tempo real e do design de persona.

Hicke et al. (HICKE et al., 2023) abordam o desafio de responder a um grande volume de perguntas de estudantes em plataformas online de (QA), especialmente em cursos de computação. Eles propõem o AI-TA, uma solução que utiliza LLMs *open source* da família LLaMA-2, combinando RAG, fine-tuning supervisionado (SFT) e Otimização por Preferência Direta (DPO) para garantir a privacidade dos dados. Experimentando com um dataset da plataforma Piazza de um curso introdutório de ciência da computação, os autores demonstram uma melhoria significativa na qualidade das respostas, com o RAG sendo uma adição particularmente impactante (HICKE et al., 2023). Este trabalho é notável por ser um dos primeiros a combinar RAG, SFT e DPO para desenvolver um sistema de (QA) robusto especificamente para o contexto educacional, utilizando LLMs *open source* (HICKE et al., 2023). A ênfase na privacidade dos dados e o uso de modelos abertos alinham-se com a necessidade de soluções acessíveis e controláveis para instituições de ensino.

O trabalho de Zhao et al. (ZHAO et al., 2025) apresenta o sistema *SlideItRight*, que combina feedback textual gerado por LLM com a recuperação de slides de aula visualmente relevantes. O sistema utiliza RAG para analisar a resposta de um aluno a uma pergunta e, em vez de fornecer apenas texto, recupera o slide mais pertinente do material do curso para complementar o feedback. A pesquisa investiga os ganhos de aprendizagem e a percepção

dos alunos em um design 2x2 (feedback humano vs. IA, com vs. sem slide). Embora não tenham encontrado diferenças estatisticamente significativas nos ganhos de aprendizagem entre as condições, o estudo revelou que os alunos consideraram o feedback com slides útil, apesar de expressarem menor confiança no feedback da IA em comparação com o humano. Este estudo inspira o presente trabalho ao demonstrar uma aplicação criativa de RAG que vai além da recuperação de texto, fundamentando a resposta da IA em um artefato multimodal do próprio curso.

Coletivamente, esses estudos indicam um forte interesse na aplicação de RAG para desenvolver chatbots educacionais mais eficazes. Os principais temas que emergem incluem a importância de (i) ancorar as respostas do chatbot no conteúdo específico do curso para garantir relevância e reduzir alucinações, (ii) utilizar LLMs *open source* para promover acessibilidade, privacidade e controle, e (iii) avaliar não apenas a acurácia técnica, mas também a experiência do usuário e o impacto pedagógico. Enquanto o RAG oferece uma melhoria significativa em relação a LLMs sem acesso a conhecimento externo, a combinação com outras técnicas como SFT e DPO, como explorado por Hicke et al. (HICKE et al., 2023), aponta para direções futuras na otimização desses assistentes virtuais. O presente TCC se baseia nesses achados, focando na viabilidade de uma arquitetura RAG *open source* e na sua avaliação qualitativa em múltiplos contextos de curso.

2.7.3 Outras Aplicações e Abordagens de IA Generativa na Educação

Além das implementações diretas de chatbots RAG para suporte ao conteúdo de cursos específicos, a Inteligência Artificial Generativa têm sido explorada em uma variedade de outras aplicações e abordagens pedagógicas no contexto educacional. Esses estudos frequentemente investigam o uso de IA para complementar o ensino tradicional, facilitar a aprendizagem autodirigida, ou desenvolver ferramentas especializadas para domínios de conhecimento particulares.

Liu et al. (LIU et al., 2024a) relatam a experiência de integrar uma suíte de ferramentas baseadas em IA no curso CS50 da Universidade de Harvard. Com o objetivo de aproximar uma razão de 1:1 entre professor e aluno através de software, eles desenvolveram ferramentas para explicar trechos de código, aprimorar o estilo de programação e responder a dúvidas curriculares e administrativas. A abordagem incluiu o uso de um chatbot (CS50 Duck, ou CS50.ai), que emprega RAG para fundamentar suas respostas no material do curso, e políticas de uso que incentivavam o uso dessas ferramentas internas em detrimento de IAs comerciais. Os autores destacam que a integração ponderada de IA, com “guarda-rails” pedagógicos que guiam os estudantes em vez de fornecer soluções diretas, pode aprimorar a experiência de aprendizado ao oferecer suporte contínuo e personalizado (LIU et al., 2024a). O estudo detalha a arquitetura do sistema, as estratégias

de prompt e os desafios de implementação, como a prevenção de ataques de injeção de prompt e o gerenciamento do uso através de um sistema de “corações” (throttling).

Focando no apoio à aprendizagem autodirigida e à autoavaliação, Ali et al. (ALI et al., 2023) projetaram e prototiparam o TeacherGAIA, um aplicativo de chatbot GenAI. Este trabalho primeiro identificou diversas abordagens construtivistas de aprendizagem que promovem a autonomia do estudante, como construção de conhecimento, aprendizagem baseada em investigação, autoavaliação e ensino por pares. Posteriormente, utilizaram engenharia de prompt com um LLM da OpenAI (GPT-4) para direcionar as interações do chatbot de forma a apoiar essas diferentes abordagens, contrastando com o modo de transmissão tradicional frequentemente observado em chatbots como o ChatGPT. Os autores ressaltam que o TeacherGAIA foi implementado para oferecer orientação cognitiva e suporte socioemocional de maneira generativa, sem regras pré-especificadas ou “hard-coding”, e antecipam sua utilidade para facilitar a aprendizagem autodirigida entre estudantes do ensino fundamental e médio” (ALI et al., 2023). O estudo enfatiza a importância do design de persona e da engenharia de prompt para alinhar o comportamento do chatbot com objetivos pedagógicos específicos.

Chen et al. (CHEN et al., 2025) apresentam o WisdomBot, um LLM especificamente ajustado (tuned) para o domínio educacional, combinando o poder dos LLMs com teorias educacionais. Para superar as limitações de LLMs generalistas em contextos de ensino, como a necessidade de conhecimento especializado e explicações concisas, eles propõem um método de tuning que utiliza conceitos de conhecimento auto-instruídos e instruções guiadas pela Taxonomia de Bloom. Adicionalmente, para aprimorar a precisão em questões factuais durante a inferência, o WisdomBot integra duas estratégias de aumento por recuperação: recuperação de uma base de conhecimento local e recuperação de um motor de busca. Os autores demonstram a eficácia de sua abordagem aplicando-a a LLMs chineses, mostrando que os modelos ajustados podem gerar respostas mais confiáveis e profissionais em diversos cenários educacionais (CHEN et al., 2025). Este trabalho destaca a importância de alinhar LLMs com teorias pedagógicas e a sinergia entre fine-tuning e RAG para otimizar o desempenho em domínios especializados como a educação.

Esses trabalhos ilustram a diversidade de abordagens para integrar IA generativa na educação. Enquanto alguns focam na aplicação de RAG para fornecer respostas contextuais e factuais, como no CS50.ai (LIU et al., 2024a), outros exploram a engenharia de prompt para alinhar LLMs com teorias de aprendizagem construtivistas, como no TeacherGAIA (ALI et al., 2023), ou propõem o fine-tuning de modelos com conhecimento pedagógico e aprimoramento por recuperação, como no WisdomBot (CHEN et al., 2025). Em comum, esses estudos reconhecem tanto o imenso potencial da IA para transformar o suporte à aprendizagem quanto os desafios inerentes à sua implementação responsável e eficaz. O presente TCC se beneficia desses insights ao propor uma arquitetura RAG baseada em ferramentas open source, com um foco particular na qualidade pedagógica da

interação e na acessibilidade da solução.

2.7.4 Considerações Técnicas, Infraestrutura Experimental e Modelos de Linguagem Menores

A implementação prática de LLMs e sistemas RAG no contexto educacional envolve considerações técnicas significativas, desde a escolha do modelo e da infraestrutura até a viabilidade de execução local versus soluções baseadas na nuvem. A crescente disponibilidade de modelos *open source*, incluindo os Modelos de Linguagem Pequenos (SLMs), oferece novas perspectivas para a criação de infraestruturas experimentais e de produção mais acessíveis e controláveis pelas instituições de ensino.

Zesch et al. (ZESCH et al., 2024) descrevem a iniciativa FLEXI (FernUni LLM Experimental Infrastructure) na FernUniversität in Hagen, Alemanha, que visa estabelecer uma infraestrutura de LLM *open source* mantida localmente. A motivação principal é superar os desafios associados ao uso de serviços comerciais baseados na nuvem, como problemas de segurança de dados, equidade educacional reduzida devido a custos e a falta de controle sobre os modelos (ZESCH et al., 2024). O projeto FLEXI opta por modelos abertos cujos pesos estão disponíveis, permitindo sua execução, modificação e aprimoramento em servidores da própria universidade. Eles detalham a configuração de hardware (servidores com GPUs NVIDIA) e software (utilizando Ollama para servir os LLMs e OpenWebUI como interface de chat), e discutem o processo de seleção de modelos, considerando aspectos como abertura, idioma, qualidade (avaliada por benchmarks como ARC, GSM8K, MMLU) e segurança (ZESCH et al., 2024). O trabalho fornece orientações práticas para instituições que consideram implementar seus próprios servidores de LLM, destacando que, embora os LLMs baseados na nuvem sejam fáceis de configurar, os modelos *open source* locais são atualmente mais custo-efetivos e oferecem maior proteção de dados (ZESCH et al., 2024). A iniciativa FLEXI visa gerar evidências sobre a viabilidade e os desafios do uso de LLMs *open source* locais no ensino superior.

Investigando especificamente a viabilidade de Modelos de Linguagem Pequenos (SLMs) com RAG como alternativa a LLMs maiores e proprietários, Liu et al. (LIU et al., 2024b) conduzem um estudo comparativo no contexto do aprendizado de Ciência da Computação. Para endereçar preocupações com segurança de dados, privacidade e controle inerentes ao uso de LLMs baseados na nuvem, os autores criaram um SLM armazenado localmente (neural-chat-7b-v3, uma versão fine-tuned do Mistral-7B) e o integraram com diferentes métodos RAG. Este sistema foi comparado com LLMs populares como gpt-3.5-turbo e gpt-4-32k, utilizando dados conversacionais reais de um curso introdutório de Ciência da Computação (CS1) (LIU et al., 2024b). Os resultados indicam que um SLM com RAG pode alcançar um desempenho similar, e em alguns casos superior, aos LLMs maiores, especialmente quando se considera a avaliação por modelos *open source* (LIU et al., 2024b).

Liu et al. (LIU et al., 2024b) concluem que o uso de SLMs com RAG é uma alternativa viável para educadores de computação, permitindo o desenvolvimento de assistentes de IA escaláveis e com maior controle sobre políticas de dados. O estudo também ressalta a importância da componente RAG e da Inteligência de Concentração (IC), um framework de recuperação em camadas proposto pelos autores, para otimizar a relevância e a especificidade das respostas em cenários de recursos limitados (LIU et al., 2024b).

Ambos os trabalhos convergem na importância de explorar alternativas *open source* e/ou locais às soluções de LLM baseadas na nuvem para o setor educacional. Zesch et al. (ZESCH et al., 2024) focam na criação de uma infraestrutura universitária que possibilite a experimentação e a inovação com uma variedade de modelos abertos, enquanto Liu et al. (LIU et al., 2024b) fornecem evidências empíricas de que SLMs, quando adequadamente aumentados com RAG, podem de fato substituir LLMs maiores em tarefas específicas de suporte ao aprendizado. Essas considerações técnicas e de infraestrutura são fundamentais para o desenvolvimento de soluções de IA educacional que sejam não apenas eficazes, mas também sustentáveis, seguras e alinhadas com os valores e necessidades das instituições de ensino. O presente TCC se beneficia dessas discussões ao optar por uma arquitetura local baseada em ferramentas como Ollama e OpenWebUI para a prototipagem e avaliação de um *chatbot* pedagógico com RAG, priorizando a acessibilidade e o controle tecnológico.

Metodologia

Este capítulo apresenta a metodologia adotada para o desenvolvimento e avaliação do chatbot pedagógico com Recuperação Aumentada por Geração para o curso CS50: Introduction to Computer Science (MALAN; UNIVERSITY, 2024). O objetivo central do projeto foi implementar uma ferramenta de apoio ao aprendizado que aliasse viabilidade técnica a uma abordagem pedagogicamente eficaz.

Inicialmente realizou-se o levantamento de requisitos e a análise de diferentes aplicações de IA na educação (Seção 3.1). Em seguida, descreve-se a seleção de tecnologias *open source* e a configuração do ambiente de implementação (Seção 3.2). A escolha e preparação dos dados educacionais (CS50), que formam a base de conhecimento do sistema (Seção 3.3).

Para a validação do sistema, foi gerado e qualificado um conjunto sintético de perguntas que simulam dúvidas de alunos - categorizadas pela Taxonomia de Bloom - e por respostas de referência, que servem como gabarito (Seção 3.4). Posteriormente, aborda-se a implementação efetiva do *chatbot*, incluindo a engenharia de *prompt* pedagógico (Seção 3.6). Por fim, apresenta-se a metodologia de avaliação das respostas do *chatbot* com RAG (Seção 3.6.3).

3.1 Brainstorming e Levantamento de Requisitos

A fase inicial do projeto consistiu em um *brainstorming* detalhado, no qual os autores discutiram diferentes perspectivas e ideias sobre o uso de modelos de linguagem extensos (LLMs) no contexto educacional. Este processo foi fundamentado por meio de uma revisão da literatura existente, na análise de tendências de mercado sobre a aplicação de inteligência artificial na educação e nas experiências empíricas dos autores como discente/docente. O objetivo central foi identificar as necessidades e oportunidades de uso de LLMs que atendessem tanto às demandas dos alunos quanto às dos professores.

Como resultado deste *brainstorming*, foram propostas diversas possibilidades de aplicação de IA em plataformas de cursos *online* como ilustrado na (Figura 2) e no texto

seguinte:

1. **Chatbot pedagógico com RAG para orientação no conteúdo do curso:**
Um agente de IA que utiliza a abordagem de Recuperação Aprimorada por Geração (*Retrieval-Augmented Generation*, RAG) para contextualizar conteúdos de cursos, focando em uma abordagem pedagógica.
2. **Agente de *feedback* para respostas de alunos em perguntas dissertativas:**
Um agente de IA equipado com RAG, projetado para fornecer *feedback* instantâneo e relevante às respostas dissertativas dos alunos nos cursos. O *feedback* baseia-se no conteúdo específico do curso, auxiliando o aluno a identificar pontos fortes e áreas de melhoria em suas respostas, promovendo um aprendizado mais ativo e direcionado.
3. **Agente de avaliação para suporte à aprovação ou reprovação do aluno:**
Este agente analisaria dados de desempenho e *analytics* do aluno, fornecendo suporte adicional para decisões de aprovação ou reprovação com base em critérios predefinidos. Embora relevante, optou-se por não prosseguir com essa opção devido aos complexos aspectos éticos envolvidos na automação de decisões acadêmicas.
4. **Gerador de exercícios e materiais complementares:** Este agente utilizaria as transcrições das videoaulas para criar exercícios e materiais suplementares que complementassem o conteúdo estudado. Contudo, essa opção foi descartada por apresentar riscos de sobreposição com as atividades dos professores, o que poderia impactar negativamente a qualidade e a personalização do curso.



Figura 2 – Aplicações de IA na Educação Online sugeridas no Brainstorming

Após uma análise das possibilidades e limitações de recursos e do escopo do trabalho vigente, optou-se por desenvolver **apenas a opção 1 (chatbot pedagógico com RAG)**. Esta escolha motivou-se pelo potencial de impacto direto no engajamento dos alunos e na eficácia do *feedback* em tempo real para dúvidas sobre o conteúdo, sem comprometer a qualidade das atividades educacionais ou sobrepor as responsabilidades pedagógicas do professor.

3.2 Ambiente de Implementação e Ferramentas Tecnológicas

Esta seção descreve o ambiente de desenvolvimento e as ferramentas tecnológicas selecionadas para a implementação do *chatbot* pedagógico com RAG. A escolha dos componentes foi guiada pelos princípios de acessibilidade, custo-benefício e alinhamento com o ecossistema *open source*, visando demonstrar a viabilidade da solução em hardwares econômicos.

3.2.1 Arquitetura Conceitual do Sistema

A arquitetura do *chatbot* pedagógico é altamente modular, facilitando a interação entre seus componentes e permitindo a substituição de tecnologias para a adaptação a diferentes cenários. Conforme ilustrado na Figura 3, o sistema é composto por: (i) uma interface de usuário (*Frontend*), (ii) um orquestrador de *backend* (OpenWebUI), (iii) um serviço de LLM e Embedding local (Ollama), e (iv) uma base de conhecimento externa no banco de dados vetorial, alimentada pelos documentos do curso educacional. Esta estrutura visa equilibrar flexibilidade, desempenho e o compromisso com soluções *open source* e execução local.

3.2.2 Ambiente de Desenvolvimento e Infraestrutura de Experimento

A concepção inicial do projeto previa a execução de todos os componentes de IA em um ambiente local para garantir a soberania total dos dados. O *hardware* utilizado para o desenvolvimento, descrito como Ryzen 5 1600, Radeon RX 580 8GB, 16GB RAM, foi inicialmente considerado para esta tarefa. No entanto, durante a fase de implementação, essa configuração mostrou-se insuficiente para hospedar simultaneamente o modelo de embedding, o backend, o frontend da aplicação e um LLM. Devido a essa limitação de recursos computacionais para executar a pilha completa, a metodologia foi adaptada para a utilização de APIs externas de inferência de LLMs. Essa mudança não apenas viabilizou o projeto, mas também permitiu a inclusão de uma valiosa análise comparativa. Além

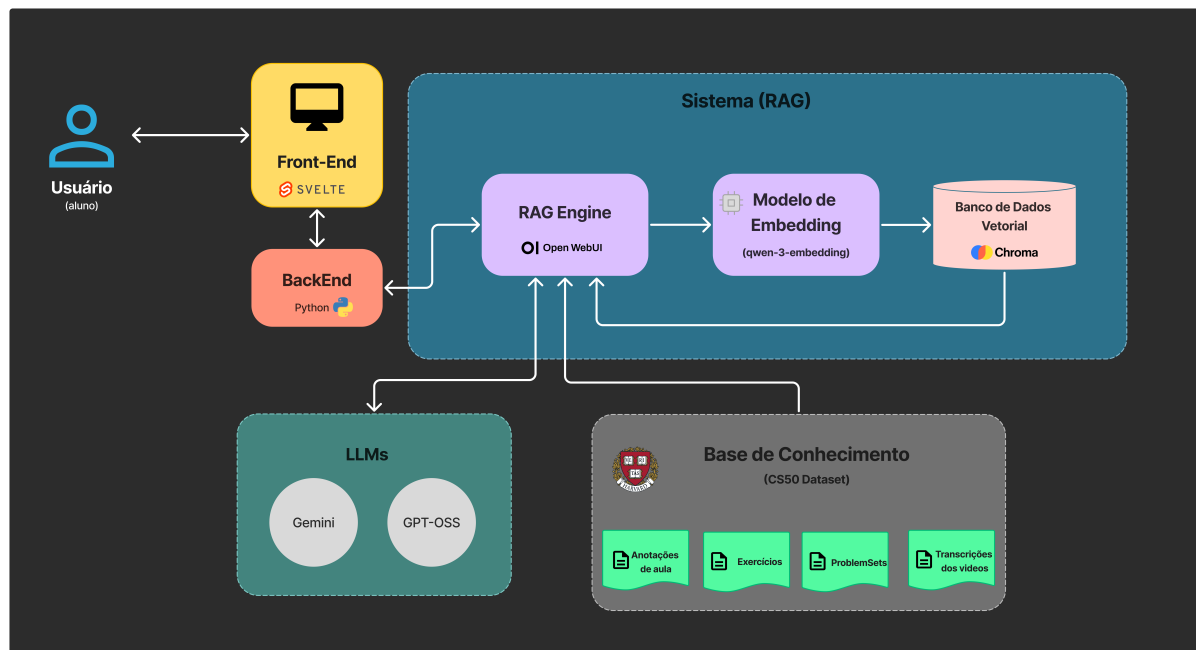


Figura 3 – Arquitetura Conceitual do Chatbot Pedagógico com RAG.

disso, ao utilizar APIs gratuitas, manteve-se a replicabilidade da solução sem a necessidade de infraestrutura de ponta, validando a tese de acessibilidade da IA em contextos de orçamento limitado (LU et al., 2025).

3.2.3 Componentes Tecnológicos do Sistema

Esta subseção detalha os principais componentes tecnológicos selecionados para a implementação do *chatbot* pedagógico, justificando suas escolhas e destacando suas funcionalidades essenciais.

3.2.3.1 Plataforma de Interação Conversacional: OpenWebUI

A **OpenWebUI** foi selecionada como a interface principal para o desenvolvimento e interação com o *chatbot* pedagógico, atuando também como orquestrador dos diversos componentes de IA. Sua escolha foi fundamentada por uma pesquisa exploratória de plataformas *open source* disponíveis, com o objetivo de identificar uma solução que aliasse maturidade tecnológica, flexibilidade para pesquisa e um conjunto de funcionalidades alinhado às demandas de um *chatbot* educacional com RAG. A interface intuitiva e responsiva do OpenWebUI, que serve como ponto de contato direto para o usuário com o chatbot pedagógico, é ilustrada na Figura 4.

A análise comparativa com outras plataformas *open source* para chatbots, resumizada na Tabela 1, reforçou a escolha do OpenWebUI devido ao seu ecossistema completo e suporte nativo e avançado a RAG.

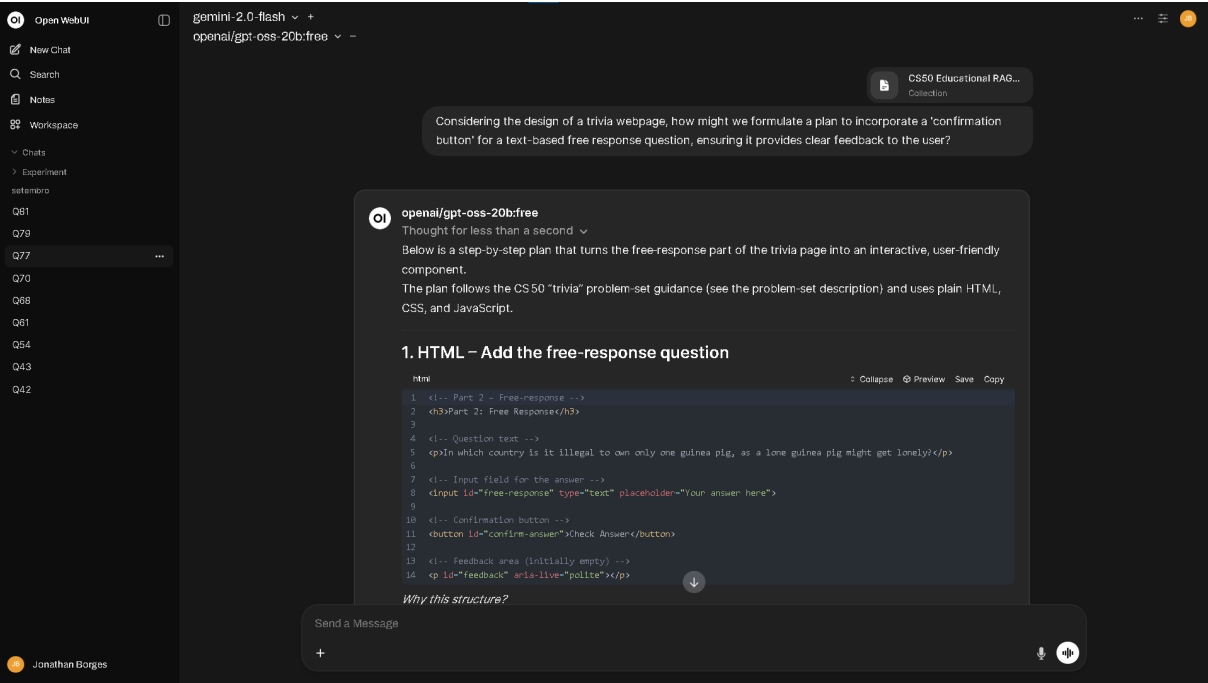


Figura 4 – Interface de interação do usuário no OpenWebUI, exibindo o chatbot com uma resposta gerada por LLM.

Tabela 1 – Análise Comparativa de Plataformas Open-Source para Chatbots

Plataforma	Foco Principal	Suporte a RAG	Maturidade da Comunidade
Chatbot UI	Interface minimalista	Básico	Alta
LibreChat	Fidelidade à API OpenAI	Avançado	Média
LobeChat	Alto desempenho e design	Em desenvolvimento	Alta
Open WebUI	Ecossistema completo e RAG	Nativo e Avançado	Muito Alta

A popularidade do OpenWebUI, evidenciada por um elevado número de contribuições e popularidade no GitHub, atesta sua validação por uma vasta comunidade de usuários e desenvolvedores, traduzindo-se em um software robusto, em constante evolução e com amplo suporte, o que o torna uma escolha técnica e cientificamente sólida para este projeto. Sua adoção fundamenta-se em pilares essenciais: a transparência e reprodutibilidade científica, inerente à sua natureza *open source* (MANCHANDA et al., 2025); a facilidade de implantação e acessibilidade, via imagem Docker; e, crucialmente, sua flexibilidade arquitetural e agnosticismo tecnológico (B; PURWAR, 2024). Essa flexibilidade é demonstrada por sua arquitetura, que combina um *backend* em Python com FastAPI e um *frontend* em Svelte/SvelteKit com TypeScript, assim como, Pydantic para validação e `python-jose` e `passlib` para segurança (Token Web JSON (JWT), *hashing* de senhas). A plataforma permite a integração com múltiplos *backends* de LLMs (como Ollama e APIs externas) e oferece liberdade na escolha dos componentes da *pipeline* de RAG, com suporte nativo a bancos de dados vetoriais como ChromaDB, Milvus, Qdrant, Pinecone, PGVector,

Elasticsearch, e para persistência de dados relacionais, SQLAlchemy com suporte a SQLite por padrão e PostgreSQL, além de Redis para cache temporário e orquestração via LangChain. Suas funcionalidades avançadas incluem interatividade em tempo real via WebSockets, monitoramento com OpenTelemetry, e um robusto sistema de Controle de Acesso Baseado em Função (RBAC) para a gestão de usuários (Figura 5). As capacidades de RAG da plataforma são aprimoradas pela ingestão de documentos locais, busca web integrada e geração de citações. Finalmente, a extensibilidade e customização (com *Framework de Pipelines* e *Construtor de Modelos e Ferramentas*) e o suporte completo à renderização de Markdown e LaTeX reforçam sua relevância para aplicações pedagógicas. É importante notar que, embora o OpenWebUI mantenha sua natureza de código aberto, sua licença BSD-3 modificada exige a aquisição de uma licença empresarial para a remoção do logo e nome da marca na interface, o que pode impactar a decisão de adoção por algumas instituições.

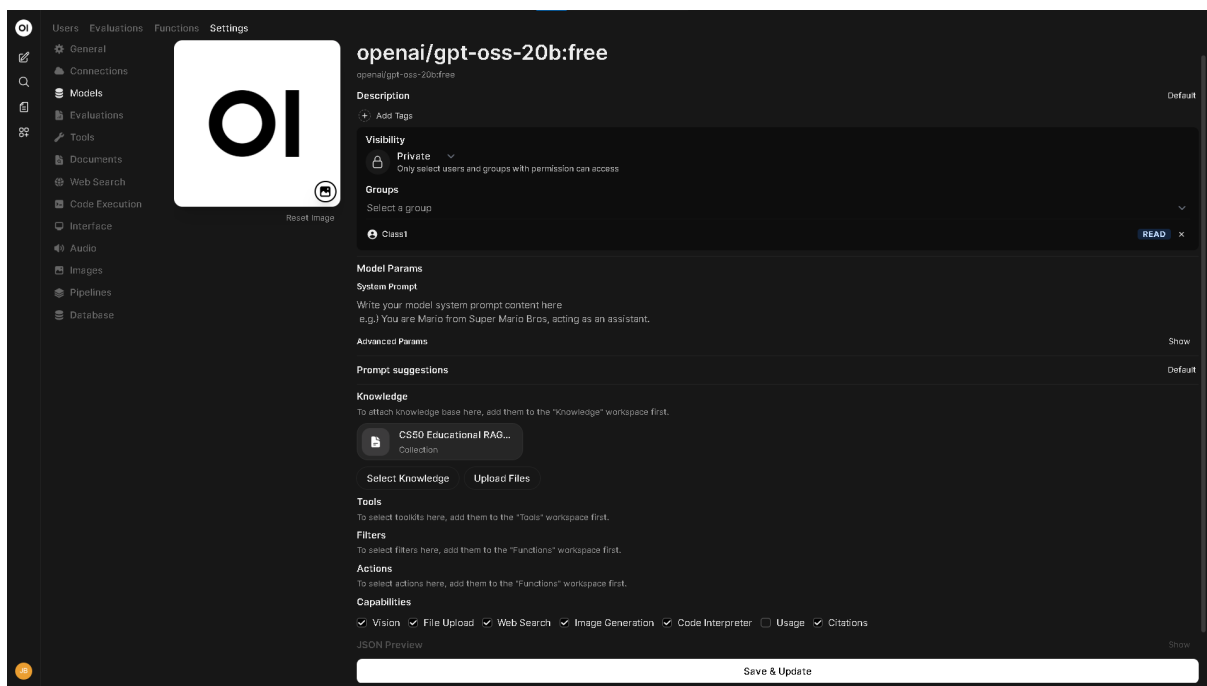


Figura 5 – Interface de administração do OpenWebUI, destacando as configurações de modelos de LLM, atribuição de acesso a grupos de usuários e associação a bases de conhecimento RAG.

3.2.3.2 Modelos de Linguagem e *Embedding*

A seleção dos modelos de linguagem e de *embedding* foi um componente crítico, orientada pela necessidade de avaliar diferentes arquiteturas e paradigmas de acesso (proprietário *vs.* *open source*).

Para a etapa de representação vetorial do texto (*embedding*), o Ollama foi empregado como gerenciador de modelos para hospedar e executar localmente o modelo *open source*

Qwen3-Embedding-0.6B (ZHANG et al., 2025). Sua escolha é justificada pela simplicidade de uso, capacidade de download e gestão de modelos leves, e compatibilidade nativa com a OpenWebUI. Apesar de seus 0.6 bilhões de parâmetros, o ‘Qwen3-Embedding’ alcança resultados de estado da arte em diversos *benchmarks* de *embedding* (ZHANG et al., 2025), tornando-o uma escolha eficiente e acessível para a execução em hardware local, contribuindo para os objetivos de democratização da IA (LU et al., 2025).

Devido a limitações do hardware de desenvolvimento para executar a pilha completa localmente, foram utilizadas APIs externas para inferência dos (LLMs). É fundamental destacar que todos os experimentos foram conduzidos utilizando os planos gratuitos das plataformas, resultando em **custo monetário zero**. Os modelos utilizados foram:

- ❑ **Google AI Studio** (Google LLC, 2025): Plataforma do Google que fornece acesso à família de modelos Gemini e Gemma. Foi utilizada em duas frentes: (1) para geração e avaliação de dados sintéticos, empregando o modelo **gemini-2.0-flash-lite** por sua eficiência em tarefas de larga escala; e (2) como um dos agentes do *chatbot* RAG, com o modelo **gemini-2.0-flash** (COMANICI et al., 2025), representando uma solução proprietária de ponta para fins de comparação.
- ❑ **OpenRouter** (OpenRouter Team, 2025): Um agregador de APIs que facilita o acesso a uma vasta gama de modelos de linguagem, incluindo muitos modelos *open source*. Para este trabalho, o OpenRouter foi o *gateway* para o modelo **gpt-oss-20b** (OPENAI et al., 2025), que serviu como o principal agente *open source* no *chatbot* RAG. Esta abordagem manteve a essência do projeto de avaliar um modelo de código aberto, contornando as limitações do hardware local.

3.2.4 Mapa Geral dos Componentes Tecnológicos Usados

A Figura 6 oferece uma visão geral dos componentes tecnológicos. Para um detalhamento funcional, a Tabela 2 descreve o papel de cada ferramenta utilizada na pesquisa, organizada conforme os blocos visuais apresentados na figura.

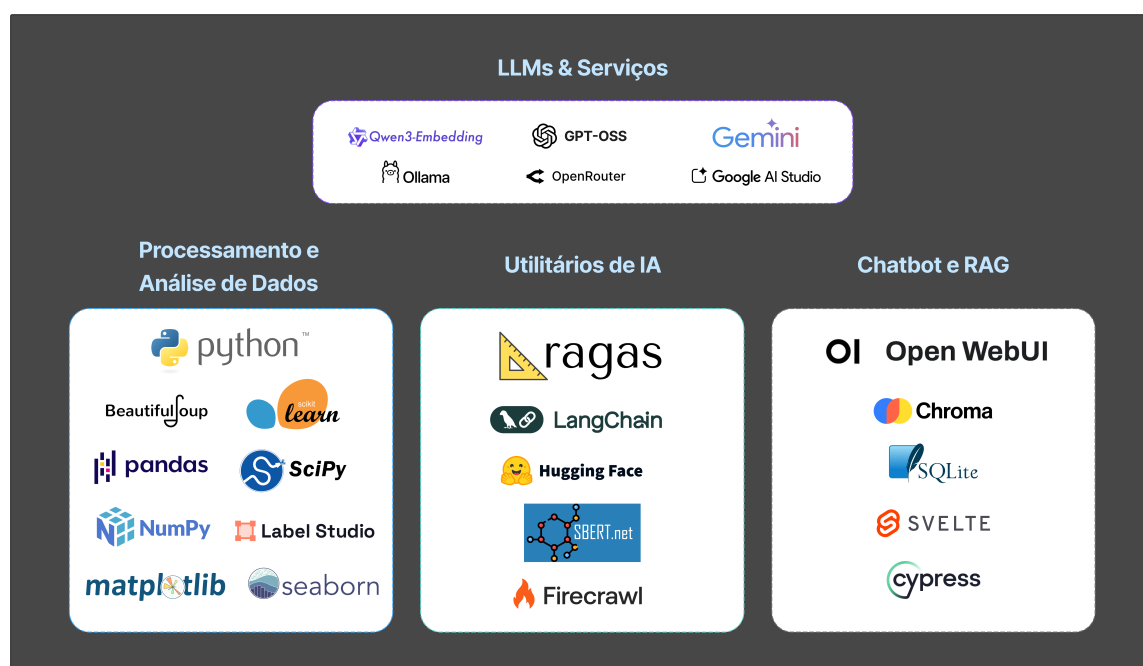


Figura 6 – Visão Geral dos Componentes Tecnológicos Utilizados.

Categoria	Descrição do Uso
LLMs & Serviços	
Qwen3-Embedding 0.6B	Modelo de <i>embedding open source</i> para vetorização de texto.
Ollama	Hospedagem e execução local de modelos (<i>ex</i> : Qwen3-Embedding).
GPT-OSS 20B	LLM <i>open-weight</i> (via OpenRouter) para geração de respostas.
OpenRouter	<i>Gateway</i> para acesso a modelos de LLM via API externa.
Gemini Flash 2.0	LLM proprietário do Google (via AI Studio) para geração de respostas.
Google AI Studio	Plataforma para acesso à API do modelo Gemini.
Processamento e Análise de Dados	
Python	Linguagem de programação principal utilizada em todo o projeto.
BeautifulSoup	Biblioteca para <i>web scraping</i> do conteúdo do CS50.
scikit-learn	Usada para calcular similaridade de cosseno.
pandas	Biblioteca para manipulação e análise de <i>datasets</i> .
NumPy	Base para cálculos numéricos, incluindo amostragem estratificada e operações com arrays.
SciPy	Biblioteca para computação científica, usada para cálculos estatísticos.
Label Studio	Ferramenta <i>open source</i> para anotação e validação de dados.
Matplotlib	Biblioteca para visualização de dados e gráficos.
Seaborn	Biblioteca para visualização de dados e gráficos.
Utilitários de IA	
Ragas	Framework para avaliação e geração de dados sintéticos.
LangChain	Framework de orquestração para aplicações com LLMs.
Hugging Face	Plataforma para modelos de <i>embeddings open source</i> .
sentence-transformers	Biblioteca Python para a geração de <i>embeddings</i> .
Firecrawl	Ferramenta para mapeamento e extração de links de páginas web.
Chatbot e RAG	
OpenWebUI	Interface principal e orquestrador do <i>chatbot</i> pedagógico.
ChromaDB	Banco de dados vetorial <i>open source</i> para RAG.
SQLite	Banco de dados relacional padrão do <i>backend</i> do OpenWebUI.
SVELTE	Framework JavaScript para o <i>frontend</i> do OpenWebUI.
Cypress	<i>Framework</i> para automação usado na coleta de dados de avaliação.

Tabela 2 – Componentes Tecnológicos Utilizados e Sua Aplicação.

3.3 Coleta e Preparação do Dataset CS50

A construção do *dataset* foi fundamentada em um processo metodológico rigoroso, que abrangeu desde a seleção criteriosa da fonte de dados e das ferramentas tecnológicas até a coleta automatizada, curadoria manual e enriquecimento semântico dos documentos. Esta seção detalha as etapas que garantiram a alta qualidade e a confiabilidade do material coletado.

3.3.1 Justificativa da Escolha da Fonte de Dados (CS50)

A escolha do curso *CS50: Introduction to Computer Science* (MALAN; UNIVERSITY, 2024) de Harvard como fonte de dados para este projeto foi deliberada e baseada em múltiplos critérios. Primeiramente, o CS50 é um curso de renome internacional, amplamente reconhecido pela sua excelência pedagógica, material didático de alta qualidade

e abordagem abrangente dos fundamentos da ciência da computação. Sua estrutura bem definida, que combina aulas teóricas, notas de aula detalhadas e problemas práticos, oferece um corpus de dados rico e diversificado.

Adicionalmente, um fator relevante para a viabilidade do estudo foi a política de licenciamento do material. O conteúdo do CS50 é disponibilizado sob a licença *Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)*, permitindo a reutilização e adaptação do material para fins não comerciais e de pesquisa, o que foi fundamental para garantir a conformidade com os direitos autorais durante a coleta e a subsequente análise dos dados.

3.3.2 Processo de Coleta e Pré-processamento dos Dados

A coleta e o pré-processamento dos dados do curso CS50 seguiram um *pipeline* personalizado para lidar com a complexidade e as especificidades do material.

3.3.2.1 Ferramentas de Coleta e Extração

A pilha tecnológica para a coleta foi focada em robustez e flexibilidade, utilizando a linguagem de programação Python e suas bibliotecas para *web scraping*. Ferramentas automatizadas genéricas para conversão de Linguagem de Marcação de Hipertexto (HTML) para Markdown mostraram-se insuficientes devido à estrutura rica e interativa do site do CS50, que incorpora elementos como Scratch Blocks, Ascinema Players e Blocos de Código Customizados. Diante disso, foi desenvolvido um pipeline customizado utilizando `BeautifulSoup`, `requests` e `markdownify` para garantir um controle granular sobre o parsing e a conversão, transpondo fielmente o conteúdo para o formato Markdown. A ferramenta `firecrawl` (Firecrawl contributors, 2025) foi empregada para mapear e extrair links, garantindo cobertura completa.

3.3.2.2 Extração de Conteúdo e Curadoria

A estrutura do dataset reflete a organização do curso, sendo composta por quatro tipos de documentos: *notes*, *transcription*, *problemset* e *exercises*. Os documentos do tipo *notes* correspondem às notas de aula, arquivos HTML do site com explicações claras. As *transcriptions* foram obtidas das legendas das vídeo-aulas do YouTube, enquanto *problemset* e *exercises* são arquivos HTML que contêm uma breve descrição e links de cada exercício e os exercícios detalhados, respectivamente. A distribuição desses documentos por tipo, bem como seu tamanho e frequência ao longo das semanas, é visualizada em detalhes na Figura 7. Os dados brutos foram coletados do site oficial do CS50 (MALAN; UNIVERSITY, 2024) e das transcrições das vídeo-aulas do YouTube.

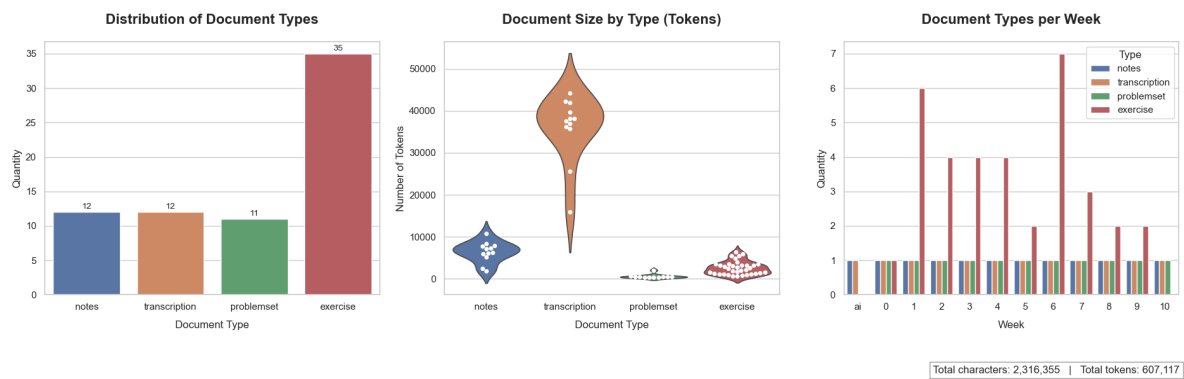


Figura 7 – Análise da distribuição dos documentos por tipo, tamanho em *tokens* e distribuição ao longo das semanas do curso.

Scripts personalizados em Python (`scrape_notes.py`) extraíram sistematicamente notas de aula e listas de problemas. As transcrições foram obtidas das legendas automáticas do YouTube, visando proporcionar um conteúdo rico e detalhado. Após a coleta, os dados passaram por um pipeline de processamento que incluiu conversão para Markdown, enriquecimento com metadados YAML (*front matter*) via `add_metadata.py` (contendo título, tipo, semana, links etc.) e uma rigorosa curadoria manual. Cada arquivo gerado foi inspecionado por um revisor humano para a correção de erros e verificação de consistência, assegurando qualidade e fidelidade. Um exemplo de cabeçalho YAML é apresentado abaixo:

```
---
link: https://youtu.be/3LPJfIKxwWc
related_files:
- week-0-notes.md
- problemset/week-0-problemset.md
- problemset/week-0-scratch.md
title: "Lecture 0: Week 0 - Scratch"
type: transcription
week/lecture: "0"
---
```

3.3.2.3 Padronização do Formato Markdown

Para facilitar a análise e a integração com sistemas de IA, todos os arquivos foram padronizados em formato Markdown. As transcrições, especificamente, foram processadas pelo script `format_transcription_md.py` para adicionar títulos de seção, onde cada título contém um link direto para o *timestamp* exato do vídeo ou para a seção correspondente no site do CS50, enriquecendo o contexto de recuperação.

```
# [Lecture 4] (https://cs50.harvard.edu/x/2024/notes/4/#lecture-4)

- [Welcome!] (#welcome)
- [Pixel Art] (#pixel-art)
...
- [File I/O] (#file-io)
- [Summing Up] (#summing-up)

## [Welcome!] (https://cs50.harvard.edu/x/2024/notes/4/#welcome)

- In previous weeks, we talked about images being made of smaller building blocks
  ↪ called pixels.
- Further, we will discuss how to access the underlying data stored in computer
  ↪ memory.

## [Pixel Art] (https://cs50.harvard.edu/x/2024/notes/4/#pixel-art)

- Pixels are squares, individual dots, of color that are arranged on an up-down ...
```

3.3.2.4 Agregação e Disponibilização

Finalmente, todos os arquivos Markdown foram agregados em formatos tabulares através do script `build_dataset_csv.py`, resultando em dois arquivos principais: `dataset.csv`, para análises estatísticas, e `dataset.jsonl`, para integração direta com sistemas de *machine learning* e o *vector store* para RAG e IA. Este dataset será disponibilizado publicamente, reforçando a transparência e a reprodutibilidade da pesquisa.

3.4 Geração de Dados Sintéticos de Perguntas e Respostas (QA)

A geração sintética de conjuntos de dados de QA é uma etapa fundamental para a avaliação sistemática de sistemas de Recuperação Aumentada por Geração (RAG) em contextos educacionais. Este trabalho apresenta uma contribuição significativa ao propor e implementar uma metodologia para a criação de um dataset de QA com avaliação humana, que não apenas preenche a lacuna de dados anotados para o domínio CS50, mas também garante diversidade cognitiva inspirada na Taxonomia de Bloom, tornando-o um referencial robusto e reprodutível. Portanto, tal processo visa criar amostras representativas e diversificadas, alinhadas ao conteúdo de um curso ou base de dados, permitindo a análise quantitativa e qualitativa do desempenho dos sistemas de IA em tarefas de compreensão e recuperação de informação. Conforme o plano de pesquisa, a metodologia de

criação de perguntas foi estruturada para garantir a qualidade e a diversidade, utilizando uma abordagem híbrida que combina a automação via LLMs com supervisão humana.

3.4.1 Fundamentação Metodológica para Geração de QA

A abordagem adotada neste projeto baseia-se na utilização de modelos de linguagem de grande porte (LLMs) para a geração automática de QA. Este processo é orquestrado pela biblioteca RAGAS (EXPLODINGGRADIENTS, 2024) para a síntese de perguntas e respostas, no qual emprega o modelo ‘gemini-2.0-flash’ (COMANICI et al., 2025) acessado pela API do Google AI Studio (Google LLC, 2025). A geração parte de um *corpus* educacional estruturado, previamente processado e transformado em um grafo de conhecimento. Neste grafo, cada nó representa um documento ou fragmento relevante do curso, enriquecido com metadados como palavras-chave, *headlines* e entidades. Para garantir a diversidade cognitiva das perguntas geradas, são incorporadas personas inspiradas na Taxonomia de Bloom (BLOOM et al., 1956), cada uma representando diferentes níveis de complexidade cognitiva (lembrar, compreender, aplicar, analisar, avaliar e criar). Cada persona possui uma descrição detalhada de seu perfil e objetivos pedagógicos, orientando o LLM na formulação de questões que abrangem desde o simples reconhecimento de fatos até a síntese criativa de conceitos.

3.4.2 Pipeline de Geração de Dados Sintéticos de QA

O pipeline de geração, implementado utilizando a biblioteca RAGAS (EXPLODINGGRADIENTS, 2024), foi projetado para criar conjuntos de dados sintéticos de QA a partir dos materiais do curso CS50 (notas de aula, transcrições das vídeo-aulas, listas de exercícios e arquivos *problemset*). O processo é composto por múltiplas etapas, ilustradas na Figura 8:

1. **Ingestão e Estruturação do Conhecimento:** Os documentos brutos do curso são carregados e estruturados em um grafo de conhecimento, no qual cada nó representa um documento ou fragmento (chunk) e as arestas representam relações semânticas ou estruturais (por exemplo, relação de pertencimento entre chunk e documento). Este grafo serve como base para a seleção dos contextos de referência das perguntas.
2. **Transformação e Enriquecimento do Grafo:** O grafo é enriquecido por meio de técnicas de PNL, como extração de títulos, divisão em tópicos e identificação de palavras-chave, utilizando modelos de linguagem (LLMs).
3. **Definição de Personas e Estilos de Consulta:** São definidas múltiplas personas, cada uma associada a um nível da Taxonomia de Bloom, representando diferentes

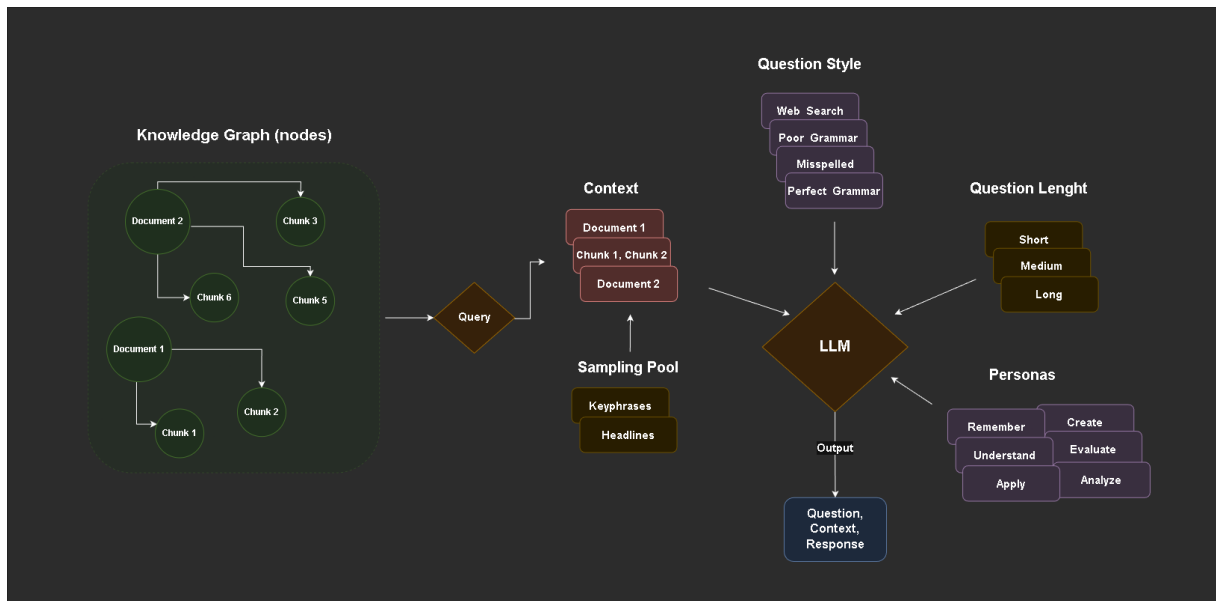


Figura 8 – Fluxo do pipeline de geração de dados sintéticos de QA.

O grafo de conhecimento (*Knowledge Graph*) é composto por nós de documentos e seus respectivos fragmentos (*chunks*), enriquecidos com propriedades como *headlines* e *keyphrases*. A partir desse grafo, são selecionados contextos e amostras semânticas para compor o *sampling pool*. O modelo de linguagem (*LLM*) recebe como entrada o contexto, parâmetros de estilo, comprimento e persona, gerando como saída pares de questão, contexto e resposta. Cada componente do diagrama representa uma etapa fundamental do processo de geração automática de conjuntos de dados para avaliação de sistemas de QA.

níveis cognitivos (lembrar, compreender, aplicar, analisar, avaliar, criar). As personas são implementadas como objetos contendo nome, descrição de papel e restrições específicas para orientar a geração de perguntas.

4. **Definição de Personas e Estilos de Consulta:** São definidas múltiplas personas, cada uma associada a um nível da Taxonomia de Bloom, representando diferentes níveis cognitivos (lembrar, compreender, aplicar, analisar, avaliar, criar). As personas são implementadas como objetos contendo nome, descrição de papel e restrições específicas para orientar a geração de perguntas. Para ilustrar a estrutura dessas personas, um exemplo resumido da implementação: 9.
5. **Distribuição de Queries:** Para cada nó do grafo, são geradas combinações entre termos relevantes (extraídos de headlines ou keyphrases), personas, estilos de pergunta e comprimentos de resposta. Este processo assegura a cobertura de diferentes tipos de questões e contextos.
6. **Geração de Perguntas e Respostas:** Utilizando o grafo enriquecido, prompts cuidadosamente projetados e uma distribuição de estilos de consulta o LLM recebe

o contexto selecionado, o perfil da persona, o termo alvo, o estilo e o comprimento desejado. O modelo é instruído a gerar perguntas e respostas fiéis ao contexto de forma automática, simulando interações realistas de alunos com diferentes perfis.

7. **Estruturação dos Dados:** Cada amostra gerada é registrada como uma instância de `SingleTurnSample`, contendo os campos `user_input` (pergunta), `reference` (resposta de referência), `reference_contexts` (contexto utilizado), `persona_name`, `query_style`, `query_length`, `synthesizer_name`, entre outros metadados relevantes para rastreabilidade e análise posterior.
8. **Anotação e Pós-processamento:** Scripts adicionais são empregados para adicionar colunas como o tipo do documento e o caminho do contexto de referência, facilitando a análise estrutural e a rastreabilidade das amostras.

```

1  from ragas.testset.persona import Persona
2
3  # Persona para o nível "Lembrar" da Taxonomia de Bloom
4  persona_remember = Persona(
5      name="Novice Recaller",
6      role_description=(
7          "A beginner student focused on recalling basic facts, definitions, and
8          ↪ key terms. "
9          "Seeks questions that require direct recall of information.\n"
10         "Questions should use keywords like: 'what is', 'who is', 'list',
11         ↪ 'define', 'when', 'where'.\n"
12         "Avoid questions that require explanation, application, or analysis."
13     ),
14 )
15
16 # Persona para o nível "Aplicar" da Taxonomia de Bloom
17 persona_apply = Persona(
18     name="Practical Implementer",
19     role_description=(
20         "A student who understands concepts and now wants to apply them to
21         ↪ solve practical problems or new situations. "
22         "Seeks questions that require using learned knowledge to solve
23         ↪ problems or address new situations.\n"
24         "Questions should use keywords like: 'how to use X for Y',
25         ↪ 'demonstrate', 'calculate', 'solve', 'implement', 'illustrate'. \n"
26         "Avoid questions that only ask for explanation of facts, analysis, or
27         ↪ evaluation."
28     ),
29 )
30
31 # return [persona_remember, persona_understand, persona_apply, ...]

```

Figura 9 – Exemplo resumido do código Python e Prompts para definição de Personas

3.4.3 Estratégias de Diversidade e Amostragem

3.4.3.1 Rodadas de Geração

O processo de geração foi realizado em duas rodadas distintas (*Round 1* e *Round 2*, cada um com parâmetros específicos para a distribuição dos estilos de *queries*. No primeiro *round*, a distribuição foi configurada para gerar 30% das perguntas a partir de *headlines* e 70% a partir de *keyphrases*, promovendo diversidade nos tipos de questões geradas. No segundo *round*, a distribuição foi ajustada para 100% *keyphrases*, buscando maior especificidade e controle sobre os tópicos abordados. Além disso, neste segundo *round*, os grafos de conhecimento foram customizados para remover documentos do tipo *transcription*, que eram excessivamente extensos e causavam desbalanceamento na dis-

tribuição de perguntas por tipo de arquivo. Essa customização garantiu uma cobertura mais equitativa dos diferentes tipos de documentos do curso.

3.4.3.2 Deduplicação Semântica de Perguntas

A geração automática em duas rodadas produziu um conjunto inicial de 372 pares de pergunta-resposta. Devido à natureza do processo, era esperado que houvesse duplicatas ou perguntas semanticamente muito similares. Para garantir a diversidade do conjunto, foi aplicado um processo de deduplicação. Utilizou-se uma abordagem baseada em *embeddings* semânticos, com o modelo Qwen3-Embedding-0.6B, para calcular a similaridade por cosseno entre todas as perguntas. Pares de perguntas com similaridade acima de um limiar de 0.78, definido empiricamente, foram agrupados, mantendo-se apenas um representante de cada grupo.

O processo resultou na remoção de 35 perguntas (uma redução de 9.4%), consolidando o conjunto em 337 amostras únicas e evitando redundâncias que poderiam enviesar a avaliação dos sistemas RAG.

3.4.3.3 Amostragem Estratificada e Redução do Conjunto

Embora o conjunto de 337 perguntas fosse semanticamente único, seu tamanho ainda inviabilizaria a supervisão humana detalhada e a análise manual. Para tornar o projeto exequível, foi realizada uma amostragem estratificada com o objetivo de criar um subconjunto menor, porém representativo e balanceado.

O tamanho final de **120 perguntas** foi definido como um balanço estratégico entre a viabilidade da anotação manual por um especialista e a necessidade de manter uma cobertura estatística suficiente das diversas facetas do curso. A principal diretriz da amostragem foi garantir uma distribuição equitativa de **10 perguntas para cada uma das 12 semanas** do curso (incluindo a semana sobre IA). Adicionalmente, a amostragem foi ponderada para manter a representatividade dos seguintes estratos:

- ❑ **Tipo de contexto de referência** (documento vs. chunk);
- ❑ **Persona** (os seis níveis cognitivos da Taxonomia de Bloom);
- ❑ **Estilo da pergunta** (*web search*, *poor grammar*, *misspelled*, *perfect grammar*);
- ❑ **Tamanho da pergunta** (curta, média ou longa).

Este procedimento assegurou que o *dataset* final de 120 perguntas, utilizado para a avaliação, fosse diversificado e imparcial, cobrindo de forma equilibrada todo o conteúdo do curso e os diferentes perfis de interação simulados.

3.4.4 Validação da Qualidade dos Dados Sintéticos de QA

Para assegurar a alta qualidade do dataset de perguntas e respostas sintéticas, uma etapa de validação foi conduzida, combinando avaliação automática e supervisão humana.

Após a seleção do conjunto final, foi conduzida uma etapa de supervisão humana utilizando a ferramenta Label Studio (TKACHENKO et al., 2020-2025). Cada pergunta e resposta foi revisada individualmente, avaliando-se critérios de clareza, coerência, relevância e alinhamento com o contexto de referência. Perguntas incoerentes, ambíguas ou desalinhadas foram corrigidas ou removidas. Essa validação manual foi fundamental para assegurar a qualidade do conjunto de dados, tornando-o adequado para experimentos rigorosos de avaliação de sistemas RAG.

3.4.5 Modelagem do Conhecimento via Grafos

A modelagem do conhecimento é realizada por meio da construção de um *grafo de conhecimento* (*Knowledge Graph*), estrutura que permite representar entidades e suas inter-relações de forma explícita e flexível. Formalmente, o grafo é definido como $G = (V, E)$, onde V é o conjunto de nós e E o conjunto de arestas 10.

- ❑ **Nós do tipo *document*:** Cada nó representa um documento integral, como transcrições, notas de aula ou listas de exercícios. Esses nós são enriquecidos com metadados estruturados, incluindo título, tipo de documento, semana de referência, caminho de origem e link original. Tal granularidade facilita a indexação e recuperação de informações.
- ❑ **Nós do tipo *chunk*:** Fragmentos textuais extraídos dos documentos originais, correspondendo a tópicos, seções ou parágrafos identificados como semanticamente relevantes. A extração é realizada por algoritmos de segmentação baseados em *headings* ou limites de *tokens*, garantindo coesão temática dos *chunks*.
- ❑ **Relações:** As arestas do grafo codificam diferentes tipos de relações. Relações hierárquicas (*child*) conectam *chunks* aos documentos dos quais foram extraídos, preservando a estrutura original do conteúdo. Relações semânticas podem ser estabelecidas entre nós com base em similaridade de palavras-chave, tópicos ou outros atributos extraídos automaticamente, permitindo a navegação entre conceitos relacionados.

O processo de construção e transformação do grafo é realizado por funções Python especializados ¹, com destaque para o uso da biblioteca Ragas. A geração dos *headlines* é realizada por algoritmos de extração implementados nesses scripts, enquanto a identificação de *keyphrases* é feita por meio de modelos de linguagem natural, especificamente o

¹ <https://github.com/dev-jonathan/ScholarRAG/tree/main/src/generation>

Gemini 2.0 Flash, que permite enriquecer os nós com informações semânticas extraídas automaticamente. Tal abordagem potencializa a representação do conhecimento, tornando o grafo apto para tarefas avançadas de consulta, geração de questões e análise contextual.

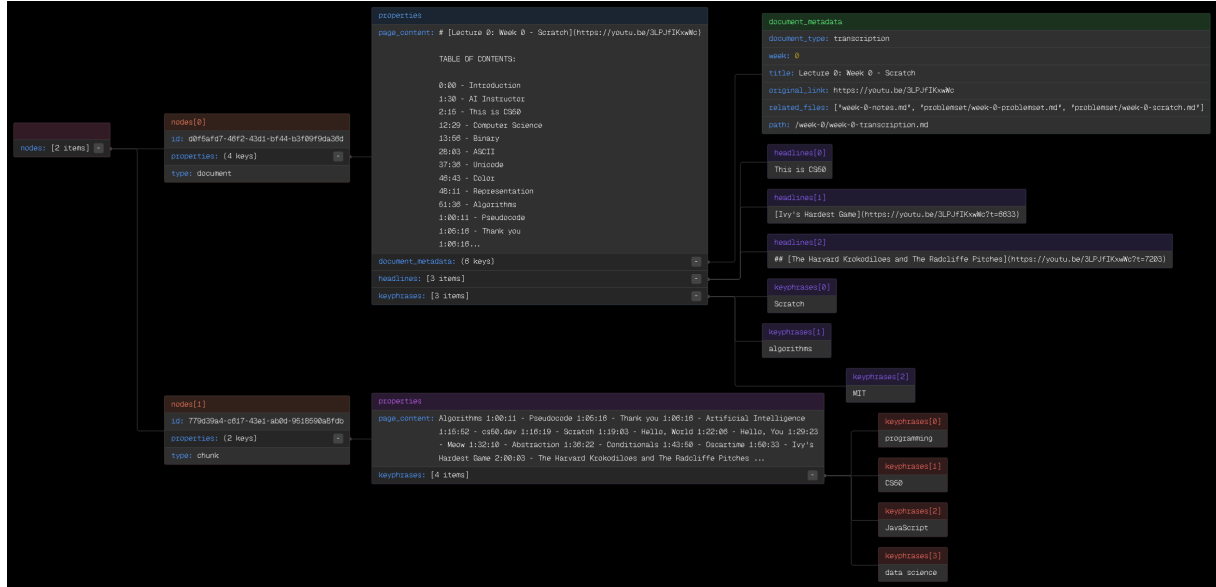


Figura 10 – Visualização de um grafo de conhecimento. Nota: A imagem apresenta uma versão ilustrativa e reduzida do grafo para facilitar a visualização.

3.4.6 Síntese de Perguntas e Respostas

A geração das instâncias de QA é realizada pelo componente `TestsetGenerator`, que recebe como entrada o grafo de conhecimento, uma lista de personas e uma distribuição de estilos de consulta. O processo segue os seguintes passos:

1. **Seleção de Contextos:** Para cada instância, um contexto de referência é selecionado a partir do grafo, podendo ser um documento completo ou um chunk específico.
2. **Configuração da Persona e Estilo:** Uma persona e um estilo de consulta são amostrados conforme uma distribuição pré-definida, garantindo diversidade cognitiva e de intenção nas perguntas.
3. **Geração da Pergunta:** Utilizando um modelo de linguagem (LLM), é sintetizada uma pergunta que simula a dúvida de um aluno com o perfil da persona, baseada no contexto selecionado.
4. **Geração da Resposta de Referência:** O mesmo contexto é utilizado para gerar uma resposta de referência, que serve como gabarito para avaliação posterior.

5. **Anotação dos Metadados:** Cada instância é anotada com informações detalhadas, incluindo o tipo e caminho do contexto, persona, estilo, comprimento da consulta, entre outros.

O processo é parametrizável, permitindo o ajuste do número de instâncias por semana, a proporção de tipos de contexto (documento vs. chunk), e a diversidade de personas e estilos.

3.4.7 Reprodutibilidade e Flexibilidade

O pipeline foi desenvolvido com foco em reprodutibilidade e flexibilidade, permitindo:

- ❑ **Reexecução controlada:** O pipeline pode ser executado para diferentes semanas, cursos ou configurações, com controle de random seed e parametrização explícita.
- ❑ **Customização de grafos:** É possível gerar versões customizadas dos grafos, por exemplo, removendo documentos de transcrição para experimentos específicos.
- ❑ **Balanceamento e amostragem estratificada:** Por meio de script python realizou-se a amostragem estratificada e balanceada dos dados, garantindo diversidade e cobertura adequada dos diferentes estratos (persona, estilo, tipo de contexto, etc.).

3.4.8 Considerações de Qualidade e Diversidade

Para garantir a qualidade das perguntas e respostas geradas, o prompt de geração é continuamente refinado, exigindo respostas detalhadas, claras e instrutivas, alinhadas ao perfil da persona. São evitadas respostas genéricas ou que apenas localizem o tema no material, promovendo explicações conceituais e exemplos sempre que possível. A distribuição das personas é monitorada para assegurar representatividade de todos os níveis cognitivos, evitando viés na amostragem.

3.4.9 Importância para Avaliação de Sistemas RAG

A geração sintética controlada de QA permite a criação de conjuntos de dados customizados, adaptados ao conteúdo e aos objetivos pedagógicos do curso. Estes dados são essenciais para a avaliação rigorosa de sistemas RAG, possibilitando a mensuração de métricas como relevância, fidelidade, cobertura e capacidade de adaptação a diferentes perfis de usuário. O pipeline descrito oferece flexibilidade para ajustes e extensões, sendo aplicável a diferentes domínios educacionais e configurações de sistemas de IA.

3.5 Metodologia de Avaliação de Dados Sintéticos de QA

Neste trabalho, geramos dados sintéticos de QA a partir do conteúdo do curso educacional CS50. Para avaliar automaticamente a qualidade e coerência desses dados, utilizamos o framework Ragas (EXPLODINGGRADIENTS, 2024), que oferece métricas robustas para análise de sistemas de QA. A seguir, detalhamos as principais métricas utilizadas e os procedimentos adotados.

3.5.1 Métrica de Fidedignidade (Faithfulness)

A métrica *Faithfulness* avalia o quão consistente e fundamentada é a resposta em relação ao contexto recuperado. Uma resposta é considerada fiel se todas as suas afirmações podem ser suportadas pelo contexto fornecido.

A pontuação de fidedignidade é calculada conforme a equação:

$$\text{Faithfulness Score} = \frac{\text{Número de afirmações suportadas pelo contexto}}{\text{Número total de afirmações na resposta}} \quad (1)$$

Exemplo de uso em Python (Adaptado da documentação ²):

```

1 from ragas.dataset_schema import SingleTurnSample
2 from ragas.metrics import Faithfulness
3
4 sample = SingleTurnSample(
5     user_input="Quando foi realizado o primeiro Super Bowl?",
6     response="O primeiro Super Bowl ocorreu em 15 de janeiro de 1967.",
7     retrieved_contexts=[
8         "O Primeiro Campeonato Mundial AFL-NFL foi realizado em 15 de janeiro
9         ↳ de 1967, no Los Angeles Memorial Coliseum."
10    ]
11 )
12 scorer = Faithfulness(llm=evaluator_llm)
13 await scorer.single_turn_ascore(sample)

```

Output:

1

3.5.2 Métrica de Critério Simples (Simple Criteria Score)

A métrica *Simple Criteria Score* permite avaliar os *inputs* de forma mais subjetiva, atribuindo uma pontuação inteira baseada em critérios definidos pelo pesquisador (por

² <https://docs.ragas.io/en/v0.2.15/concepts/metrics/available_metrics/faithfulness/>

exemplo, clareza, relevância, similaridade, etc.). O avaliador (humano ou LLM) atribui uma nota de acordo com a definição do critério.

Exemplo de uso (Adaptado ³):

```

1 from ragas.dataset_schema import SingleTurnSample
2 from ragas.metrics import SimpleCriteriaScore
3
4 sample = SingleTurnSample(
5     user_input="Onde está localizada a Torre Eiffel?",
6     response="A Torre Eiffel está localizada em Paris.",
7     reference="A Torre Eiffel está localizada no Egito"
8 )
9
10 question_quality = SimpleCriteriaScore(
11     name="question_quality",
12     definition=(
13         "Score from 0 to 5 the following question based on how clear,
14         ↪ relevant, and well-grounded it is in the provided context(s).\n"
15         "A score of 0 means the question is completely unrelated or
16         ↪ nonsensical given the context.\n"
17         "A score of 1 means the question is barely related or very unclear.\n"
18         "A score of 2 means the question is somewhat related but lacks clarity
19         ↪ or relevance.\n"
20         "A score of 3 means the question is generally relevant and clear, but
21         ↪ could be improved.\n"
22         "A score of 4 means the question is clear and relevant, with minor
23         ↪ issues.\n"
24         "A score of 5 means the question is perfectly clear, highly relevant,
25         ↪ and directly based on the context.\n"
26         "Consider whether the question could plausibly be asked by a real
27         ↪ student after reading the context. "
28         "Examples:\n"
29         "0: 'What is the capital of France?' (context: JavaScript arrays)\n"
30         "1: 'Explain arrays in JavaScript.' (context: JavaScript variables)\n"
31         "3: 'How do you declare a variable in JavaScript?' (context:
32         ↪ JavaScript variables, but question is vague)\n"
33         "5: 'What is the difference between let and var in JavaScript?'
34         ↪ (context: variable declarations)\n"
35         "Answer only with a number from 0 to 5 (likert scale)."
36     ),
37 )

```

³ <https://docs.ragas.io/en/v0.2.15/concepts/metrics/available_metrics/general_purpose/>

```

28     llm=evaluator_llm,
29 )
30
31 await question_quality.single_turn_ascore(sample)
32

```

Output:

5

3.5.3 Métrica de Relevância da Resposta (Response Relevancy)

A métrica *Response Relevancy* avalia o quão relevante é a resposta em relação à pergunta original. A pontuação é baseada na similaridade entre o embedding da pergunta e o embedding de perguntas artificiais geradas a partir da resposta, conforme as equações:

$$\text{Answer Relevancy} = \frac{1}{N} \sum_{i=1}^N \text{cosine similarity}(E_{g_i}, E_o) \quad (2)$$

$$\text{Answer Relevancy} = \frac{1}{N} \sum_{i=1}^N \frac{E_{g_i} \cdot E_o}{\|E_{g_i}\| \|E_o\|} \quad (3)$$

Onde:

- E_{g_i} : Embedding da i -ésima pergunta gerada.
- E_o : Embedding da pergunta original.
- N : Número de perguntas geradas (tipicamente 3).

Exemplo de uso (adaptado ⁴):

```

1 from ragas.dataset_schema import SingleTurnSample
2 from ragas.metrics import ResponseRelevancy
3
4 sample = SingleTurnSample(
5     user_input="Quando foi realizado o primeiro Super Bowl?",
6     response="O primeiro Super Bowl ocorreu em 15 de janeiro de 1967.",
7     retrieved_contexts=[
8         "O Primeiro Campeonato Mundial AFL-NFL foi realizado em 15 de janeiro
9         ↳ de 1967, no Los Angeles Memorial Coliseum."
10     ]
11 )

```

⁴ <https://docs.ragas.io/en/v0.2.15/concepts/metrics/available_metrics/answer_relevance/>

```
11 scorer = ResponseRelevancy(llm=evaluator_llm, embeddings=evaluator_embeddings)
12 await scorer.single_turn_ascore(sample)
```

Output:

0.92

3.5.4 Procedimento do Experimento

Os dados sintéticos foram avaliados automaticamente utilizando as métricas acima, com modelos de linguagem (Gemini-2.0-flash) e embeddings de última geração (Qwen-3-embedding). Para cada amostra, as métricas foram calculadas e os resultados exportados para análise estatística.

Em etapa posterior, será realizada avaliação supervisionada humana via plataforma Label Studio (TKACHENKO et al., 2020-2025), utilizando escalas de Likert para garantir validação cruzada e rigor científico.

3.5.5 Reprodutibilidade

Todos os experimentos foram conduzidos em ambiente controlado, com scripts e configurações disponíveis no repositório do projeto. O código-fonte das avaliações automáticas encontra-se em: <https://github.com/dev-jonathan/ScholarRAG>

3.5.6 Definição dos Critérios de Avaliação Humana

Para garantir a comparabilidade entre avaliação automática e supervisão humana, os critérios de avaliação manual foram cuidadosamente alinhados às definições das métricas automáticas utilizadas. Cada critério foi operacionalizado por meio de perguntas claras ao anotador, utilizando escala de Likert de 1 a 5 pontos, conforme detalhado a seguir:

❑ Faithfulness (Fidedignidade ao Contexto)

- *Pergunta para o rótulo:* A resposta está fiel ao contexto apresentado?
- *Escala Likert:*
 - * 1: Nada fiel
 - * 2: Pouco fiel
 - * 3: Parcialmente fiel
 - * 4: Quase totalmente fiel
 - * 5: Totalmente fiel

❑ Response Relevancy (Relevância da Resposta)

Question (User Input)

Given that the lecture mentions that letters are represented using ones and zeros, how effective is the ASCII standard in mapping letters to numbers?

Context / Reference

Lecture 0

- [Welcome!](#)
- [What's Ahead](#)
- [Commsy!](#)
- [Computer Science](#)
- [ASCII](#)
- [Unicode](#)
- [Representation](#)
- [Algorithms](#)
- [Pseudocode](#)
- [Artificial Intelligence](#)
- [Scratch](#)
- [Hello World](#)
- [Hello You](#)
- [Meow and Abstraction](#)
- [Conditionals](#)
- [Oscars](#)
- [Ivy's Hardest Game](#)
- [Summing Up](#)

Welcome!

- This class is about more than computer programming!
- Indeed, this class is about problem-solving in a way that is exceedingly empowering! You will likely take the problem solving that you learn here will likely be instantly applicable to your work beyond this course and even your career as a whole!
- However, it will not be easy! You will be "drinking from the firehose" of knowledge during this course. You'll be amazed at what you will be able to accomplish in the coming weeks.
- This course is far more about you advancing "you" from "where you are today" than hitting some imagined standard.
- The most important opening consideration in this course: Give the time you need to learn through this

Answer (Reference)

The ASCII standard is quite effective in mapping letters to numbers. It was created to establish a consistent way to represent letters using binary patterns of ones and zeros. For example, the letter 'A' is mapped to the number 65, which is represented as 01000001 in binary. This standardization is crucial because it allows different systems to agree on the values that represent letters, ensuring that text messages and other forms of communication can be understood across various platforms. Without such a standard, the interpretation of binary data representing letters would be inconsistent, leading to significant communication problems. The lecture highlights the importance of standards like ASCII for this very reason.

Evaluation Criteria Guide

Faithfulness (Contextual Fidelity)

Is the answer faithful to the given context?

- 1 - Not faithful at all^[1]
- 2 - Slightly faithful^[2]
- 3 - Partially faithful^[3]
- 4 - Mostly faithful^[4]
- 5 - Completely faithful^[5]

Response Relevancy

Is the answer relevant and appropriately responsive to the question?

- 1 - Completely irrelevant^[6]
- 2 - Slightly relevant^[7]
- 3 - Partially relevant^[8]
- 4 - Mostly relevant^[9]
- 5 - Completely relevant^[10]

Simple Criteria Score (Question Quality)

Is the question clear, relevant, and well-grounded in the given context?

- 1 - Very poor (incoherent or out of context)^[11]
- 2 - Poor (unclear or hardly relevant)^[12]
- 3 - Fair (acceptable but can improve)^[13]
- 4 - Good (clear and relevant, minor improvements possible)^[14]
- 5 - Excellent (clear, relevant, and perfectly grounded in context)^[15]

☐ Remove question^[16]

Comments (optional)

Add any comments or observations here...

Figura 11 – Tela do label Studio usado pelo especialista

- *Pergunta para o rótulo:* A resposta é relevante e responde adequadamente à pergunta?
- *Escala Likert:*
 - * 1: Totalmente irrelevante
 - * 2: Pouco relevante
 - * 3: Parcialmente relevante
 - * 4: Quase totalmente relevante
 - * 5: Totalmente relevante

☐ Simple Criteria Score (Qualidade da Pergunta)

- *Pergunta para o rótulo:* A pergunta é clara, relevante e bem fundamentada no contexto apresentado?
- *Escala Likert:*
 - * 1: Muito ruim (incoerente ou fora do contexto)
 - * 2: Ruim (pouco clara ou pouco relevante)
 - * 3: Regular (aceitável, mas pode melhorar)
 - * 4: Boa (clara e relevante, pequenas melhorias possíveis)
 - * 5: Excelente (pergunta clara, relevante e perfeitamente fundamentada no contexto)

O anotador recebeu instruções detalhadas, exemplos práticos e acesso ao contexto de referência para cada amostra, promovendo consistência e rigor na avaliação. As decisões de anotação foram documentadas e divergências resolvidas por consenso ou terceira avaliação.

3.5.7 Detalhamento Formal das Métricas Utilizadas

Faithfulness: Conforme implementado em `ragas.metrics.Faithfulness`, esta métrica de avaliação automática de fidedignidade opera em três etapas: (i) decomposição da resposta em afirmações atômicas via LLM, (ii) verificação, para cada afirmação, se pode ser inferida diretamente do contexto recuperado, e (iii) cálculo da razão entre afirmações suportadas e o total de afirmações. O score final varia de 0 a 1, sendo 1 atribuído quando todas as afirmações da resposta são sustentadas pelo contexto. O método é robusto a respostas longas e múltiplas sentenças, promovendo avaliação granular e transparente.

Response Relevancy: Implementada em `ragas.metrics.ResponseRelevancy`, esta métrica avalia a relevância da resposta para a pergunta original. O LLM gera múltiplas perguntas artificiais a partir da resposta e calcula-se a similaridade de cosseno entre os embeddings dessas perguntas e o embedding da pergunta original. O score é a média dessas similaridades, penalizando respostas vagas, redundantes ou que não abordam o cerne da questão. O valor final varia tipicamente entre 0 e 1.

Simple Criteria Score: Conforme `ragas.metrics.SimpleCriteriaScore`, esta métrica permite avaliação subjetiva baseada em critério textual definido pelo pesquisador. O LLM (ou anotador humano) atribui uma nota de 0 a 5, de acordo com a clareza, relevância e fundamentação da pergunta no contexto. O prompt pode ser customizado para refletir nuances do domínio educacional, e a métrica é útil tanto para avaliação automática quanto para validação cruzada com anotação humana.

Todas as métricas foram implementadas e executadas conforme os códigos-fonte oficiais do Ragas, garantindo reprodutibilidade e aderência às melhores práticas da literatura.

3.6 Implementação e Avaliação do Chatbot Pedagógico com RAG

Esta seção detalha a implementação do chatbot pedagógico, as configurações específicas do sistema de Recuperação Aumentada por Geração (RAG) e a metodologia adotada para a avaliação qualitativa de suas interações. O objetivo é demonstrar a viabilidade técnica e pedagógica da solução em um ambiente controlado, utilizando ferramentas e modelos *open source*.

3.6.1 Configuração do Chatbot com RAG no OpenWebUI

A implementação do chatbot pedagógico foi realizada na plataforma **OpenWebUI**, que serve como interface de interação e orquestrador dos componentes de IA. As configurações foram cuidadosamente selecionadas para alinhar-se aos princípios de acessibilidade, custo-benefício e eficácia pedagógica.

- **Modelo de Linguagem (LLM):** Para a etapa de geração de respostas, foi selecionado o modelo open source gpt-oss-20b (OPENAI et al., 2025), acessado via API através da plataforma OpenRouter. A escolha por este modelo se alinha à premissa de acessibilidade do projeto por dois motivos: primeiro, por ser um modelo de código aberto (licença Apache 2.0) e, segundo, por sua arquitetura de 20 bilhões de parâmetros ser projetada para uma execução eficiente (OPENAI et al., 2025). Embora neste trabalho o acesso tenha sido via API devido às limitações de hardware para executar a pilha completa de software (conforme discutido na Seção 3.2.2), sua eficiência o torna um candidato ideal para futuras implementações totalmente locais em hardware de consumidor, reforçando a viabilidade de soluções de baixo custo (LU et al., 2025, p. 1-2).
- **Modelo de Embedding:** Para a etapa de representação vetorial do texto (embeddings), foi utilizado o modelo **Qwen3-Embedding-0.6B** (ZHANG et al., 2025). A justificativa para esta escolha é multifacetada:
 - *Natureza Open Source e Leveza:* O Qwen3-Embedding-0.6B é um modelo *open source* com apenas 0.6 bilhões de parâmetros, o que o torna excepcionalmente leve e eficiente para execução em ambientes locais com hardware modesto (TEAM, 2025). Isso se alinha perfeitamente com os objetivos deste trabalho de promover soluções acessíveis.
 - *Performance Estado da Arte:* Apesar de sua leveza, este modelo ocupa uma posição proeminente (4^a) no MTEB Leaderboard, um benchmark reconhecido para avaliação de modelos de embedding (TEAM, 2025). Essa performance indica que o Qwen3-Embedding-0.6B oferece um balanço superior entre custo

computacional e qualidade de resultados em tarefas de recuperação por similaridade, sendo a escolha ideal para o projeto.

❑ **Configurações do RAG no OpenWebUI:** A plataforma OpenWebUI oferece um sistema RAG integrado com diversas opções. Para este trabalho, foi adotada a implementação de RAG tradicional (baseline), devido a limitações de recursos computacionais e para manter o foco na avaliação de sua eficácia. As configurações específicas dos componentes do RAG são detalhadas a seguir:

- *Text Splitter:* Utilizou-se a estratégia de tokenização baseada em **Tiktoken**. Esta ferramenta é amplamente reconhecida pela sua eficiência e compatibilidade com os padrões de tokenização utilizados por modelos de linguagem modernos, contribuindo para a precisão na divisão do texto em unidades processáveis (LO, 2023).
- *Chunk Size e Chunk Overlap:* O tamanho dos segmentos de texto (*Chunk Size*) foi definido como **1024 tokens**. Esta escolha é fundamentada na necessidade de capturar um contexto semântico abrangente dentro de cada bloco, o que é crucial para materiais educacionais que frequentemente exigem explicações detalhadas e compreensão de relações entre conceitos. Trabalhos como (BHAT et al., 2025) demonstram que *chunk sizes* maiores (512-1024 tokens) são mais eficazes para *datasets* que demandam um entendimento contextual mais amplo, alinhando-se com a natureza de um *chatbot* educacional. Além disso, modelos de linguagem com janelas de contexto estendidas, como o LLM empregado neste estudo (128 mil tokens), são capazes de processar eficientemente blocos maiores. A sobreposição entre os blocos (*Chunk Overlap*) foi definida como **12%** do tamanho do *chunk*, resultando em aproximadamente **123 tokens**. Essa sobreposição visa mitigar a fragmentação de informações cruciais nas fronteiras dos *chunks*, garantindo a continuidade do contexto com um balanço otimizado entre a retenção de contexto e a minimização de redundância excessiva na base vetorial (LENG et al., 2024).
- *Top K:* O parâmetro *top k*, que define o número de *chunks* mais relevantes a serem recuperados pelo sistema de *retrieval*, foi configurado para **22**. Esta decisão é crucial para a performance em cenários de longo contexto. Estudos como (LENG et al., 2024) demonstram que, embora as janelas de contexto dos LLMs modernos sejam vastas (128 mil tokens neste caso), a acurácia de resposta do LLM em RAG pode diminuir significativamente para a maioria dos modelos *open-source* quando o contexto total excede 16 mil-32 mil tokens, devido ao fenômeno de “lost in the middle” (perda de informação no meio do contexto) (LENG et al., 2024). Ao utilizar 22 *chunks* de 1024 tokens, o contexto total fornecido ao LLM será de aproximadamente 22.528 tokens (22

$\times 1024$), que se insere na faixa de desempenho ótimo (16 mil-32 mil tokens) identificada para a maioria dos LLMs *open-source* de longo contexto. Isso assegura que o LLM receba um volume substancial de informações relevantes sem ser sobrecarregado, otimizando o balanço entre a abrangência do contexto e a capacidade efetiva de processamento do modelo.

- *Content Extraction Engine (Docling)*: Para a extração de conteúdo de diversos formatos de arquivo, o OpenWebUI integra o **Docling**, um *toolkit open source* eficiente para conversão e processamento de documentos (LIVATHINOS et al., 2025). Embora o Docling ofereça robustez para lidar com diferentes tipos de arquivos diretamente, para este trabalho, priorizou-se o rigor acadêmico e a qualidade dos dados. Consequentemente, o *dataset* do CS50 foi previamente curado e formatado para Markdown, garantindo uma base de conhecimento padronizada e otimizada para o sistema RAG. Essa estratégia não apenas otimiza o desempenho do RAG, mas também facilita a disponibilização pública do *dataset* para a comunidade científica, reforçando a transparência e a reprodutibilidade da pesquisa.

□ **Configurações do Large Language Model (LLM)**: O LLM empregado neste estudo é um modelo de última geração com de 20 bilhões de parâmetros e uma janela de contexto de 128 mil tokens. Para otimizar a qualidade e a confiabilidade das respostas para um chatbot educacional, os seguintes parâmetros de geração foram configurados com base nos achados de (BARKER et al., 2025):

- *Temperatura (Temperature)*: A temperatura de geração foi definida como **0.1**. Esta escolha visa minimizar a aleatoriedade e a probabilidade de alucinações, garantindo que o modelo produza respostas mais determinísticas e factualmente precisas, o que é crucial para a segurança e o alinhamento em um contexto educacional. Barker et al. (2025) demonstram que temperaturas baixas (0.00-0.12) são consistentemente ótimas para maximizar a segurança e o alinhamento das respostas em sistemas RAG e LLM (BARKER et al., 2025).
- *Top-p (Nucleus Sampling)*: O parâmetro *top-p* foi configurado para **0.8**. Este valor busca um equilíbrio entre a diversidade lexical e a coerência semântica. Ao realizar a amostragem de *tokens* a partir de um subconjunto com probabilidade cumulativa de 80%, o modelo gera respostas mais naturais e menos repetitivas, sem introduzir criatividade excessiva que poderia comprometer a factualidade. Embora não otimizado explicitamente em (BARKER et al., 2025), este valor se alinha com as práticas comuns para equilibrar a precisão e a fluidez em aplicações sensíveis à factualidade.

□ **Engenharia de Prompt Pedagógico (System Prompt)**: A eficácia do chatbot pedagógico depende criticamente de como o LLM é instruído a interagir. Foi de-

envolvido um *System Prompt* detalhado para guiar o comportamento do modelo, baseando-se em técnicas de engenharia de prompt discutidas na Seção 2.3: Este prompt incorpora: (i) **Role-Playing**, atribuindo ao LLM a persona de um tutor (SCHULHOFF et al., 2025; BOONSTRA, 2025); (ii) **instruções para perguntas orientadoras**, direcionando o modelo a estimular a reflexão do aluno, um princípio pedagógico fundamental (LO, 2023); e (iii) **instruções de uso do RAG**, exigindo que as respostas sejam ancoradas no contexto recuperado, garantindo fidelidade factual ao conteúdo do curso (LEWIS et al., 2021). O prompt foi refinado através de testes empíricos iterativos para otimizar a qualidade pedagógica das interações.

Persona:

You are an **experienced teacher** for the CS50 educational course. Your goal is to

- ↪ explain fundamental computer science and programming concepts, and to answer
- ↪ student questions clearly, comprehensively, well-structured, and easy to
- ↪ understand. The course is structured by 'weeks' (e.g., Week 0, Week 1), focusing
- ↪ on foundational computing, and involves lectures, notes, and problem sets. Your
- ↪ primary focus is to ensure the student understands the material, utilizing only
- ↪ the provided knowledge base. Factual accuracy is paramount.

Task:

Respond to the student's query using the provided context, incorporating inline

- ↪ citations in the format [id] **only when the <source> tag includes an explicit id**
- ↪ **attribute** (e.g., <source id="1">).

Guidelines:

- If you don't know the answer, clearly state that.
- If uncertain, ask the student for clarification.
- If the context is unreadable or of poor quality, inform the student and provide the
- ↪ best possible answer based on the *available* information.
- Your answer should directly address the question, explaining the concept itself,
- ↪ not just where it appears in the material.
- Avoid generic or overly brief answers. Do not add any information that cannot be
- ↪ found or reasonably inferred from the context.
- **Only include inline citations using [id] (e.g., [1], [2]) when the <source> tag**
- ↪ **includes an id attribute.**
- Do not cite if the <source> tag does not contain an id attribute.
- Do not use XML tags in your response.
- Ensure citations are concise and directly related to the information provided.

Example of Citation:

If the student asks about a specific topic and the information is found in a source

- ↪ with a provided id attribute, the response should include the citation like in
- ↪ the following example:

* "According to the study, the proposed method increases efficiency by 20% [1]."

Output Format:

Provide a clear, direct, and ****pedagogically structured**** response to the student's
 ↪ query, including inline citations in the format [id] only when the <source> tag
 ↪ with an id attribute is present in the context.

```
<context>
{{CONTEXT}}
</context>
```

```
<user_query>
{{QUERY}}
</user_query>
```

3.6.2 Procedimento de Experimento Automatizado

Para a avaliação sistemática do chatbot, foi desenvolvido um procedimento experimental automatizado. A automação foi implementada utilizando a ferramenta **Cypress**, um framework de testes *end-to-end* que interage diretamente com a interface web do OpenWebUI.

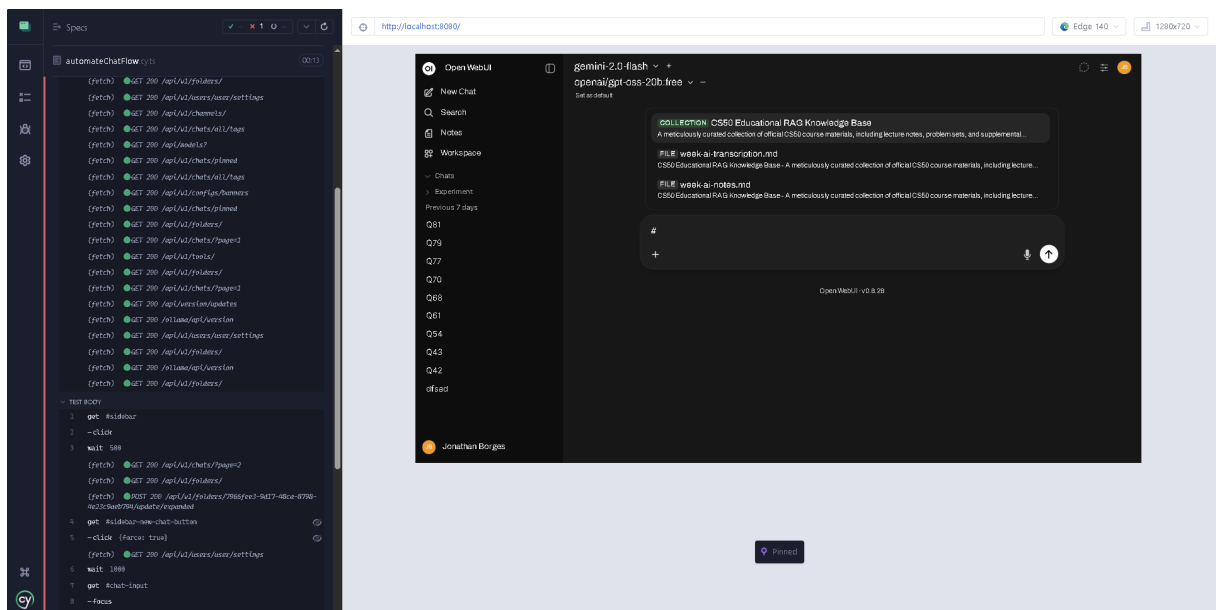


Figura 12 – Cypress executando o experimento

- ❑ **Configuração de Teste: Um Chat por Pergunta:** A decisão crucial foi submeter cada uma das 97 perguntas do dataset de avaliação (gerado na Seção 3.4) em um novo e isolado chat dentro do OpenWebUI. Esta abordagem garante um *rigor experimental* elevado, eliminando qualquer influência do histórico de conversação no LLM. Isso permite que a capacidade do chatbot de responder a cada pergunta seja

avaliada de forma independente, mitigando os efeitos da sensibilidade do prompt e da acumulação de contexto que podem distorcer os resultados em interações contínuas (SCHULHOFF et al., 2025).

- ❑ **Coleta de Dados Automatizada:** O script Cypress interage programaticamente com a interface do OpenWebUI para tarefas: (i) iniciar um novo chat, (ii) selecionar o LLM e o contexto RAG associado ao CS50, (iii) enviar a pergunta, (iv) aguardar a geração da resposta do LLM, e (v) exportar o Notação de Objeto JavaScript (JSON) completo da conversa. Cada arquivo JSON contém a pergunta do usuário, a resposta gerada pelo LLM, as fontes de recuperação (*chunks* do RAG) e metadados como *timestamps*, essenciais para análises futuras.

Este procedimento automatizado permite a coleta de um volume significativo de dados de interação de forma padronizada e reproduzível, formando a base para a subsequente avaliação qualitativa.

3.6.3 Metodologia de Avaliação Automática do RAG com Ragas

A avaliação do desempenho do chatbot pedagógico foi realizada por meio de uma metodologia quantitativa baseada no framework RAGAS (EXPLODINGGRADIENTS, 2024), que permite mensurar a qualidade das respostas geradas por sistemas baseados em Recuperação-Augmentada por Geração (RAG). O conjunto de dados de avaliação consistiu em 97 perguntas, para as quais as respostas dos modelos foram avaliadas segundo métricas automáticas de qualidade.

Foram utilizadas as seguintes métricas principais: **Factual Correctness** e **Semantic Similarity**. A métrica *NonLLMContextPrecisionWithReference* foi descartada devido à sua baixa sensibilidade e distribuição inadequada (predominância de valores nulos ou zero).

3.6.3.1 Factual Correctness

A métrica *Factual Correctness* avalia o alinhamento factual entre a resposta gerada (*response*) e a referência (*reference*). O cálculo é realizado em três etapas: (i) decomposição das sentenças em afirmações atômicas (*claims*), (ii) identificação do grau de sobreposição factual entre resposta e referência via inferência natural de linguagem, e (iii) cálculo das métricas de precisão, revocação e F1-score. O *score* final varia de 0 a 1, sendo 1 o alinhamento factual perfeito.

As fórmulas utilizadas são:

True Positive (TP) = Número de afirmações na resposta presentes na referência

False Positive (FP) = Número de afirmações na resposta ausentes na referência

False Negative (FN) = Número de afirmações na referência ausentes na resposta

$$\text{Precisão} = \frac{TP}{TP + FP}$$

$$\text{Revocação} = \frac{TP}{TP + FN}$$

$$\text{F1-score} = \frac{2 \times \text{Precisão} \times \text{Revocação}}{\text{Precisão} + \text{Revocação}}$$

Exemplo de uso (adaptado ⁵):

```

1 from ragas.dataset_schema import SingleTurnSample
2 from ragas.metrics._factual_correctness import FactualCorrectness
3
4 sample = SingleTurnSample(
5     response="The Eiffel Tower is located in Paris.",
6     reference="The Eiffel Tower is located in Paris. It has a height of 1000ft."
7 )
8
9 scorer = FactualCorrectness(llm = evaluator_llm)
10 await scorer.single_turn_ascore(sample)

```

Output:

0.67

O modo padrão é o F1-score, mas pode-se optar por precisão ou revocação ajustando o parâmetro `mode`.

3.6.3.2 Semantic Similarity

A métrica *Semantic Similarity* avalia a similaridade semântica entre a resposta gerada e a referência, utilizando modelos de embeddings bi-encoder e a similaridade do cosseno. O *score* também varia de 0 a 1, indicando maior alinhamento semântico para valores próximos de 1.

O cálculo segue os seguintes passos:

1. Vetorização da resposta e da referência usando o modelo de embeddings.
2. Cálculo da similaridade do cosseno entre os vetores.

⁵ <https://docs.ragas.io/en/v0.2.15/concepts/metrics/available_metrics/factual_correctness/>

Exemplo de uso (adaptado ⁶):

```
1 from ragas.dataset_schema import SingleTurnSample
2 from ragas.metrics import SemanticSimilarity
3 from ragas.embeddings import LangchainEmbeddingsWrapper
4
5 sample = SingleTurnSample(
6     response="The Eiffel Tower is located in Paris.",
7     reference="The Eiffel Tower is located in Paris. It has a height of 1000ft."
8 )
9
10 scorer =
11     SemanticSimilarity(embeddings=LangchainEmbeddingsWrapper(evaluator_embedding))
12
13 await scorer.single_turn_ascore(sample)
```

Output:

0.815

3.6.3.3 Configuração dos Parâmetros

A granularidade da decomposição factual pode ser ajustada via os parâmetros *atomicity* e *coverage*, permitindo controlar o nível de detalhamento e abrangência das afirmações extraídas, conforme a necessidade do estudo.

3.6.3.4 Implementação Experimental

As métricas foram aplicadas sobre as respostas dos modelos Gemini e GPT-OSS, permitindo a comparação quantitativa do desempenho factual e semântico. Os resultados foram analisados estatisticamente por meio de medidas descritivas, boxplots, scatterplots e testes de hipótese (t de Student e Mann-Whitney), conforme detalhado na Seção 4.4.2.

3.6.4 Considerações Éticas

A implementação de sistemas de Inteligência Artificial em contextos educacionais exige uma análise rigorosa das implicações éticas, mesmo em ambientes de demonstração local. Este trabalho aborda as seguintes considerações:

3.6.5 Escopo Ético e Público-Alvo do Sistema

O presente sistema foi concebido e avaliado para uso focado no ensino superior. A decisão de mirar nível de ensino deve-se às complexas nuances éticas, pedagógicas e regulatórias envolvidas no desenvolvimento de sistemas de IA com acesso irrestrito para

⁶ <https://docs.ragas.io/en/v0.2.15/concepts/metrics/available_metrics/semantic_similarity/>

estudantes do ensino fundamental e médio. Tais considerações, que demandariam estudos aprofundados sobre proteção de dados de menores, vieses de aprendizado e responsabilidade parental, não foram abrangidas por esta pesquisa, delimitando o escopo ético e prático do trabalho.

3.6.5.1 Privacidade e Soberania de Dados

A escolha de um ambiente de execução local para o chatbot, utilizando ferramentas *open source* como Ollama e OpenWebUI, mitiga significativamente os riscos associados à privacidade e soberania dos dados. (...)

3.7 Considerações Éticas

A implementação de sistemas de Inteligência Artificial em contextos educacionais exige uma análise rigorosa das implicações éticas, mesmo em ambientes de demonstração local. Este trabalho aborda as seguintes considerações:

O presente sistema foi concebido e avaliado para uso no ensino superior. A decisão de focar neste nível de ensino deve-se às complexas nuances éticas, pedagógicas e regulatórias (como a LGPD) envolvidas no desenvolvimento de sistemas de IA com acesso irrestrito para estudantes do ensino fundamental e médio. Tais considerações, que demandariam estudos aprofundados sobre proteção de dados de menores, vieses de aprendizado e responsabilidade parental, não foram abrangidas por esta pesquisa.

3.7.1 Privacidade e Soberania de Dados

A escolha de um ambiente de execução local para o chatbot, utilizando ferramentas *open source* como Ollama e OpenWebUI, mitiga significativamente os riscos associados à privacidade e soberania dos dados. Diferentemente de soluções baseadas em APIs de nuvem, onde os dados de interação podem transitar por servidores de terceiros, a execução *on-premise* garante que todas as interações e informações geradas permaneçam sob o controle do usuário/instituição. O dataset do CS50 é público, mas esta abordagem tecnológica estabelece um precedente para lidar com dados sensíveis de forma segura no futuro.

3.7.2 Transparência e Reprodutibilidade

A utilização predominante de componentes *open source* e a disponibilização dos scripts e do dataset curado reforçam o compromisso com a transparência e a reprodutibilidade da pesquisa. Isso permite que outros pesquisadores auditem, repliquem e estendam o trabalho, contribuindo para o avanço coletivo da IA na educação (MANCHANDA et al., 2025).

3.7.3 Mitigação do Risco de Dependência: Design e Governança

Este estudo reconhece o risco de que ferramentas de IA, se mal projetadas, possam promover uma dependência prejudicial nos alunos, minando o desenvolvimento do pensamento crítico (SCHULHOFF et al., 2025). Para endereçar essa preocupação, a abordagem deste trabalho se baseia em duas frentes: o design pedagógico do agente e a governança da plataforma.

Primeiramente, o prompt de sistema do chatbot (detalhado na Seção 3.6.1) foi cuidadosamente projetado para instruir o agente guiando o aluno com respostas pedagógicas simulando um professor.

Em segundo lugar, e de forma crucial, a escolha de uma plataforma como a OpenWebUI é estratégica, pois suas funcionalidades de controle de acesso e histórico de conversas (discutidas na Seção 3.2.3.1) capacitam os professores. Em uma aplicação real, eles poderiam auditar as interações para oferecer instrução personalizada e garantir que o chatbot seja uma ferramenta de apoio, e não uma que possa inibir o aprendizado com dependência cognitiva. A tecnologia, nesse contexto, serve para ampliar a capacidade de observação e intervenção do educador.

Contudo, é imperativo destacar que a presente pesquisa se concentrou em um experimento de ambiente controlado. A validação empírica destes benefícios pedagógicos e da dinâmica entre professor, aluno e IA não foi realizada com usuários reais e constitui uma direção clara para trabalhos futuros.

3.7.4 Vieses em Modelos *Open Source* e Curadoria Pedagógica

Mesmo modelos *open source* podem conter vieses herdados de seus dados de treinamento (MANCHANDA et al., 2025). O design do *System Prompt* e a metodologia de avaliação qualitativa visam identificar e mitigar respostas potencialmente enviesadas ou inadequadas. A curadoria humana dos dados sintéticos de QA, bem como a avaliação qualitativa das interações do chatbot por um especialista, atuam como camadas de controle para garantir que o comportamento do chatbot seja ético, justo e alinhado aos objetivos educacionais.

Resultados e Discussão

4.1 Análise Qualitativa de Exemplo: A Importância do RAG

Antes de apresentar a análise quantitativa agregada, é instrutivo analisar um exemplo concreto extraído do resultado final. A Figura 13 ilustra o fluxo de processamento para uma única pergunta, comparando a resposta de referência (gabarito) com as respostas geradas por duas configurações de modelo distintas: o GPT-OSS operando com o mecanismo RAG e o Gemini 2.5 Flash¹ operando em modo “puro” (sem acesso à base de conhecimento). Este exemplo visual destaca de forma clara a eficácia do RAG na prevenção de alucinações e na garantia da fidelidade factual.

Como observado na Figura 13, o modelo GPT-OSS com RAG foi capaz de extrair a informação correta dos contextos recuperados e formular uma resposta com 100% de correção factual. Em contrapartida, o modelo Gemini 2.5 Flash, sem o auxílio do RAG, falhou em responder à pergunta e gerou um conteúdo completamente não relacionado, caracterizando uma alucinação. Este caso particular exemplifica a principal hipótese deste trabalho: a utilização do RAG é um componente crítico para a confiabilidade de chatbots em domínios de conhecimento específicos.

4.2 Caracterização do Dataset de Avaliação

Após os processos de deduplicação e amostragem estratificada, o dataset final para avaliação foi consolidado em 120 pares de pergunta-resposta. Uma análise descritiva foi conduzida para verificar a distribuição das variáveis e confirmar a eficácia da estratégia de amostragem. O objetivo era garantir um conjunto de dados balanceado e representativo para uma avaliação imparcial dos modelos de RAG.

¹ O uso do modelo Gemini 2.5 Flash é pontual e serve a um propósito ilustrativo: demonstrar que mesmo um modelo de última geração pode alucinar sem o auxílio do RAG. Este modelo específico não foi utilizado nas avaliações quantitativas do estudo, que se concentraram no Gemini 2.0 Flash.

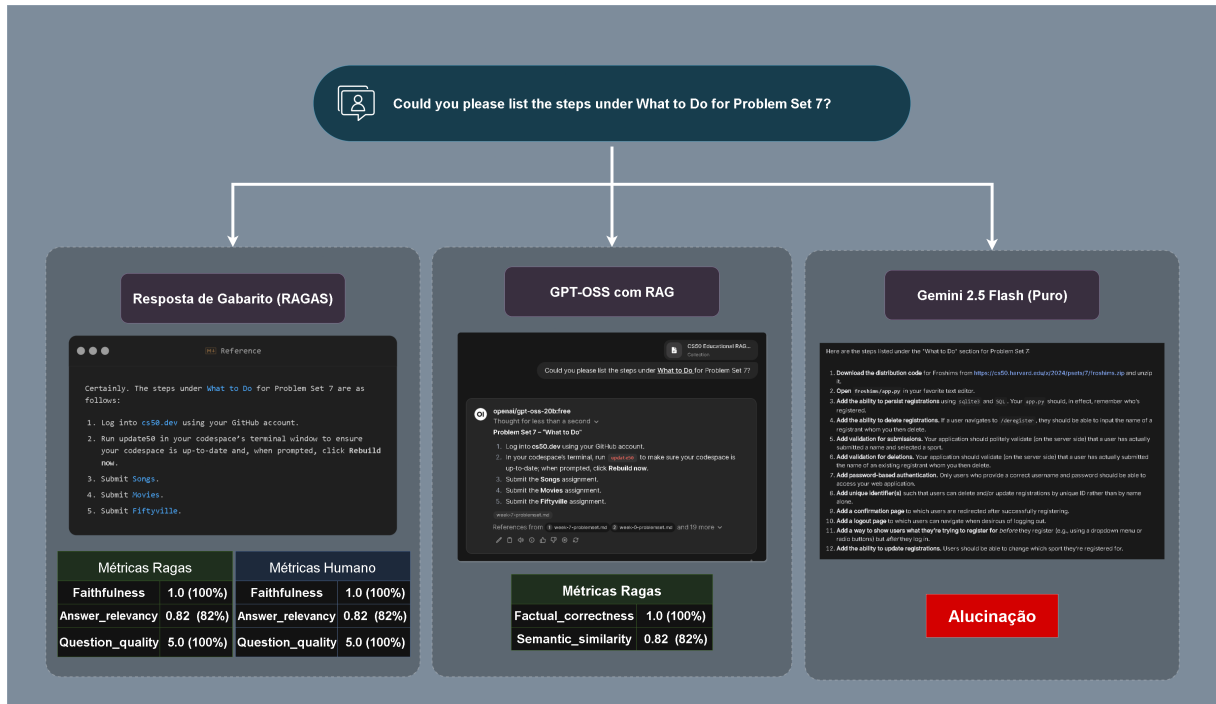


Figura 13 – Estudo de caso visual para uma amostra do *dataset* de avaliação. A figura exhibe: (a) a resposta de referência do gabarito e suas métricas de qualidade, (b) a resposta factualmente correta do GPT-OSS com RAG (*Factual Correctness* de 100%), e (c) a resposta do Gemini 2.5 Flash sem RAG, que demonstra o fenômeno da alucinação ao gerar informações incorretas e não relacionadas ao contexto solicitado.

Conforme planejado na metodologia, o *dataset* final apresenta uma distribuição uniforme entre as semanas do curso, com exatamente 10 perguntas por semana. A Tabela 3 sumariza a distribuição das principais variáveis categóricas.

A análise da Tabela 3 revela que, embora a amostragem tenha buscado o equilíbrio, algumas características se destacam. A maioria das perguntas (71.7%) foi gerada a partir de *chunks* (fragmentos de texto), em detrimento de documentos inteiros, refletindo um cenário de recuperação de informação mais granular. A distribuição entre as personas da Taxonomia de Bloom e os estilos de pergunta é relativamente equilibrada, sem que nenhuma categoria domine excessivamente. Isso garante que o desempenho dos modelos seja testado em uma gama diversificada de complexidades cognitivas e variações linguísticas.

Por fim, é importante notar que o processo de supervisão humana e avaliação de qualidade, descrito na Seção 4.3, resultou na exclusão de 23 amostras, levando ao conjunto final de 97 perguntas efetivamente utilizadas para a comparação de desempenho entre os modelos Gemini 2.0 Flash e GPT-OSS. A caracterização apresentada aqui refere-se ao conjunto de 120 perguntas pré-anotação no Label-Studio, que formou a base para todo o processo de avaliação.

Tabela 3 – Distribuição das Variáveis no Dataset de Avaliação Estratificado (N=120)

Variável	Categoria	Percentual (%)
Tipo de Contexto	Chunk	71.7
	Documento	28.3
Persona (Bloom)	Practical Implementer	25.8
	Novice Recaller	18.3
	Critical Judge	16.7
	Innovative Synthesizer	15.8
	Conceptual Comprehender	12.5
	Analytical Deconstructor	10.8
Estilo da Pergunta	Com Erros Ortográficos	26.7
	Gramática Perfeita	25.8
	Gramática Pobre	24.2
	Similar a Busca Web	23.3
Tamanho da Pergunta	Longa	37.5
	Curta	31.7
	Média	30.8

4.3 Processamento e Análise de Dados de Avaliação Humana

4.3.1 Metodologia de Processamento

O pipeline de processamento implementado para análise dos dados de avaliação humana segue uma abordagem sistemática e cientificamente rigorosa, priorizando a avaliação humana como *ground truth* enquanto utiliza métricas automáticas do RAGAS como ferramenta de análise comparativa. A metodologia adotada incorpora três critérios de exclusão distintos, implementados de forma hierárquica para garantir a qualidade do *dataset* final.

4.3.1.1 Critérios de Exclusão

Os critérios de exclusão foram estabelecidos com base em padrões empíricos e revisão da literatura, alinhados com o padrão de qualidade de 60% (equivalente à escala Likert 3/5):

1. **Exclusão Explícita Humana:** Amostras marcadas manualmente para remoção durante a anotação no Label Studio, representando o critério primário de exclusão.
2. **Exclusão Automática RAGAS:** Amostras com scores RAGAS abaixo do threshold de 60% de qualidade. **Nota:** Este critério foi desabilitado para priorizar a avaliação humana, mantendo amostras com scores RAGAS baixos mas scores humanos elevados.

3. **Exclusão Baseada em Scores Humanos:** Mecanismo de segurança para capturar casos onde avaliadores humanos atribuíram scores baixos (≤ 3) mas não marcaram explicitamente para exclusão.

Os thresholds específicos implementados foram:

- ❑ **Faithfulness RAGAS:** ≤ 0.6 (60% do range 0-1)
- ❑ **Answer Relevancy RAGAS:** ≤ 0.6 (60% do range 0-1)
- ❑ **Question Quality RAGAS:** ≤ 3 (60% da escala Likert 1-5)
- ❑ **Scores Humanos:** ≤ 3 (60% da escala Likert 1-5)

4.3.1.2 Processamento de Dados

O pipeline de processamento implementa validação estatística robusta, incluindo verificação de valores ausentes, análise de distribuição, conversão explícita de tipos de dados, *parsing* das respostas Likert do Label Studio e análise de correlação.

4.3.2 Resultados Estatísticos

4.3.2.1 Validação Estatística dos Dados

A validação estatística revelou alta qualidade dos dados, com apenas 0.8% de valores ausentes na métrica de faithfulness:

Tabela 4 – Validação Estatística dos Dados de Avaliação

Métrica	n	μ	σ	Valores Ausentes	Min	Max
RAGAS Faithfulness	119	0.953	0.141	1 (0.8%)	0.500	1.000
RAGAS Answer Relevancy	120	0.799	0.173	0	0.471	1.000
RAGAS Question Quality	120	4.950	0.339	0	3.000	5.000
Human Faithfulness	120	4.567	0.914	0	2.000	5.000
Human Response Relevancy	120	4.367	0.961	0	1.000	5.000
Human Question Quality	120	4.542	0.952	0	1.000	5.000

μ : Média; σ : Desvio Padrão.

4.3.2.2 Análise de Exclusão

A aplicação dos critérios de exclusão resultou em uma redução controlada do dataset, mantendo a integridade científica:

- ❑ **Amostras Originais:** 120
- ❑ **Excluídas Explícitas pelo Humano:** 23 (19.2%)

- ❑ **Scores RAGAS Baixos ($\leq 60\%$) mas Mantidos:** 4 (3.3%)
- ❑ **Total Excluído:** 23 (19.2%)
- ❑ **Dataset Final Limpo:** 97 amostras
- ❑ **Taxa de Retenção:** 80.8%

4.3.3 Análise de Correlação

4.3.3.1 Correlação Spearman: RAGAS vs Avaliação Humana

A análise de correlação Spearman revelou associações estatisticamente significativas entre todas as métricas RAGAS e humanas ($p < 0.001$), demonstrando que o RAGAS captura uma porção substancial da variância nas avaliações humanas:

Tabela 5 – Correlações Spearman entre Métricas RAGAS e Humanas

Métrica	r	p-valor	n
Faithfulness	0.447	< 0.001	119
Response Relevancy	0.434	< 0.001	120
Question Quality	0.323	< 0.001	120

4.3.3.2 Interpretação das Correlações

A avaliação humana, realizada por especialista, demonstrou ser particularmente exigente, contribuindo para que as correlações observadas entre as métricas RAGAS e as avaliações humanas fossem de intensidade moderada a fraca.

Faithfulness ($r = 0.447$): Correlação moderada, indicando que o RAGAS captura aproximadamente 20% da variância nas avaliações humanas de fidelidade. Embora o RAGAS seja capaz de identificar tendências gerais de fidedignidade, sua pontuação deve ser interpretada com cautela e complementada por avaliação humana. A divergência, conforme a análise dos autores, pode ser atribuída a *samples* específicas onde o contexto fornecido ao LLM foi excessivamente curto, o que prejudicou tanto a geração da pergunta quanto a resposta, criando nuances difíceis de serem avaliadas consistentemente, mesmo por especialista humano. No entanto, os autores observam que, apesar dessa correlação, uma parte significativa dos diagnósticos do RAGAS alinhou-se com o julgamento humano.

Response Relevancy ($r = 0.434$): Correlação moderada similar, demonstrando capacidade satisfatória de avaliação de relevância. As diferenças podem surgir da subjetividade na percepção da “adequação” da resposta a uma pergunta, que um modelo pode interpretar de forma diferente de um ser humano com conhecimento pedagógico, especialmente em contextos onde o contexto de origem era limitado.

Question Quality ($r = 0.323$): Correlação mais fraca, possivelmente devido à concentração de scores RAGAS no valor máximo (5.0), reduzindo a discriminabilidade. A baixa correlação para a qualidade da pergunta é particularmente notável. Os autores ressaltam que, como *Question Quality* foi uma métrica customizada e desenvolvida para este trabalho, sua metodologia e o *prompt* carecem de maior evolução para capturar as subjetividades da avaliação humana. A concentração dos scores RAGAS em valores altos sugere que o LLM avaliador do RAGAS tende a superestimar a qualidade das perguntas ou tem dificuldade em identificar nuances de clareza e relevância que um humano percebe, limitando sua utilidade para refinar a qualidade de perguntas sintéticas sem supervisão humana robusta.

4.3.4 Análise de Contingência

4.3.4.1 Tabela de Contingência: Exclusão Humana vs Scores RAGAS Baixos

A análise de contingência revelou associação estatisticamente significativa entre exclusão humana e detecção de problemas pelo RAGAS ($\chi^2 = 28.592$, $p < 0.001$, $df = 1$):

Tabela 6 – Tabela de Contingência: Exclusão Humana vs Detecção RAGAS

Excluída pelo Humano	RAGAS: Métrica Abaixo do Threshold	
	False	True
False	93	4
True	12	11

4.3.4.2 Análise de Performance do RAGAS

- ❑ **Amostras Não Excluídas:** 97 (79.2%)
- ❑ **Amostras Excluídas:** 23 (19.2%)
- ❑ **Falsos Negativos RAGAS:** 12 amostras (RAGAS não detectou problemas identificados por humanos)
- ❑ **Detecções Corretas RAGAS:** 11 amostras (RAGAS detectou problemas também identificados por humanos)
- ❑ **Taxa de Acordo RAGAS:** 47.8% (11/23)

4.3.5 Visualizações e Análise Gráfica

A Figura 14 apresenta três gráficos de dispersão comparando scores RAGAS e humanos para cada métrica, incluindo linhas de regressão e correlação perfeita para referência. Os gráficos demonstram:

- ❑ **Faithfulness:** Distribuição positiva com dispersão moderada, indicando correlação consistente
- ❑ **Response Relevancy:** Padrão similar ao faithfulness, com alguns outliers notáveis
- ❑ **Question Quality:** Concentração de scores RAGAS no máximo (5.0), explicando a correlação mais fraca

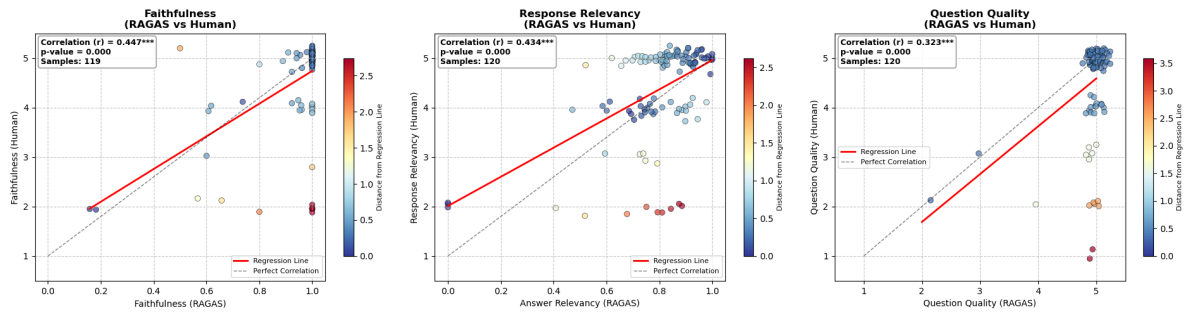


Figura 14 – Análise de Correlação: RAGAS vs Avaliação Humana. Cada subplot mostra a relação entre métricas RAGAS (eixo-x) e humanas (eixo-y), incluindo linha de regressão (vermelha), correlação perfeita (cinza tracejada) e coloração baseada na distância da linha de regressão.

A Figura 15 apresenta análise visual da concordância entre RAGAS e humanos, combinando:

- ❑ **Gráfico de Barras:** Distribuição de amostras por decisão de exclusão humana e detecção RAGAS
- ❑ **Matriz de Correlação:** Correlações intra-métricas e cross-métricas entre RAGAS e avaliação humana

4.3.6 Análise de Amostras com Baixa Correlação

A análise identificou amostras com desvios significativos da linha de regressão, fornecendo insights sobre limitações do RAGAS:

- ❑ **Faithfulness:** 5 amostras com desvios máximos, incluindo casos onde RAGAS atribuiu score 1.0 mas humanos avaliaram como 2
- ❑ **Response Relevancy:** 5 amostras com diferenças extremas, destacando desafios na avaliação de relevância
- ❑ **Question Quality:** 5 amostras com discrepâncias significativas, principalmente devido à concentração de scores RAGAS no máximo

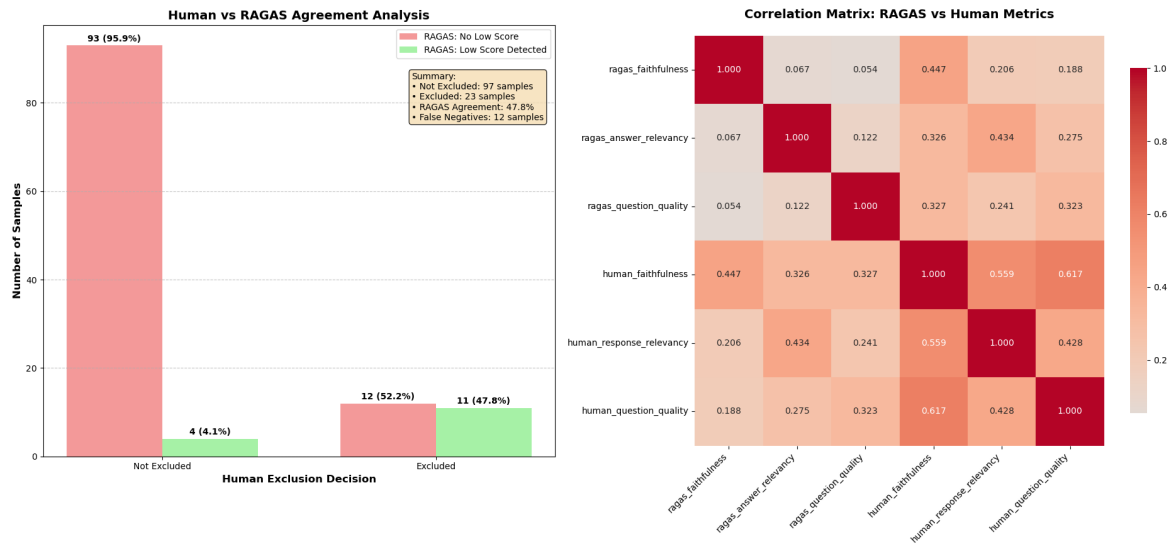


Figura 15 – Análise de Contingência e Matriz de Correlação. Esquerda: Análise de concordância entre exclusão humana e detecção RAGAS, mostrando distribuição de amostras e taxa de acordo. Direita: Matriz de correlação Spearman entre todas as métricas RAGAS e humanas.

A normalização das diferenças entre scores RAGAS e humanos revelou padrões consistentes de discordância, particularmente em casos onde o contexto fornecido era limitado ou ambíguo.

4.3.7 Implicações Científicas

A validação da metodologia e sua implementação de exclusão baseada prioritariamente na avaliação humana demonstrou ser cientificamente robusta, permitindo que o RAGAS funcione como ferramenta de análise sem comprometer a integridade do *dataset*.

Ademais, a performance do RAGAS como Ferramenta de Análise com taxa de acordo de 47.8%, o RAGAS demonstra capacidade moderada de identificar problemas de qualidade, funcionando como ferramenta satisfatória de *Quality Assurance* (QA) que pode complementar a avaliação humana.

Entretanto, as seguintes limitações foram identificadas:

- ❑ **Question Quality (Necessidade de Evolução Metodológica):** Observou-se uma tendência de superavaliação e baixa discriminabilidade, atribuindo *scores* máximos independentemente de nuances de qualidade real. Esta métrica, por ser uma criação do próprio estudo para um cenário específico, demanda aprimoramento no *prompt* e na metodologia de avaliação para futuros trabalhos, a fim de aumentar sua correlação e utilidade prática.
- ❑ **Contexto Limitado em Amostras Específicas:** Em algumas amostras, a limi-

tação do contexto fornecido ao LLM gerador resultou em perguntas e respostas que, embora factualmente corretas dentro do contexto restrito, pareceram estranhas ou incompletas para os avaliadores humanos. Este tipo de nuance subjetiva dificultou a avaliação consistente tanto pelo RAGAS quanto pelos humanos, contribuindo para a dispersão observada nas correlações.

- ❑ **Calibração de Escalas:** Identificaram-se diferenças sistemáticas entre escalas RAGAS (0 – 1) e humanas (1 – 5), exigindo mapeamento cuidadoso.
- ❑ **Tamanho da Amostra e Número de Avaliadores:** A avaliação humana foi realizada em um *dataset* final de 97 pares de perguntas e respostas por apenas um avaliador especializado. Embora a curadoria tenha sido rigorosa, esta configuração pode introduzir vieses individuais e limitar a generalização dos resultados. Futuros trabalhos deverão expandir o número de *samples* e incluir múltiplos avaliadores para maior robustez e confiabilidade estatística.

4.3.8 Conclusões

O pipeline de processamento implementado demonstrou robustez científica, priorizando corretamente a avaliação humana como *ground truth* enquanto utiliza o RAGAS como ferramenta analítica valiosa. As correlações moderadas, mas estatisticamente significativas, confirmam a utilidade do RAGAS como ferramenta de métricas automáticas. Apesar das correlações não serem elevadas, a taxa de acordo entre RAGAS e avaliadores humanos em casos de detecção de problemas foi significativa, e os autores consideram que o RAGAS performou bem no geral, especialmente dada a complexidade e a subjetividade de se avaliar tanto perguntas quanto respostas, uma abordagem ainda pouco explorada em profundidade na literatura. As limitações identificadas, como as amostras com contexto curto e a necessidade de evolução da métrica *Question Quality* como metodologia de avaliação para trabalhos futuros. A qualidade do *dataset* final (80.8% de retenção) foi assegurada pela metodologia de exclusão hierárquica.

4.4 Resultados da Avaliação do Chatbot

A análise do desempenho dos modelos Gemini-2.0-Flash e GPT-OSS-20B no contexto de Recuperação-Aumentada por Geração (RAG) foi conduzida com base em métricas quantitativas de qualidade. Os resultados, apresentados a seguir, incluem uma análise estatística descritiva, visualizações da distribuição de dados e testes de hipótese para determinar a significância das diferenças observadas.

4.4.1 Análise Estatística Descritiva e Visualização

A Tabela 7 apresenta as estatísticas descritivas para as métricas de Factual Correctness e Semantic Similarity de cada modelo, incluindo média, desvio padrão, min e max.

Tabela 7 – Estatísticas Descritivas das Métricas de Avaliação por Modelo

Modelo	Métrica	n	Média	Desvio Padrão	Min	Max
GPT	Factual Correctness	97	0.648	0.246	0.000	1.000
	Semantic Similarity	97	0.708	0.185	0.066	0.903
Gemini	Factual Correctness	97	0.691	0.220	0.000	1.000
	Semantic Similarity	97	0.814	0.109	0.106	0.974

A distribuição das pontuações para cada métrica é visualizada por meio de boxplots na Figura 16. O gráfico ilustra a mediana, os quartis e a dispersão das pontuações de ambos os modelos. Para a métrica *Factual Correctness*, a mediana do modelo Gemini (0.750) é ligeiramente superior à do modelo GPT (0.680), e a distribuição de suas pontuações é mais condensada, conforme indicado pelo menor desvio padrão (0.220 vs 0.246). No entanto, as caixas dos boxplots se sobrepõem significativamente. Já para a métrica *Semantic Similarity*, o boxplot do Gemini está visivelmente mais alto, com uma mediana (0.836) consideravelmente superior à do GPT (0.768) e uma distribuição mais coesa.

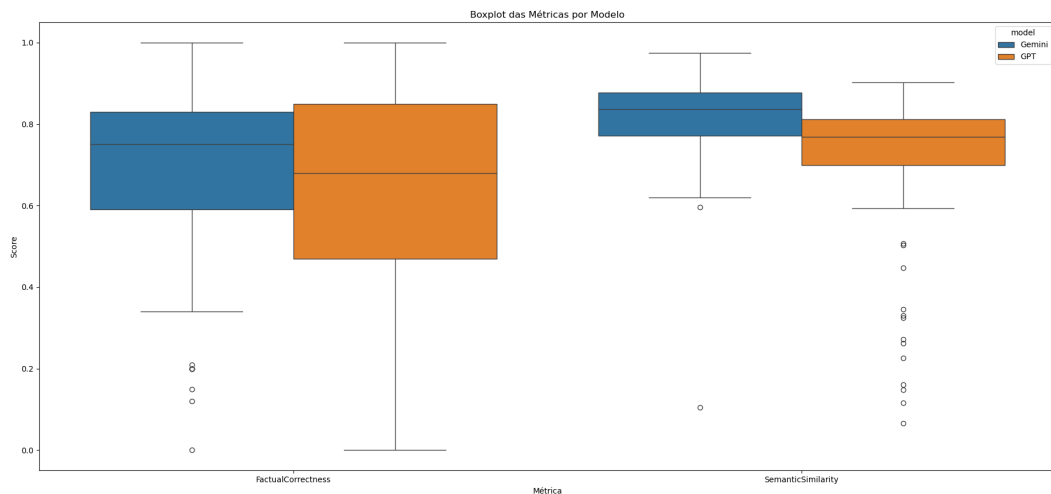


Figura 16 – Distribuição das pontuações para Factual Correctness e Semantic Similarity por modelo, visualizada por meio de boxplots.

A relação entre as duas métricas é explorada no scatterplot da Figura 17. Observa-se que, enquanto o desempenho em *Factual Correctness* apresenta ampla dispersão para ambos os modelos, os pontos do Gemini (em azul) tendem a se concentrar na porção superior do gráfico (alta *Semantic Similarity*). Em contraste, os pontos do GPT (em

laranja) demonstram maior variabilidade e uma concentração em pontuações de *Semantic Similarity* mais baixas, incluindo vários valores atípicos.

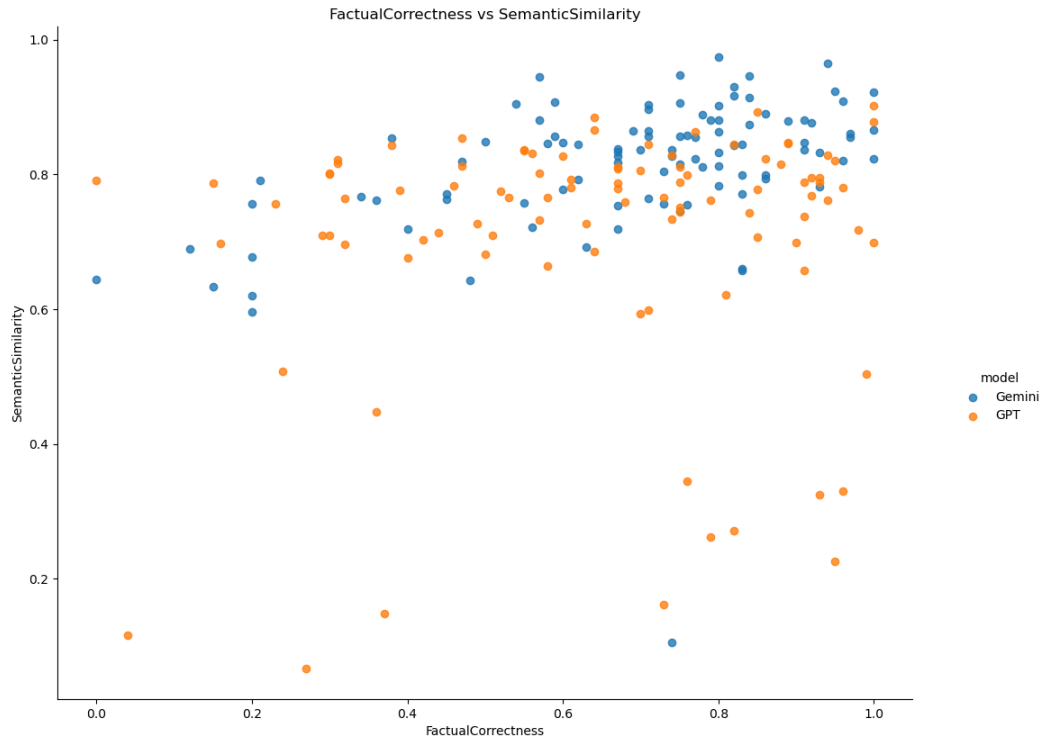


Figura 17 – Relação entre as pontuações de Factual Correctness e Semantic Similarity para os modelos Gemini e GPT.

4.4.2 Análise Estatística Inferencial

Para verificar se as diferenças observadas nas médias de cada métrica são estatisticamente significativas, foram realizados o teste t de Student para amostras independentes (considerando variâncias desiguais) e o teste de Mann-Whitney U (não-paramétrico). Os resultados são apresentados a seguir:

Métrica: Factual Correctness

❑ **Teste t de Student:** $t = 1.270, p = 0.2056$

❑ **Teste de Mann-Whitney U:** $U = 5163.000, p = 0.2413$

Ambos os testes retornaram um valor de $p > 0.05$. Consequentemente, não há evidências estatísticas para rejeitar a hipótese nula de que as médias dos modelos Gemini e GPT são iguais em relação à métrica *Factual Correctness*. As diferenças observadas podem ser atribuídas à variabilidade aleatória da amostra.

Métrica: Semantic Similarity

□ **Teste t de Student:** $t = 4.868, p < 0.0001$

□ **Teste de Mann-Whitney U:** $U = 6901.000, p < 0.0001$

Em contraste, os testes para a métrica *Semantic Similarity* resultaram em um valor de p baixo ($p < 0.0001$). Esse resultado indica uma diferença estatisticamente alta entre os modelos. A média de *Semantic Similarity* do modelo Gemini (0.814) é significativamente superior à do modelo GPT (0.708), o que sugere que o Gemini gera respostas que são semanticamente mais alinhadas com as referências, de forma consistente.

4.4.3 Discussão dos Resultados

A diferença de desempenho na métrica *Semantic Similarity* entre os modelos Gemini e GPT-OSS, em favor do primeiro, é um resultado significativo. Entretanto, é crucial considerar o viés metodológico introduzido pelo processo de geração do conjunto de dados de avaliação. As respostas de referência (gabarito) utilizadas no estudo foram geradas pelo próprio modelo Gemini 2.0-Flash. Esse alinhamento de modelos tende a favorecer o Gemini nas métricas de similaridade semântica, uma vez que a linguagem, a estrutura de frases e os vieses de resposta do modelo testado (Gemini) podem ser intrinsecamente mais semelhantes aos da referência gerada por ele mesmo, corroborando para uma pontuação mais alta. Esse achado ressalta a importância de se utilizar referências de gabarito geradas por um processo independente (e.g., curadoria humana) em futuras avaliações para garantir uma comparação mais imparcial e rigorosa entre os modelos. Adicionalmente, é fundamental destacar a diferença metodológica entre as métricas de avaliação. A métrica de *Semantic Similarity* depende exclusivamente de modelos de *embeddings*, que podem ser menos sensíveis a nuances contextuais. Em contraste, a *Factual Correctness* utiliza um LLM para inferir e validar a acurácia das afirmações, tornando-a a métrica mais rigorosa e confiável para avaliar a qualidade factual das respostas. A ausência de uma diferença estatisticamente significativa na *Factual Correctness* sugere que o modelo de código aberto (GPT-OSS), apesar de suas limitações, performou de maneira similar ao modelo de fronteira (Gemini) em sua capacidade de gerar respostas factualmente corretas. Este é o resultado mais importante da pesquisa, pois demonstra que modelos menores e acessíveis podem competir com modelos de grande escala em tarefas críticas de RAG, abrindo caminho para soluções mais eficientes e de menor custo.

Conclusão

A presente pesquisa teve como objetivo propor, desenvolver e avaliar uma arquitetura e metodologia baseada em ferramentas *open source* para a criação de chatbots pedagógicos com Recuperação Aumentada por Geração (RAG), visando sua aplicação em plataformas educacionais online de forma acessível, replicável e democrática. Este capítulo sumariza as principais descobertas, detalha as contribuições advindas do trabalho e propõe direções para futuras pesquisas.

5.1 Principais Contribuições

Este estudo contribui para o campo da Inteligência Artificial aplicada à educação, particularmente no que tange à democratização e acessibilidade de tecnologias avançadas:

1. Proposta de Arquitetura RAG *Open Source* para Chatbots Pedagógicos:

Foi implementado uma arquitetura modular e robusta que integra um orquestrador (*OpenWebUI*), um serviço de LLM local (*Ollama*), modelos de *embedding* (*Qwen3-Embedding-0.6B*) e uma base de conhecimento externa (*Vector DB*). Esta arquitetura demonstra a viabilidade de construir soluções de IA pedagógicas potentes com recursos limitados, desafiando a dependência de plataformas proprietárias e garantindo a soberania dos dados.

2. Metodologia para Geração de Dados Sintéticos de QA com Diversidade

Cognitiva: Estabeleceu-se um *pipeline* inovador para a geração de conjuntos de dados sintéticos de Perguntas e Respostas (QA), ancorado na Taxonomia de Bloom e utilizando LLMs para simular diferentes perfis cognitivos de alunos. Esta metodologia provou ser eficaz para criar dados sintéticos de boa qualidade, reproduzíveis e diversificados, essenciais para a avaliação rigorosa de sistemas RAG em contextos educacionais específicos, como o curso CS50.

3. **Engenharia de Prompt Pedagógico Fundamentada em Princípios Educacionais:** Foi concebido e implementado um *System Prompt* detalhado que incorpora o *Role-Playing* de um tutor experiente, instruindo respostas orientadores para perguntas. Esta abordagem visa estimular o raciocínio crítico, guiar a descoberta e promover o desenvolvimento contínuo de competências nos alunos, posicionando o *chatbot* como uma ferramenta complementar ao aprendizado.
4. **Validação Híbrida do Dataset Sintético de QA:** Para avaliar os modelos de forma rigorosa, foi desenvolvido um dataset de 97 pares de pergunta-resposta simulando dúvidas de alunos. A qualidade deste conjunto foi assegurada por uma metodologia de validação híbrida, que combinou métricas automáticas do *framework Ragas* com uma detalhada supervisão humana para avaliação qualitativa. Este processo resultou em um instrumento de avaliação confiável e contextualizado ao domínio educacional como do CS50.
5. **Avaliação Quantitativa da Performance de Chatbots RAG:** Utilizando o dataset validado, a performance de dois modelos foi comparada de forma automática e quantitativa, com foco na métrica de Factual Correctness do Ragas. O estudo revelou uma notável similaridade de desempenho factual entre um modelo *open source* (*gpt-oss-20b*) e um modelo proprietário (*Gemini-2.0-Flash*). Este resultado evidencia que soluções acessíveis e de código aberto podem competir eficazmente em tarefas críticas de RAG, validando a tese central do trabalho.
6. **Compromisso com a Transparência e Reprodutibilidade:** A escolha por ferramentas *open source* e a disponibilização pública da metodologia, scripts e dataset curado reforçam o compromisso com a ciência aberta. Esta postura permite que a comunidade acadêmica audite, replique e estenda o trabalho, fomentando o avanço coletivo da IA na educação de forma ética e colaborativa.

Em síntese, o desenvolvimento e a avaliação do chatbot pedagógico com RAG demonstraram com sucesso a viabilidade técnica e a eficácia pedagógica preliminar de uma abordagem baseada em ferramentas *open source*. A pesquisa conclui que a hipótese de que LLMs e SLMs *open source*, quando aumentados com RAG e guiados por engenharia de *prompt* pedagógica, podem oferecer suporte educacional de alta qualidade de forma acessível e controlável, foi validada pelos experimentos executados.

5.2 Trabalhos Futuros

A partir das descobertas e das limitações identificadas neste trabalho, diversas direções para futuras pesquisas emergem, visando expandir o escopo e aprofundar a eficácia dos chatbots pedagógicos:

1. **Expansão e Aprofundamento da Avaliação Humana:** Realizar estudos de usuário com uma amostra maior e mais diversificada de estudantes e educadores em ambientes de aprendizagem reais. Isso incluiria a avaliação do impacto direto na motivação, engajamento e resultados de aprendizagem, além da percepção sobre a utilidade e a qualidade do feedback.
2. **Investigação de Técnicas Avançadas de RAG:** Explorar a integração de técnicas de RAG mais sofisticadas, como *query rewriting*, recuperação híbrida (palavras-chave e vetorial), *re-ranking* de documentos recuperados e abordagens de RAG multi-estágio, para otimizar ainda mais a acurácia, relevância e a capacidade de lidar com contextos complexos.
3. **Exploração de Outros SLMs e Modelos de *Embedding Open Source*:** Testar e comparar o desempenho de outros Modelos de Linguagem Pequenos (SLMs) e modelos de *embedding open source* que surgem continuamente na comunidade, buscando otimizações de performance e eficiência computacional para hardware ainda mais restrito.
4. **Desenvolvimento de Mecanismos de Adaptação Dinâmica:** Investigar a adaptação dinâmica da persona do chatbot ou do estilo de *prompting* com base no perfil de aprendizado, histórico de interação ou nível de proficiência do aluno, visando um suporte pedagógico ainda mais personalizado.
5. **Estudos de Viés e Curadoria Pedagógica Contínua:** Implementar mecanismos robustos para detecção e mitigação de vieses nos modelos *open source* e na geração de respostas, além de desenvolver processos de curadoria pedagógica contínua para garantir a adequação cultural e ética do chatbot.
6. **Integração direta com Plataformas Educacionais:** Explorar a integração do chatbot diretamente com sistemas existentes, como Moodle, para facilitar a adoção e o gerenciamento por instituições de ensino.
7. **Comparativo com Ferramentas Comerciais de QA/RAG:** Realizar estudos comparativos aprofundados com plataformas comerciais ou em desenvolvimento, como o NotebookLM do Google, para analisar o desempenho, as funcionalidades e a experiência do usuário em diferentes contextos pedagógicos.

Essas direções futuras não apenas aprofundarão o conhecimento sobre a aplicação de IA na educação, mas também contribuirão para a construção de ferramentas mais inteligentes, acessíveis e alinhadas aos valores pedagógicos de um ensino de qualidade.

Referências

- ALI, F. et al. Supporting self-directed learning and self-assessment using teachergaia, a generative ai chatbot application: Learning approaches and prompt engineering. **Learning: Research and Practice**, v. 9, p. 1–13, 09 2023. Citado na página 35.
- B, G.; PURWAR, A. **Evaluating the Efficacy of Open-Source LLMs in Enterprise-Specific RAG Systems: A Comparative Study of Performance and Scalability**. 2024. Disponível em: <<https://arxiv.org/abs/2406.11424>>. Citado 3 vezes nas páginas 28, 29 e 42.
- BAI, J. et al. **Qwen Technical Report**. 2023. Disponível em: <<https://arxiv.org/abs/2309.16609>>. Citado na página 19.
- BARKER, M. et al. **Faster, Cheaper, Better: Multi-Objective Hyperparameter Optimization for LLM and RAG Systems**. 2025. Disponível em: <<https://arxiv.org/abs/2502.18635>>. Citado na página 66.
- BHAT, S. R. et al. **Rethinking Chunk Size For Long-Document Retrieval: A Multi-Dataset Analysis**. 2025. Disponível em: <<https://arxiv.org/abs/2505.21700>>. Citado na página 65.
- BLOOM, B. S. et al. **Taxonomy of Educational Objectives: The Classification of Educational Goals. Handbook I: Cognitive Domain**. New York: David McKay Company, 1956. Citado na página 50.
- BOONSTRA, L. **Prompt Engineering**. 2025. Whitepaper. Disponível em: <<https://www.leeboonstra.dev/writing/write-prompting-whitepaper/>>. Citado 4 vezes nas páginas 24, 25, 26 e 67.
- BROWN, T. B. et al. **Language Models are Few-Shot Learners**. 2020. Citado na página 18.
- CHANG, E. Y. **Prompting Large Language Models With the Socratic Method**. 2023. Citado na página 23.
- CHEN, J. et al. **WisdomBot: Tuning Large Language Models with Artificial Intelligence Knowledge**. 2025. Disponível em: <<https://arxiv.org/abs/2501.12877>>. Citado na página 35.

CHIANG, W.-L. et al. **Chatbot Arena: An Open Platform for Evaluating LLMs by Human Preference**. 2024. Citado na página 19.

CHURCH, K. et al. Emerging trends: a gentle introduction to rag. **Natural Language Engineering**, v. 30, p. 870–881, 09 2024. Citado 3 vezes nas páginas 27, 29 e 30.

COMANICI, G. et al. **Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities**. 2025. Disponível em: <<https://arxiv.org/abs/2507.06261>>. Citado 3 vezes nas páginas 21, 44 e 50.

DEEPSEEK-AI et al. **DeepSeek LLM: Scaling Open-Source Language Models with Longtermism**. 2024. Disponível em: <<https://arxiv.org/abs/2401.02954>>. Citado 2 vezes nas páginas 19 e 20.

DEVLIN, J. et al. **BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding**. 2019. Citado 2 vezes nas páginas 14 e 18.

Digital Education Council. **AI or Not AI: What Students Want. Digital Education Council Global AI Student Survey 2024**. [S.l.], 2024. Acessado em [Data de acesso]. Disponível em: <<https://www.digitaleducationcouncil.com/post/digital-education-council-global-ai-student-survey-2024>>. Citado 3 vezes nas páginas 14, 26 e 27.

EXPLODINGGRADIENTS. **Ragas: Supercharge Your LLM Application Evaluations**. 2024. <<https://github.com/explodinggradients/ragas>>. Citado 3 vezes nas páginas 50, 58 e 69.

FALCÃO, T. P. et al. Tutoria: a software platform to improve feedback in education. **Journal on Interactive Systems**, v. 14, n. 1, p. 383–393, Aug. 2023. Disponível em: <<https://journals-sol.sbc.org.br/index.php/jis/article/view/3247>>. Citado 2 vezes nas páginas 15 e 26.

Firecrawl contributors. **Firecrawl: The Web Data API for AI**. 2025. Último acesso: 2025-01-16. Disponível em: <<https://github.com/firecrawl/firecrawl>>. Citado na página 47.

GAO, Y. et al. **Retrieval-Augmented Generation for Large Language Models: A Survey**. 2024. Disponível em: <<https://arxiv.org/abs/2312.10997>>. Citado 3 vezes nas páginas 27, 28 e 30.

Gemini Team. **Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context**. 2024. Citado 2 vezes nas páginas 18 e 19.

Google LLC. **Google AI Studio and Generative Language API**. 2025. API de modelos generativos disponível em <https://generativelanguage.googleapis.com/v1beta/models/>. Disponível em: <<https://aistudio.google.com/>>. Citado 2 vezes nas páginas 44 e 50.

HICKE, Y. et al. **AI-TA: Towards an Intelligent Question-Answer Teaching Assistant using Open-Source LLMs**. 2023. Disponível em: <<https://arxiv.org/abs/2311.02775>>. Citado 2 vezes nas páginas 33 e 34.

- JIANG, A. Q. et al. **Mistral 7B**. 2023. Disponível em: <<https://arxiv.org/abs/2310.06825>>. Citado na página 19.
- JOHNSON, J.; DOUZE, M.; JéGOU, H. **Billion-scale similarity search with GPUs**. 2017. Disponível em: <<https://arxiv.org/abs/1702.08734>>. Citado na página 29.
- LENG, Q. et al. **Long Context RAG Performance of Large Language Models**. 2024. Disponível em: <<https://arxiv.org/abs/2411.03538>>. Citado na página 65.
- LEWIS, P. et al. **Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks**. 2021. Disponível em: <<https://arxiv.org/abs/2005.11401>>. Citado 5 vezes nas páginas 27, 29, 30, 31 e 67.
- LIU, R. et al. Teaching cs50 with ai: Leveraging generative artificial intelligence in computer science education. In: **Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1**. New York, NY, USA: Association for Computing Machinery, 2024. (SIGCSE 2024), p. 750–756. ISBN 9798400704239. Disponível em: <<https://doi.org/10.1145/3626252.3630938>>. Citado 2 vezes nas páginas 34 e 35.
- LIU, S. et al. Can small language models with retrieval-augmented generation replace large language models when learning computer science? In: **Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1**. New York, NY, USA: Association for Computing Machinery, 2024. (ITiCSE 2024), p. 388–393. ISBN 9798400706004. Disponível em: <<https://doi.org/10.1145/3649217.3653554>>. Citado 2 vezes nas páginas 36 e 37.
- LIVATHINOS, N. et al. **Docling: An Efficient Open-Source Toolkit for AI-driven Document Conversion**. 2025. Disponível em: <<https://arxiv.org/abs/2501.17887>>. Citado na página 66.
- LO, L. S. The clear path: A framework for enhancing information literacy through prompt engineering. **The Journal of Academic Librarianship**, v. 49, n. 4, p. 102720, 2023. ISSN 0099-1333. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0099133323000599>>. Citado 4 vezes nas páginas 24, 26, 65 e 67.
- LU, Z. et al. **Small Language Models: Survey, Measurements, and Insights**. 2025. Disponível em: <<https://arxiv.org/abs/2409.15790>>. Citado 4 vezes nas páginas 20, 41, 44 e 64.
- MAITY, S.; DERROY, A. **The Future of Learning in the Age of Generative AI: Automated Question Generation and Assessment with Large Language Models**. 2024. Disponível em: <<https://arxiv.org/abs/2410.09576>>. Citado 4 vezes nas páginas 15, 21, 26 e 27.
- MALAN, D. J.; UNIVERSITY, H. **CS50x: Introduction to Computer Science**. 2024. Acesso em: 15 jan. 2025. Disponível em: <<https://cs50.harvard.edu/x/2024/>>. Citado 4 vezes nas páginas 28, 38, 46 e 47.
- MANCHANDA, J. et al. **The Open Source Advantage in Large Language Models (LLMs)**. 2025. Disponível em: <<https://arxiv.org/abs/2412.12004>>. Citado 4 vezes nas páginas 19, 42, 72 e 73.

- MEYER, J. et al. Using llms to bring evidence-based feedback into the classroom: Ai-generated feedback increases secondary students' text revision, motivation, and positive emotions. **Computers and Education: Artificial Intelligence**, v. 6, p. 100199, 2024. ISSN 2666-920X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2666920X23000784>>. Citado 5 vezes nas páginas 14, 15, 21, 26 e 27.
- NEUMANN, A. T. et al. An llm-driven chatbot in higher education for databases and information systems. **IEEE Trans. on Educ.**, IEEE Press, v. 68, n. 1, p. 103–116, fev. 2025. ISSN 0018-9359. Disponível em: <<https://doi.org/10.1109/TE.2024.3467912>>. Citado 2 vezes nas páginas 32 e 33.
- OPENAI et al. **gpt-oss-120b and gpt-oss-20b Model Card**. 2025. Disponível em: <<https://arxiv.org/abs/2508.10925>>. Citado 3 vezes nas páginas 22, 44 e 64.
- OPENAI. Gpt-4 technical report. In: . [s.n.], 2023. Disponível em: <<https://api.semanticscholar.org/CorpusID:257532815>>. Citado 2 vezes nas páginas 14 e 18.
- OpenRouter Team. **OpenRouter: Unified API for AI Models with GPT-5 and Gemini Integration**. 2025. Plataforma de API unificada para acesso a múltiplos modelos de IA, incluindo GPT-5 e Gemini. Disponível em: <<https://openrouter.ai/>>. Citado na página 44.
- REIMERS, N.; GUREVYCH, I. **Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks**. 2019. Disponível em: <<https://arxiv.org/abs/1908.10084>>. Citado na página 29.
- RODRIGUES, L. et al. Assessing the quality of automatic-generated short answers using gpt-4. **Computers and Education: Artificial Intelligence**, v. 7, p. 100248, 2024. ISSN 2666-920X. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2666920X24000511>>. Citado 3 vezes nas páginas 16, 18 e 26.
- SAHOO, P. et al. **A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications**. 2025. Disponível em: <<https://arxiv.org/abs/2402.07927>>. Citado 3 vezes nas páginas 24, 25 e 26.
- SCHULHOFF, S. et al. **The Prompt Report: A Systematic Survey of Prompt Engineering Techniques**. 2025. Disponível em: <<https://arxiv.org/abs/2406.06608>>. Citado 6 vezes nas páginas 24, 25, 26, 67, 69 e 73.
- SKINNER, B. **Verbal Behavior**. Cambridge, MA: Copley Publishing Group, 1957. 1 p. Original quote: "Men act upon the world, and change it, and are changed in turn by the consequences of their action.". Citado na página 14.
- SWACHA, J.; GRACEL, M. Retrieval-augmented generation (rag) chatbots for education: A survey of applications. **Applied Sciences**, v. 15, p. 4234, 04 2025. Citado 2 vezes nas páginas 31 e 32.
- TEAM, Q. **Qwen3 Technical Report**. 2025. Disponível em: <<https://arxiv.org/abs/2505.09388>>. Citado na página 64.
- THWAY, M. et al. **Battling Botpoop using GenAI for Higher Education: A Study of a Retrieval Augmented Generation Chatbots Impact on Learning**. 2024. Disponível em: <<https://arxiv.org/abs/2406.07796>>. Citado na página 33.

- TKACHENKO, M. et al. **Label Studio: Data labeling software**. 2020–2025. Open source software available from <https://github.com/HumanSignal/label-studio>. Disponível em: <<https://github.com/HumanSignal/label-studio>>. Citado 2 vezes nas páginas 55 e 61.
- TOUVRON, H. et al. **LLaMA: Open and Efficient Foundation Language Models**. 2023. Disponível em: <<https://arxiv.org/abs/2302.13971>>. Citado na página 19.
- VASWANI, A. et al. **Attention Is All You Need**. 2023. Disponível em: <<https://arxiv.org/abs/1706.03762>>. Citado na página 18.
- YANG, Z. et al. The dawn of lmms: Preliminary explorations with gpt-4v(ision). **ArXiv**, abs/2309.17421, 2023. Citado na página 19.
- YAO, S. et al. **ReAct: Synergizing Reasoning and Acting in Language Models**. 2023. Disponível em: <<https://arxiv.org/abs/2210.03629>>. Citado na página 25.
- ZESCH, T. et al. **FernUni LLM Experimental Infrastructure (FLEXI) – Enabling Experimentation and Innovation in Higher Education Through Access to Open Large Language Models**. 2024. Disponível em: <<https://arxiv.org/abs/2407.13013>>. Citado 2 vezes nas páginas 36 e 37.
- ZHANG, Y. et al. **Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models**. 2025. Disponível em: <<https://arxiv.org/abs/2506.05176>>. Citado 3 vezes nas páginas 22, 44 e 64.
- ZHAO, C. Q. et al. **SlideItRight: Using AI to Find Relevant Slides and Provide Feedback for Open-Ended Questions**. 2025. Disponível em: <<https://arxiv.org/abs/2505.04584>>. Citado na página 33.
- ZHENG, C. et al. **Progressive-Hint Prompting Improves Reasoning in Large Language Models**. 2023. Citado na página 23.

Apêndices

Repositórios da Pesquisa

Para garantir a reprodutibilidade e permitir que a comunidade de pesquisa utilize os artefatos gerados neste trabalho, todos os códigos e dados foram disponibilizados em repositórios públicos, conforme detalhado na Tabela 8.

Tabela 8 – Links e descrições dos repositórios do projeto.

Recurso	Descrição	Link
Repositório Principal	Contém todo o código principal: geração de QA, análises e a execução dos experimentos com o chatbot RAG.	< https://github.com/dev-jonathan/ScholarRAG >
Coleta de Dados	Contém os scripts específicos para web scraping, parsing e formatação do conteúdo bruto do curso CS50 em arquivos Markdown e csv.	< https://github.com/dev-jonathan/cs50-course-dataset >
Dataset Publicado	Repositório no Hugging Face contendo os datasets finais e curados, prontos para uso, incluindo o corpus do curso e o conjunto de avaliação.	< https://huggingface.co/datasets/dev-jonathanb/cs50-educational-rag >