UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

ANDERSON CAPORALE ANAN

Desenvolvimento de um Chatbot Acadêmico Baseado em Small Language Models e Retrieval-Augmented Generation para o Instituto de Informática da UFRGS

Monografia apresentada como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação

Orientador: Prof. Dr. Dennis Giovani Balreira

Porto Alegre
2026

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Reitora: Profª. Marcia Cristina Bernardes Barbosa
Vice-Reitor: Prof. Pedro de Almeida Costa
Pró-Reitora de Graduação: Profª. Nádya Pesce da Silveira
Diretor do Instituto de Informática: Prof. Luciano Paschoal Gaspary
Coordenador do Curso de Ciência de Computação: Prof. Álvaro Freitas Moreira
Bibliotecário-chefe do Instituto de Informática: Alexsander Borges Ribeiro

AGRADECIMENTOS

Gostaria de agradecer a todos que, de alguma forma, contribuíram para que eu chegasse até aqui, especialmente às seguintes pessoas:

À minha mãe, Inês Caporale, por todo o apoio, incentivo e confiança ao longo da minha trajetória acadêmica.

Ao meu pai, Lyoji Anan, por sempre me incentivar a buscar conhecimento, valorizar os estudos e investir no meu desenvolvimento pessoal e profissional. Estendo também meus agradecimentos à sua companheira, Isa Ferreira, por compartilhar e fortalecer esse apoio durante essa caminhada.

À Nicole, minha namorada e companheira, por estar sempre ao meu lado, me apoiar nos momentos difíceis, me incentivar a ser uma pessoa melhor e por servir de exemplo com sua dedicação e comprometimento.

Aos meus colegas e amigos, Alberto, Bruno, Clayton, Haroldo, Henrique, Lucas Feksa e Lucas Schirmbeck, por tornarem essa jornada mais leve, agradável e memorável, compartilhando aprendizados, desafios, conquistas e bons momentos ao longo desses anos.

Por fim, registro meus agradecimentos ao meu professor e orientador, Prof. Dr. Dennis Giovani Balreira, pela orientação, confiança e apoio durante o desenvolvimento deste trabalho. Seus ensinamentos e contribuições foram fundamentais para que esta pesquisa pudesse ser realizada.

RESUMO

Devido à grande gama de informações, processos, regimentos e comunicados das disciplinas do Instituto de Informática da UFRGS, surgiu a necessidade de uma plataforma capaz de centralizar informações e facilitar o acesso dos alunos a elas. Essas informações não são repassadas diretamente aos alunos, que precisam buscar no site da universidade ou entrar em contato por e-mail ou telefone. Isso torna o processo mais difícil e demorado, além de demandar tempo e esforço de professores e funcionários. Diante disso, torna-se necessário um meio de esclarecer dúvidas de forma rápida e auxiliando alunos, professores e demais profissionais. O objetivo deste trabalho é desenvolver um Chatbot capaz de responder às perguntas de forma direta aos alunos por meio de um simples bate-papo. Além disso, a aplicação foi desenvolvida utilizando *Small Language Models* para diminuir o consumo de recursos computacionais. Para dar contexto e personalizar o Chatbot, foi utilizada a técnica de *Retrieval-Augmented Generation*, que elevou a acurácia do modelo Qwen 2.5 em 48%, permitindo que assistentes de pequeno porte atingissem até 88% de acerto em questões institucionais, conseguindo responder dúvidas relacionadas aos conteúdos que foram previamente disponibilizados e carregados no banco de dados da aplicação.

Palavras-chave: Ciência da Computação. Chatbot. SLM. RAG. LangChain.

Development of an Academic Chatbot Based on Small Language Models and Retrieval-Augmented Generation for the Informatics Institute of UFRGS

ABSTRACT

Due to the wide range of information, processes, regulations, and announcements from the disciplines of the Institute of Informatics at UFRGS, the need arose for a platform capable of centralizing information and facilitating student access to it. This information is not directly passed on to students, who need to search the university website or contact them by email or phone. This makes the process more difficult and time-consuming, in addition to demanding time and effort from professors and staff. Therefore, a means of quickly clarifying doubts and assisting students, professors, and other professionals is necessary. The objective of this work is to develop a chatbot capable of directly answering students' questions through a simple chat. Furthermore, the application was developed using the Small Language Model to reduce the consumption of computational resources. To provide context and personalize the chatbot, the Retrieval-Augmented Generation technique was used, which increased the accuracy of the Qwen 2.5 model by 48%, allowing small-scale assistants to achieve up to 88% accuracy in institutional questions, successfully answering questions related to content that had been previously made available and loaded into the application's database.

Keywords: Computer Science. Chatbot. SLM. RAG. LangChain.

SUMÁRIO

LISTA DE FIGURAS **8**
LISTA DE TABELAS **9**
LISTA DE ABREVIATURAS E SIGLAS **10**
1 INTRODUÇÃO **11**
2 CONCEITOS E TECNOLOGIAS **13**
2.1 *Language Model* **13**
2.1.1 *Large Language Model* 13
2.1.2 *Small Language Model* 14
2.2 *Embeddings* **14**
2.3 *Retrieval-Augmented Generation* **15**
2.4 Chatbot **16**
2.5 Tecnologias **16**
2.5.1 QDrant 16
2.5.2 LangChain 17
2.5.3 Ollama 17
3 TRABALHOS RELACIONADOS **18**
3.1 Utilização de RAG para aumentar a precisão de Chatbots **18**
3.2 Potencial das SLMs **19**
3.3 Capacidade de SLMs com a técnica de RAG **20**
3.4 Análise Comparativa dos Trabalhos **21**
4 METODOLOGIA **24**
4.1 Coleta e Tratamento de Dados **24**
4.2 Estratégia de Desenvolvimento do Protótipo **25**
4.2.1 Uso de *Small Language Models* 26
4.2.2 Escolha do método RAG 27
4.2.3 A linguagem Python e o Framework LangChain 28
4.3 Protocolo de Avaliação e Modelos Utilizados **29**
4.3.1 Modelos Utilizados 29
4.3.2 Base de Testes e Categorias 30
4.3.3 Procedimento dos Testes 30
5 DESENVOLVIMENTO DO SISTEMA DE CHATBOT **32**
5.1 Arquitetura do Sistema **32**
5.2 Interface do Usuário (*Front-End*) **33**
5.3 Processamento dos Dados (*Back-End*) **34**
5.4 Banco de Dados **37**
6 RESULTADOS E DISCUSSÕES **39**
6.1 Comparação entre os modelos **39**
6.2 O Impacto Quantitativo do RAG na Acurácia **40**
6.3 Análise por Categoria de Conhecimento **41**
6.4 Análise Geral dos Resultados **43**
7 CONCLUSÃO **44**
7.1 Limitações **45**
7.2 Trabalhos futuros **45**
REFERÊNCIAS **47**

APÊNDICE A — PERGUNTAS ELABORADAS..... 49
APÊNDICE B — RESPOSTAS RECEBIDAS..... 68

LISTA DE FIGURAS

Figura 2.1 Exemplo do processo de criação de um *embedding*, envolvendo as fases de carregamento, divisão, *embedding* e armazenamento..... 15
Figura 2.2 Exemplo de fluxograma de execução do RAG, mostrando as etapas de Recuperação, Aumento e Geração. 16

Figura 5.1 Print da tela inicial do Chatbot mostrando a interface para o usuário interagir. 33
Figura 5.2 Exemplo criação de cadeia de *chains* usando o Framework LangChain em Python. 34
Figura 5.3 Fluxograma do tratamento da mensagem do usuário, mostrando o caminho da mensagem até a geração da resposta. 35
Figura 5.4 *Prompt* de configuração da *Chain* de Roteamento usando LangChain. .. 36
Figura 5.5 *Prompt* de configuração da *Chain* de temas não relacionados usando LangChain..... 36
Figura 5.6 *Prompt* de configuração da *chain* de Atendimento usando LangChain... 37

Figura 6.1 Gráfico que mostra a diferença dos resultados entre RAG e Resposta Direta. 40
Figura 6.2 Gráfico que mostra a porcentagem de melhoria nos acertos dos modelos ao utilizar RAG..... 41
Figura 6.3 Gráfico que mostra a porcentagem de acertos em cada categoria usando RAG. 42

LISTA DE TABELAS

Tabela 3.1 Comparação dos trabalhos relacionados e o trabalho desenvolvido..... 23

Tabela 4.1 Comparação dos principais métodos de customização de LLMs: *Prompt Engineering*, *Fine-Tuning* e RAG. 27

LISTA DE ABREVIATURAS E SIGLAS

CIC	Ciência da Computação
COMGRAD	Comissão de graduação
GPT	*Generative Pre-trained Transformer*
IA	Inteligência Artificial
INF	Instituto de Informática
LLM	*Large Language Models*
RAG	*Retrieval-Augmented Generation*
SLM	*Small Language Model*
UFRGS	Universidade Federal do Rio Grande do Sul

1 INTRODUÇÃO

O Instituto de Informática da Universidade Federal do Rio Grande do Sul está em constante transformação, sempre buscando oferecer a melhor qualificação para seus alunos.

> O Instituto de Informática da Universidade Federal do Rio Grande do Sul (INF/UFRGS) é uma instituição dedicada ao ensino superior e à produção de conhecimento. Ele tem como missão formar profissionais de reconhecida qualificação, gerar conhecimento e tecnologia, e apoiar o desenvolvimento de produtos e empresas, compartilhando novos conhecimentos que beneficiem a sociedade. (INF, 2026)

Nesse contexto, ao longo dos anos são sugeridas alterações nos currículos de seus dois cursos: Ciência da Computação e Engenharia de Computação. O objetivo é melhorar a formação de seus alunos, alinhando-se às novidades nos campos de pesquisa e de inovação tecnológica.

Essas alterações incluem novas disciplinas que serão oferecidas, disciplinas que serão mescladas ou que não existirão mais. Além de alterar os requisitos para se formar, como os novos créditos de extensão que conectam o conhecimento universitário à comunidade em geral por meio de projetos, palestras, eventos, etc. Contudo, essas alterações muitas vezes não são transmitidas diretamente aos discentes de maneira clara. Elas são mantidas nos sites oficiais do INF e da UFRGS, dentre diversas outras informações importantes. Portanto, tornam-se de difícil localização ou as informações podem nem ser encontradas.

Nesse contexto, os alunos devem então entrar em contato via e-mail, telefone ou outros meios de contato. Porém, esse processo demanda tempo até se obter uma resposta, e esforço da parte de professores e outros profissionais. Então, os alunos precisam aguardar e caso haja mais alguma dúvida o processo se repete.

Como meio de facilitar essa busca por informação, surgiu a proposta de centralizar os dados e desenvolver um Chatbot acessível por navegador web capaz de responder às dúvidas de alunos sobre o Instituto de Informática da UFRGS. A proposta surgiu a partir de uma demanda da COMGRAD-CIC, que identificou a necessidade de disponibilizar um canal mais acessível e centralizado para consulta de informações acadêmicas. O fato de o sistema ser acessível via navegador permite que os estudantes utilizem a ferramenta em diferentes dispositivos sem necessidade de instalação, além de facilitar a manutenção, atualização e disponibilização do sistema. Com o intuito de reduzir custos de infraestrutura e processamento, foram utilizados *Small Language Models* (SLMs), que demandam menor

custo computacional quando comparados às *Large Language Models* (LLMs).

O Chatbot tem como objetivo responder de forma clara, rápida e direta, sem haver interação de outra pessoa, facilitando que os alunos consigam encontrar o que desejam saber. Assim, não há necessidade de precisar esperar dias para obter uma resposta, podendo consultar quantas vezes quiserem e a qualquer momento. Espera-se obter maior transparência e facilitar a acessibilidade das informações curriculares e institucionais, para reduzir dúvidas e dificuldades enfrentadas pela comunidade universitária durante a jornada acadêmica.

Além de desenvolver uma prova de conceito de um sistema de Chatbot para assuntos acadêmicos, o presente trabalho tem como finalidade comparar o desempenho de diferentes SLMs (devido ao menor custo computacional) combinadas à técnica de *Retrieval-Augmented Generation* (RAG), que mostrou bom desempenho quando utilizado para escopo específico (LIU et al., 2024).

Dessa forma, os objetivos deste trabalho são (i) desenvolver uma prova de conceito de Chatbot acadêmico baseado em *Small Language Models* e na técnica de *Retrieval-Augmented Generation*, bem como (ii) avaliar e comparar diferentes modelos quanto à qualidade das respostas. Busca-se, assim, investigar a viabilidade dessa abordagem para facilitar o acesso a informações institucionais e curriculares no contexto universitário.

O restante do trabalho está dividido da seguinte forma. O Capítulo 2 descreve os principais conceitos e tecnologias que foram utilizados para o desenvolvimento da solução proposta. O Capítulo 3 aborda trabalhos relacionados à metodologia adotada, nos quais são comparados diferentes Modelos de Linguagem Pequenos e analisado o aumento de desempenho quando combinados com a técnica de Geração Aumentada de Recuperação. O Capítulo 4 apresenta a metodologia utilizada neste trabalho, como o levantamento bibliográfico, a coleta e o tratamento dos dados, o desenvolvimento do protótipo e como foi realizada a avaliação. O Capítulo 5 descreve o processo de desenvolvimento da aplicação apresentada, considerando a arquitetura do sistema, o desenvolvimento do *Front-End* e do *Back-End* e a utilização do banco de dados. Os resultados obtidos são discutidos e analisados no Capítulo 6. Por fim, o Capítulo 7 apresenta as conclusões alcançadas a partir do desenvolvimento deste trabalho.

2 CONCEITOS E TECNOLOGIAS

Para compreender melhor o desenvolvimento deste trabalho, é importante conhecer alguns conceitos e tecnologias fundamentais. Assim, este capítulo tem como finalidade apresentar e explicar os principais elementos utilizados na elaboração do trabalho, considerados relevantes pelo autor.

2.1 *Language Model*

Um modelo de linguagem é um sistema computacional treinado para prever a probabilidade de ocorrência de palavras ou "*tokens*" a partir de um determinado contexto, geralmente textual. De forma simplificada, sua função principal é mapear sequências de entrada para uma distribuição de probabilidades sobre um vocabulário, permitindo identificar quais termos são mais propensos a completar uma frase ou preencher lacunas ocultas. Devido à sua capacidade de capturar padrões gramaticais e semânticos, os modelos de linguagem servem como a base para diversas tarefas de Processamento de Linguagem Natural. (CHANG; BERGEN, 2024)

Pode-se separar os Modelos de Linguagem em dois grupos distintos, os *Large Language Models* e os *Small Language Models*. Conforme (GUPTA et al., 2026), ainda não é possível traçar uma linha divisória universalmente aceita entre SLMs e LLMs. Diversos estudos dividem os dois tipos com relação ao número de parâmetros, alguns utilizam um limite de até poucos bilhões de parâmetros, enquanto outros utilizam uma quantidade relativa, conforme será explicado abaixo.

2.1.1 *Large Language Model*

Os *Large Language Model*, ou modelos de linguagem de grande escala em português, são sistemas de inteligência artificial de última geração desenvolvidos para compreender e gerar textos de maneira coerente, simulando a comunicação humana. Eles se baseiam na arquitetura de redes neurais conhecida como *Transformers* e se diferenciam de modelos anteriores por sua escala massiva, sendo definidos tecnicamente por possuírem 10 bilhões de parâmetros ou mais e por serem treinados com volumes gigantescos de dados (NAVEED et al., 2025).

2.1.2 *Small Language Model*

São modelos de linguagem projetados para serem eficientes em termos de recursos, visando a implantação em dispositivos pessoais como desktops, smartphones e dispositivos vestíveis (*wearables*). O objetivo central da pesquisa em SLMs é democratizar o acesso à inteligência artificial, tornando-a acessível e barata para tarefas do dia a dia. Os SLMs possuem uma estrutura reduzida, geralmente definida entre 100 milhões e 5 bilhões de parâmetros (LU et al., 2024). E conforme (CHEN; VAROQUAUX, 2024), os SLMs são definidos de forma relativa como modelos que possuem um número significativamente menor de parâmetros.

Dados os tamanhos dos modelos atuais que podem ultrapassar trilhões de parâmetros (REN et al., 2023), para este trabalho foram considerados *Small Language Models* os modelos com até 7 bilhões de parâmetros.

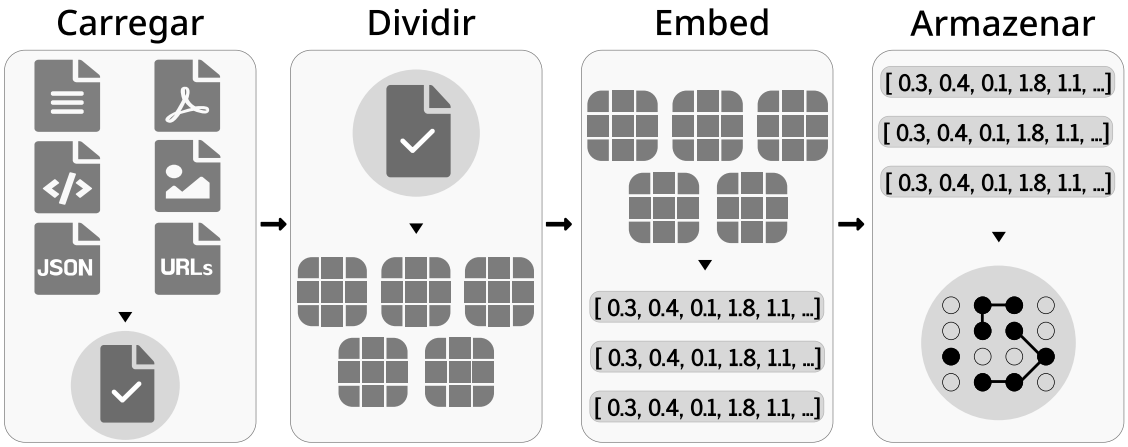
2.2 *Embeddings*

Os *embeddings* são definidos como representações numéricas de dados (como textos, imagens ou áudios) transformadas em matrizes de n-dimensões, conhecidas como vetores, para serem utilizadas como entrada em modelos de aprendizado profundo. Essa técnica funciona convertendo informações complexas e multimodais em listas de números que capturam o contexto e o significado semântico dos dados, permitindo que o sistema realize cálculos matemáticos precisos para comparar a similaridade entre diferentes itens (BOYKIS, 2023).

Um ponto importante é o tamanho do vetor, ele representa a quantidade de dimensões e varia dependendo do modelo de linguagem e da complexidade das relações semânticas. Vetores menores são mais eficientes para manter na memória e para processar, enquanto vetores maiores custam mais processamento, mas podem capturar relações mais complexas. (HESSANI, 2025)

Na Figura 2.1 é mostrado um exemplo de criação de *Embeddings* a partir de diferentes fontes: primeiro os documentos são selecionados e carregados; depois o conteúdo é dividido em pequenos pedaços, e por fim, cada pedaço é transformado em uma representação vetorial de números decimais, que então são armazenados em um banco de dados vetorial.

Figura 2.1 – Exemplo do processo de criação de um *embedding*, envolvendo as fases de carregamento, divisão, *embedding* e armazenamento.



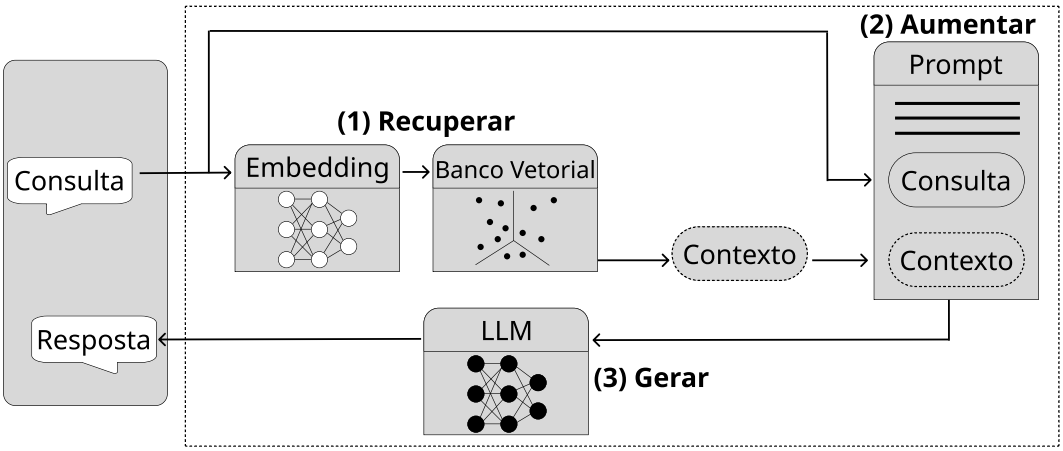
Fonte: Elaborado pelo autor com base em (LANGCHAIN, 2022)

2.3 *Retrieval-Augmented Generation*

Retrieval-Augmented Generation (RAG) é uma arquitetura que combina o conhecimento interno de modelos de linguagem pré-treinados com o acesso a um banco de dados externo de documentos, como a Wikipedia. O processo funciona em duas etapas: primeiro, um componente de recuperação busca documentos relevantes para uma pergunta específica; em seguida, um componente de geração utiliza esses documentos como contexto adicional para produzir uma resposta fundamentada em fatos. Essa técnica é especialmente útil para tarefas que exigem conhecimento intensivo, pois reduz a ocorrência de informações falsas e permite que o conhecimento do sistema seja atualizado de forma simples, apenas trocando os documentos do banco de dados, sem a necessidade de treinar o modelo novamente (LEWIS et al., 2020).

A Figura 2.2 ilustra os três processos executados no RAG: Recuperação, Aumento e Geração. Primeiramente a mensagem do usuário é transformada em *Embedding* e é buscado dados semelhantes no banco de dados através de comparação de vetores. Após isso, a informação obtida é inserida junto à consulta do usuário, gerando o contexto para o Modelo de Linguagem. E, por último, o modelo processa a mensagem do usuário junto do contexto fornecido, tendo mais embasamento para retornar a resposta.

Figura 2.2 – Exemplo de fluxograma de execução do RAG, mostrando as etapas de Recuperação, Aumento e Geração.



Fonte: Elaborado pelo autor com base em (MONIGATTI, 2023)

2.4 Chatbot

Um Chatbot é um programa de computador projetado para simular conversas com usuários humanos, especialmente através da Internet, sendo considerada um dos exemplos mais comuns de Interação Humano-Computador inteligente. O sistema opera como uma entidade que compreende uma ou mais línguas humanas por meio do Processamento de Linguagem Natural, sendo capaz de mimetizar diálogos para diversas finalidades, como educação, recuperação de informação, negócios, e entretenimento (CAHN, 2017).

2.5 Tecnologias

Esta seção tem o propósito de descrever as tecnologias empregadas no desenvolvimento do trabalho, proporcionando uma melhor compreensão sobre a implementação do sistema.

2.5.1 QDrant

O Qdrant é um banco de dados popular de vetores distribuído. Ele é um sistema especializado em gerenciamento de dados projetado para armazenar, indexar e pesquisar representações vetoriais de alta dimensão (OCKERMAN et al., 2025).

Foi escolhido esse banco de dados vetorial por ser uma solução web e de fácil

gerenciamento. Além de ser possível armazenar *Embeddings*, também pode-se visualizar os dados e fazer os cálculos de similaridade na própria plataforma online.

2.5.2 LangChain

O LangChain é um Framework de *software* versátil e modular projetado para facilitar o desenvolvimento, a implantação e o monitoramento de aplicações que utilizam LLMs. Ele funciona como uma estrutura que conecta esses modelos a ferramentas externas e bases de conhecimento, permitindo a criação de sistemas inteligentes, como Chatbots avançados e mecanismos de busca baseados em RAG (MAVROUDIS, 2024).

Esse Framework é o núcleo da aplicação, é ele que gerencia as comunicações com os modelos de linguagem, a conexão com as bases de dados e controla o fluxo das mensagens. Ele abstrai grande parte da implementação, sendo possível trocar um modelo com poucas linhas de código.

2.5.3 Ollama

O Ollama é uma das plataformas de *software* mais populares voltada para executar *Large Language Models* (LLMs) localmente no computador do usuário, sem depender exclusivamente de serviços em nuvem. Ela fornece ferramentas para baixar, gerenciar e executar modelos como Llama, Gemma, DeepSeek, Qwen e Mistral por meio de linha de comando, API local e bibliotecas para programação (Ollama, 2026).

Com ela, é possível baixar, atualizar e gerenciar uma LLM usando um computador ou notebook, sem enviar informações privadas para a nuvem e permite que os modelos sejam executados offline após serem baixados localmente.

3 TRABALHOS RELACIONADOS

O levantamento bibliográfico foi a primeira etapa deste trabalho e teve como objetivo entender como as tecnologias de Processamento de Linguagem Natural vêm sendo utilizadas na área da educação. Essa pesquisa ajudou a identificar as limitações dos modelos de linguagem tradicionais e as soluções mais recentes que podem ser aplicadas no desenvolvimento de assistentes virtuais para instituições de ensino. Para a realização desta pesquisa, utilizou-se principalmente o Google Scholar como ferramenta de busca, por meio de pesquisas baseadas em palavras-chave, como SLM, RAG e Chatbot. Além disso, foram priorizados artigos mais recentes, com preferência para publicações dos últimos três anos.

Os trabalhos escolhidos para esta seção ajudam a mostrar a viabilidade técnica e a importância prática da solução proposta. Embora existam muitos estudos sobre RAG, principalmente em assistentes conversacionais, os artigos selecionados foram priorizados por abordarem temas importantes para este trabalho, como confiabilidade das respostas, viabilidade computacional e aplicação em ambientes acadêmicos.

Os artigos analisados foram organizados em três temas principais. O primeiro foi o uso da técnica de Geração Aumentada por Recuperação (RAG) para aumentar a precisão de Chatbots. Foram analisados estudos que mostram como essa técnica ajuda a reduzir as chamadas alucinações dos Chatbots, fazendo com que as respostas sejam baseadas em documentos e informações oficiais.

O segundo tema foi o potencial dos *Small Language Models* (SLMs). A pesquisa investigou a capacidade de modelos com menos de 7 bilhões de parâmetros para executar tarefas específicas. Os estudos analisados mostraram que esses modelos têm evoluído rapidamente e, em alguns casos, conseguem resultados próximos aos de modelos muito maiores quando utilizados em contextos bem definidos.

Por fim, no terceiro tema investiga-se a capacidade de SLMs com a técnica de RAG substituírem modelos maiores em cenários de ensino, justificando sua adoção no desenvolvimento da solução proposta.

3.1 Utilização de RAG para aumentar a precisão de Chatbots

A implementação de Chatbots baseados em modelos de linguagem de grande escala, tem demonstrado uma capacidade superior em fornecer interações humanas e

empáticas em comparação aos modelos tradicionais de *Machine Learning*. Através da arquitetura de *Retrieval-Augmented Generation* (RAG), esses assistentes conseguem consultar bases de dados externas dinamicamente, o que permite alcançar uma precisão superior a 95% ao fornecer informações baseadas em documentos oficiais, conforme o trabalho de (VAKAYIL et al., 2024). O fluxo de trabalho envolve a extração de texto de arquivos PDF, a fragmentação em *chunks* e a conversão em representações vetoriais armazenadas no ChromaDB, garantindo que a resposta gerada seja contextualmente apropriada e fundamentada em fatos .

O trabalho de (SWACHA; GRACEL, 2025) fornece um panorama atualizado sobre como a técnica de RAG está sendo utilizada para superar o desafio das alucinações [1] em Chatbots educacionais baseados em modelos de linguagem. Os autores realizaram um levantamento de 47 artigos publicados entre 2022 e 2025 para mapear o progresso dos Chatbots RAG na educação. O estudo justifica o uso do RAG como uma solução eficaz para aumentar a confiabilidade dos modelos, integrando técnicas de recuperação de informação para fornecer contexto adicional, o que é crucial em áreas onde a precisão é fundamental, como o ensino.

Embora os componentes para criar Chatbots RAG estejam cada vez mais acessíveis, a integração perfeita entre a recuperação da informação e o modelo de linguagem ainda é o fator crítico para o sucesso da aplicação.

3.2 Potencial das SLMs

A inteligência artificial está evoluindo, e agora modelos menores também conseguem ter bons resultados, desafiando os grandes modelos que usam muito mais poder computacional. O levantamento feito por (GUPTA et al., 2026) analisa cerca de 160 trabalhos e mostra que modelos de linguagem pequenos, com 1 a 9 bilhões de parâmetros, podem alcançar e até superar o desempenho de modelos muito maiores em alguns casos. O estudo mostra que os SLMs ajudam a equilibrar desempenho, eficiência e custo, sendo uma boa opção para ambientes com limitações de *hardware*. Além disso, os autores destacam que esses modelos menores conseguem lidar tanto com tarefas gerais quanto específicas, mostrando que eles ainda têm um papel importante no avanço da tecnologia.

[1] Uma alucinação é uma resposta que não corresponde à realidade. Ou seja, quando questionados, grandes modelos de linguagem às vezes inventam respostas que soam razoáveis (JURAFSKY; MARTIN, 2026).

O trabalho de (LU et al., 2024) é um levantamento abrangente focado unicamente em Modelos de Linguagem Pequenos (SLMs). O estudo escolhe uma abordagem diferente dos demais, que costumam focar em modelos gigantes (LLMs), analisando modelos de 100M a 5B de parâmetros que podem ser executados localmente em dispositivos como smartphones e computadores básicos. Os pesquisadores analisaram 70 modelos de código aberto sob as perspectivas de inovação técnica, desempenho em diversos domínios e custos operacionais, como latência e uso de memória. Foram realizados experimentos com as SLM selecionados em três tarefas: raciocínio de senso comum, resolução de problemas e matemática. Como resultados, foi constatado que entre 2022 e 2024, os SLMs melhoraram seu desempenho mais rápido do que a série Llama-7B, mostrando um potencial de crescimento para tarefas locais. Alguns destaques são para os modelos da família Phi, Qwen e SmolLM que obtiveram resultados semelhantes ao modelo considerado estado da arte Llama 3.1 com 7 bilhões de parâmetros.

Desse modo, os modelos pequenos deixaram de ser apenas uma opção mais barata e começaram a ter um papel importante em sistemas que combinam diferentes tipos de IA. Segundo (CHEN; VAROQUAUX, 2024), eles têm vantagens como serem mais rápidos, eficientes e fáceis de entender, o que faz com que sejam muito usados em áreas como saúde e finanças, onde é importante saber como as decisões são tomadas. O estudo mostra também que modelos pequenos podem cuidar de tarefas mais simples e repetitivas, enquanto os modelos maiores ficam responsáveis por problemas mais complexos. No futuro, os autores acreditam que esses modelos menores serão essenciais para sistemas de IA mais avançados.

3.3 Capacidade de SLMs com a técnica de RAG

O estudo de (LIU et al., 2024) analisa se modelos de linguagem pequenos, combinados com a técnica RAG, conseguem substituir grandes modelos de linguagem no contexto educacional. Para isso, os pesquisadores compararam um SLM de 7 bilhões de parâmetros com modelos maiores, como GPT-3.5 e GPT-4. Os resultados mostraram que o SLM com RAG conseguiu alcançar resultados parecidos com os modelos da OpenAI e, em alguns testes com modelos de código aberto, até superou o GPT-4. Além disso, o modelo menor teve uma geração de respostas mais rápida.

O estudo também destaca que essa abordagem é útil para ajudar estudantes, oferecendo respostas mais precisas e contextualizadas. Para melhorar a usabilidade para os

alunos, os autores recomendam executar os modelos e os bancos de dados de vetores em um servidor e hospedar uma página da web que os alunos possam acessar. Além disso, é proposto que uma implementação local é mais segura para a privacidade dos dados e que o conteúdo educacional permaneça dentro do ambiente institucional. Somado a isso, a possibilidade de atualizar o banco de dados vetorial com informações relevantes do curso ao longo do tempo faz com que essa abordagem seja uma alternativa viável aos LLMs em aplicações educacionais.

Complementando essa ideia, o estudo de (CHEN; VAROQUAUX, 2024) mostra que os modelos pequenos deixaram de ser apenas opções mais baratas e passaram a ter um papel importante em sistemas de IA colaborativos. Os autores destacam que o uso de RAG com modelos menores ajuda a reduzir alucinações, pois o modelo consegue buscar informações em fontes externas atualizadas sem precisar de novos treinamentos frequentes. Em áreas como direito e educação, onde transparência e eficiência são importantes, os SLMs com RAG conseguem oferecer respostas mais confiáveis, baseadas em documentos específicos, com menor custo e mais rapidez. Assim, essa combinação permite criar assistentes especializados que equilibram precisão, eficiência e funcionamento local de forma mais sustentável.

3.4 Análise Comparativa dos Trabalhos

A análise dos trabalhos relacionados mostra que a combinação de *Small Language Models* (SLMs) com a técnica de *Retrieval-Augmented Generation* (RAG) é uma alternativa interessante para o desenvolvimento de Chatbots educacionais eficientes, confiáveis e de baixo custo, pois permite obter respostas mais confiáveis sem a necessidade de utilizar modelos de grande porte. Além disso, os estudos também evidenciam que o RAG tem sido amplamente adotado para reduzir problemas de alucinação e aumentar a precisão das informações fornecidas, característica fundamental em ambientes acadêmicos. Dessa forma, o levantamento bibliográfico forneceu a base teórica necessária para este trabalho e reforçou a hipótese de que modelos menores, quando alimentados com informações relevantes e contextualizadas, podem atender de maneira satisfatória às necessidades da comunidade do INF/UFRGS.

Este trabalho busca aplicar os conceitos discutidos na literatura em uma solução real para o Instituto de Informática da UFRGS, desenvolvida a partir de uma demanda identificada pela COMGRAD-CIC. A proposta diferencia-se por utilizar um SLM integrado

com RAG para disponibilizar informações acadêmicas de forma centralizada e acessível via navegador web, reduzindo custos computacionais, dependência de serviços externos e facilitando o acesso dos alunos às informações institucionais que hoje estão espalhadas em sites e documentos oficiais.

Além disso, como pode-se observar na Tabela 3.1, este trabalho avança e diferencia-se ao comparar diferentes modelos pequenos de linguagem, como Llama 3.2, Qwen 2.5, Phi 3.5, SmolLM2 e Mistral, todos com menos ou até 7 bilhões de parâmetros utilizando a técnica de RAG no âmbito educacional. Portanto, enquanto a literatura fornece as bases sobre o potencial de modelos menores, esta pesquisa analisa de maneira prática o quanto o uso de RAG melhora o desempenho de cada modelo em tarefas específicas, como interpretação de regimentos e planos de ensino. Essa comparação foi realizada com o objetivo de identificar qual modelo apresenta o melhor desempenho para ser utilizado no chatbot acadêmico. Além disso, a avaliação permite verificar quais modelos conseguem trabalhar de forma eficiente com o RAG em contextos institucionais mais complexos, servindo como base para a escolha do modelo mais adequado.

Tabela 3.1 – Comparação dos trabalhos relacionados e o trabalho desenvolvido.

Trabalho	**Usou SLM**	**Implementou RAG**	**Criou Chatbot**	**Domínio**	**Tipo de Trabalho**
(VAKAYIL et al., 2024)	Sim (Llama-2-7B)	Sim	Sim	Saúde Mental	Implementação Prática
(SWACHA; GRACEL, 2025)	Usou SLMs e LLMs	Não	Não	Educação	Estudo (Survey)
(GUPTA et al., 2026)	Sim (modelos de 1B a 9B)	Não	Não	Estado da arte	Estudo (Survey)
(LU et al., 2024)	Sim (modelos de 100M a 5B)	Não	Não	Educação	Estudo (Survey)
(CHEN; VAROQUAUX, 2024)	Sim (BERT-base e LLaMA-8B)	Não	Não	Sistemas de IA colaborativos.	Estudo (Survey)
(LIU et al., 2024)	Sim (neural-chat-7B-v3)	Sim	Sim	Educação	Implementação Prática
Este Trabalho	Sim (Llama-3.2-3B, Qwen-2.5-3B, Phi-3.5-3.8B, SmolLM-2-1.7B, Mistral-7B)	Sim	Sim	Educação	Implementação Prática

Fonte: Autor

4 METODOLOGIA

Este capítulo apresenta a metodologia utilizada no desenvolvimento e na avaliação do Chatbot desenvolvido para o INF. O objetivo da pesquisa foi verificar a viabilidade do uso de Modelos de Linguagem de Pequeno Porte combinados com a técnica de Geração Aumentada por Recuperação em um ambiente acadêmico. Juntamente com a metodologia que foi proposta para executar esse trabalho, é apresentada a justificativa das escolhas que foram tomadas.

Para organizar o trabalho e garantir que os resultados pudessem ser reproduzidos, a metodologia foi dividida em três etapas principais. Na primeira etapa foi realizada a coleta e o tratamento dos dados. Nessa fase, foram extraídas e organizadas informações do site oficial, dos regimentos e dos planos de ensino do Instituto de Informática.

A segunda etapa consistiu no desenvolvimento do protótipo. Nela foram definidas a arquitetura do sistema, as ferramentas utilizadas para a orquestração dos componentes e as tecnologias escolhidas para a implementação da solução. Por fim, na terceira etapa foi elaborado o protocolo de avaliação. Para isso, foi criada uma base de testes com 150 perguntas objetivas, utilizada para comparar o desempenho dos diferentes modelos selecionados ao longo da pesquisa.

4.1 Coleta e Tratamento de Dados

Os dados são um fator crucial para o desenvolvimento de um Chatbot utilizando a técnica de RAG. Sem um conjunto de informações estruturadas e confiáveis, essa abordagem se torna inviável ou completamente ineficaz. A qualidade das respostas geradas pela LLM ou SLM depende diretamente da qualidade do material fornecido. Obter esses dados foi um trabalho manual e extenso, e foi uma etapa que ocupou grande parte do tempo de desenvolvimento. Porém, devido à sua importância, esse foi um processo indispensável.

Para popular o banco de dados da aplicação desenvolvida, os dados foram extraídos de diversas fontes, como do site oficial do INF/UFRGS, sites de outros departamentos da UFRGS com informações relevantes, dos Regimentos do Instituto de Informática (UFRGS, 2026), e dos planos de ensino das disciplinas obrigatórias do curso de Ciência da Computação (UFRGS, 2026). A busca nos sites oficiais foi realizada manualmente, acessando cada página e copiando seus dados em arquivos de texto que seriam lidos pelos *scripts* da aplicação. Como as informações estavam dispostas de maneira a serem lidas em

websites, foi necessário realizar um tratamento manual após copiar os textos. Dentre esses processamentos dos dados, pode-se citar: agrupar e separar as informações em parágrafos, tratar listas de informações muito extensas e separar o texto em pedaços menores (*chunks*).

Cada página principal do site do INF foi separada em um arquivo de texto e cada tópico dentro da página virou um parágrafo. Dessa forma, facilitando o posterior armazenamento desses dados no banco vetorial, e otimizando o funcionamento do mecanismo de recuperação baseado em RAG, como será explicado na Subseção 4.2.2.

Para os dados referentes aos Regimentos e aos Planos de Ensino, o texto extraído diretamente dos arquivos PDF foi armazenado no banco vetorial. Como os arquivos dos Regimentos já possuem uma estrutura bem definida, não foi necessário realizar nenhum tratamento adicional. Por outro lado, os Planos de Ensino apresentam uma formatação que pode dificultar a geração e o armazenamento dos *embeddings*. No entanto, essa limitação será avaliada durante os testes.

Além das fontes utilizadas no projeto, também foi realizada uma tentativa de obtenção de dados reais de dúvidas e atendimentos do canal de comunicação Chama INF, com o objetivo de utilizar essas informações como base complementar para fornecer contexto ao Chatbot. A utilização desse material poderia contribuir significativamente para tornar as respostas do sistema mais próximas das demandas reais dos estudantes e usuários do Instituto.

Entretanto, não foi possível obter acesso a esses dados. A principal limitação encontrada foi relacionada às restrições da Lei Geral de Proteção de Dados Pessoais (LGPD), que estabelece regras rigorosas sobre o tratamento, armazenamento e compartilhamento de informações que possam identificar usuários. Como as mensagens do canal podem conter dados pessoais, informações acadêmicas sensíveis e conteúdos privados dos estudantes, o acesso ao histórico de atendimentos não pôde ser autorizado para utilização na pesquisa.

4.2 Estratégia de Desenvolvimento do Protótipo

A estratégia para o desenvolvimento do Chatbot do INF foi criar uma solução com baixo custo para a instituição, que protegesse os dados acadêmicos e fornecesse respostas confiáveis. Para isso, o projeto foi baseado em três pilares principais: o uso de modelos de linguagem menores (SLMs), a aplicação da técnica *Retrieval-Augmented Generation* (RAG) e a utilização de um *Framework* modular para organizar e integrar os componentes

do sistema.

Outro ponto importante é que o desenvolvimento foi executado em uma máquina local equipada com um processador AMD Ryzen 7 5700x e 32 GB de RAM DDR4. O sistema operava com o Windows 11 e utilizava exclusivamente a placa de vídeo dedicada NVIDIA GeForce GTX 1070.

4.2.1 Uso de *Small Language Models*

A decisão de utilizar *Small Language Models* (SLMs) tem como principal objetivo a redução dos custos de processamento e infraestrutura. Enquanto os modelos de larga escala (LLMs) exigem infraestruturas massivas de servidores, os SLMs são projetados para operar de forma mais eficiente em termos de recursos, permitindo a implantação em dispositivos comuns, como desktops e na infraestrutura da própria universidade. Essa característica reduz a dependência de serviços de nuvem de alto custo, tornando a implementação e a manutenção da inteligência artificial mais viáveis para instituições de ensino.

Além do fator econômico, a privacidade e a segurança dos dados são pontos fortes para o uso de modelos menores. Como apontado por (LIU et al., 2024), a implementação local de uma IA garante que as informações educacionais e os dados dos alunos permaneçam dentro do ambiente institucional, evitando o tráfego de dados sensíveis em servidores externos. Em um contexto acadêmico, onde a integridade da informação é prioritária, ter o controle total sobre o modelo e a base de conhecimento é uma característica relevante.

Outro ponto relevante é que, para tarefas de escopo específico (como tirar dúvidas sobre o Instituto de Informática), não é necessário um modelo com trilhões de parâmetros que detenha todo o conhecimento do mundo. Pesquisas recentes, como as de (LU et al., 2024), demonstram que modelos pequenos (entre 100 milhões e 5 bilhões de parâmetros) têm evoluído muito rapidamente, apresentando desempenho em tarefas locais que se aproxima de modelos bem maiores. Famílias de modelos como Phi, Qwen e Llama já conseguem realizar raciocínios complexos de forma ágil, superando em eficiência modelos considerados estado da arte de gerações anteriores.

Por fim, a combinação de SLMs com a técnica de RAG (Retrieval-Augmented Generation) potencializa esses resultados. Estudos indicam que modelos pequenos, quando alimentados com o contexto correto de uma base de dados externa, podem igualar ou até superar a acurácia de LLMs como o GPT-4 em domínios específicos (LIU et al., 2024). Dessa forma, o uso de SLMs neste projeto é uma maneira de garantir uma resposta rápida

para o usuário, reduzindo a latência e o esforço computacional sem sacrificar a qualidade da informação entregue.

4.2.2 Escolha do método RAG

Existem diversas abordagens para customizar a resposta gerada por um Modelo de Linguagem, como a Engenharia de *Prompts* (Por exemplo: *Chain-of-Thought*, *Few-Shot*) em que se orienta como a LLM deve responder; o *Fine Tuning* em que se treina a LLM novamente com novos dados de um domínio específico, alterando seus parâmetros; e, por fim, a abordagem que foi utilizada neste trabalho, a técnica de *Retrieval-Augmented Generation* (RAG) em que é adicionada uma nova etapa antes de uma LLM (ou SLM) gerar a sua resposta.

Tabela 4.1 – Comparação dos principais métodos de customização de LLMs: *Prompt Engineering*, *Fine-Tuning* e RAG.

Abordagem	**Melhor para**	**Custo**	**Dependência de Dados**	**Atualização**
Prompt Engineering	Prototipar, orientar comportamento	Baixo	Baixa	Imediata
Fine Tuning	Especialização e consistência	Médio / Alto	Alta	Exige *retraining*
RAG	Informação atualizada e confiável	Médio	Alta	Simples (atualiza documentos)

Fonte: Autor

Cada abordagem possui seus pontos positivos e negativos, e depende do contexto em que será usada, como mostra a Tabela 4.1. A Engenharia de *Prompts* é simples e não possui custo de treinamento, mas adiciona mais texto aos *prompts*, podendo impactar o processamento e a latência, e depende das limitações de conhecimento do modelo. O *Fine Tuning* é bom para focar em um domínio específico e gera melhor performance e coerência nas respostas, mas demanda muito mais processamento e o custo computacional é elevado, além de precisar de muitos dados de boa qualidade. E o RAG permite customizar o modelo com menor custo comparado ao *Fine Tuning*, ele reduz as alucinações, e, para se ter mais informações, basta adicionar novos dados ou atualizar os documentos, e não o modelo. Contudo, esta abordagem pode aumentar a latência por ter passos extras, requer mais infraestrutura, como bancos vetoriais, e as respostas dependem da qualidade dos dados

fornecidos e geração dos *embeddings*.

Aprofundando na técnica de RAG, o processo possui três etapas principais: Recuperação (*Retrieval*), Aumento (*Augmentation*) e Geração (*Generation*). Considera-se que já se possui um banco com os dados nos quais a LLM deve se basear para gerar a resposta. Na Recuperação, é realizada a busca de dados para responder à mensagem do usuário, geralmente é feita uma busca por similaridade de *Embeddings* (vetores) para encontrar o trecho com as informações mais relevantes. Após isso, é a etapa do Aumento que é feita a concatenação das informações recuperadas com o *prompt* original do usuário, adicionando o contexto dos dados para a LLM analisar. E, por último, é feita a etapa de Geração em que a LLM irá gerar a resposta baseada no seu próprio conhecimento e mais o contexto fornecido, para assim, gerar uma resposta mais coerente e precisa. Esse processo é ilustrado na Figura 2.2.

Dessa forma, foi escolhido o método RAG devido à sua alta escalabilidade, seu menor custo de processamento, e sua facilidade em incrementar o conhecimento do modelo conforme se adiciona mais dados, visto que a universidade está em constante evolução. Espera-se que a utilização dessa técnica permita gerar respostas mais recentes, coerentes e contextualizadas aos documentos institucionais da universidade, reduzindo alucinações da SLM e facilitando a atualização da base de conhecimento. Assim, sendo necessária a construção de uma base de dados confiável para alimentar o sistema.

4.2.3 A linguagem Python e o Framework LangChain

Para o desenvolvimento do sistema, a escolha da linguagem Python 3 foi fundamental, visto que ela é hoje a principal ferramenta utilizada no campo da Inteligência Artificial e do Processamento de Linguagem Natural. O Python oferece um ecossistema robusto de bibliotecas que facilitam a integração entre diferentes componentes do *software*, desde o tratamento inicial dos dados acadêmicos até a comunicação final com os modelos de linguagem. (Python Software Foundation, 2026)

Integrado ao Python, o Framework LangChain atua como o núcleo da aplicação, funcionando como uma estrutura modular que conecta os modelos de linguagem a bases de conhecimento externas. O grande diferencial dessa tecnologia é a sua capacidade de abstração; através da *LangChain Expression Language* (LCEL), conseguimos criar cadeias de execução (*chains*) que organizam todo o fluxo de processamento. Isso permite que a saída de uma função (como a recuperação de um dado no banco vetorial) seja passada

diretamente para a próxima etapa, garantindo que o modelo receba o contexto correto antes de gerar a resposta.

Essa modularidade oferecida pelo LangChain também traz uma vantagem estratégica para o projeto: a facilidade de manutenção. Com ele, é possível trocar o modelo de linguagem utilizado (como alternar entre o Llama e o Phi, por exemplo) com pouquíssimas alterações no código. Essa flexibilidade foi essencial para realizar as comparações de desempenho apresentadas neste trabalho e garante que o Chatbot possa evoluir facilmente conforme novos modelos e tecnologias surgirem.

4.3 Protocolo de Avaliação e Modelos Utilizados

Esta seção apresenta os critérios utilizados para escolher os modelos de linguagem e a forma como os testes foram realizados. O principal objetivo da avaliação foi comparar o desempenho de diferentes modelos de pequeno porte em um cenário próximo do uso real do Chatbot do INF, além de analisar como a técnica de RAG influencia na qualidade e precisão das respostas.

4.3.1 Modelos Utilizados

Para avaliar o desempenho dos *Small Language Models* (SLMs) no contexto do projeto, foram escolhidos cinco modelos que se destacam pela capacidade de execução local e pela boa relação entre desempenho e consumo de recursos. A seleção também considerou diferentes tamanhos de parâmetros, todos dentro do limite de até 7 bilhões, permitindo observar como o tamanho do modelo influencia nos resultados.

Os modelos utilizados foram:

- Llama 3.2 (3B) (Meta AI, 2024): escolhido pelo bom equilíbrio entre velocidade, eficiência e capacidade de raciocínio (LU et al., 2024).
- Qwen 2.5 (3B) (Qwen Team, 2024) e *Phi* 3.5 (3.8B) (Microsoft, 2024): modelos recentes que vêm apresentando bons resultados em tarefas locais e têm evoluído rapidamente em comparação com modelos maiores (LU et al., 2024).
- SmolLM2 (1.7B) (Hugging Face TB, 2025): utilizado para verificar se modelos muito pequenos conseguem compreender perguntas e relacioná-las corretamente com os textos fornecidos pelo sistema RAG (LU et al., 2024).

- Mistral (7B) (Mistral AI, 2023): escolhido como referência de maior porte dentro da categoria de execução local, já que estudos recentes mostram que ele pode alcançar resultados próximos aos modelos *GPT*-3.5 e *GPT*-4 quando combinado com RAG (LIU et al., 2024).

4.3.2 Base de Testes e Categorias

A avaliação foi realizada utilizando um conjunto de 150 perguntas objetivas criadas especificamente sobre o Instituto de Informática da UFRGS. Cada questão possui cinco alternativas, de A até E, com apenas uma resposta correta. As respostas corretas foram distribuídas de forma equilibrada entre as alternativas, com cada letra aparecendo em 20% das questões. Por ser uma etapa extensa e trabalhosa, as perguntas foram geradas com o auxílio da IA do Google, o NotebookLM, devido sua capacidade de gerar material de avaliação com base exclusivamente nos documetos anexados. O conjunto completo das perguntas está disponível no Apêndice A deste trabalho.

As perguntas foram divididas em três grupos com 50 questões cada, buscando representar diferentes tipos de informação e níveis de dificuldade:

- Site do INF: perguntas relacionadas a informações institucionais, história e dados disponíveis no site oficial.
- Regimentos: questões sobre regras, normas acadêmicas, sanções disciplinares e procedimentos administrativos extraídos de documentos oficiais em PDF que são bem estruturados.
- Planos de Ensino: perguntas baseadas nos planos de ensino das disciplinas obrigatórias da Ciência da Computação, incluindo cronogramas e fórmulas de cálculo de notas, extraídos de documentos PDF com baixa estruturação para indexação no banco vetorial.

4.3.3 Procedimento dos Testes

Para medir o impacto da técnica de RAG, cada modelo foi avaliado em duas abordagens diferentes. Em cada abordagem, foram perguntadas as mesmas 150 perguntas objetivas citadas na Subseção 4.3.2. Além disso, durante os testes, a etapa de verificação de escopo não foi considerada. Para fins de avaliação, assumiu-se que todas as

perguntas estavam dentro do escopo do Chatbot, permitindo que a análise fosse focada exclusivamente na capacidade dos modelos de responder corretamente utilizando o contexto fornecido pelo RAG.

Na primeira abordagem, chamada de Resposta Direta, o modelo precisava responder às perguntas utilizando apenas o conhecimento aprendido durante o treinamento original, sem acesso aos documentos do Instituto de Informática e sem nenhum contexto. Esse teste permitiu identificar limitações de conhecimento específico e possíveis casos de "alucinação".

Na segunda abordagem, chamada de Com RAG, o modelo recebia trechos relevantes recuperados do banco vetorial Qdrant antes de gerar a resposta. Dessa forma, foi possível avaliar se o uso de informações oficiais melhorava a precisão e a confiabilidade das respostas fornecidas pelo Chatbot.

5 DESENVOLVIMENTO DO SISTEMA DE CHATBOT

Este capítulo tem como objetivo apresentar o processo de desenvolvimento do sistema proposto. Ele está dividido em quatro seções, sendo elas: Arquitetura do Sistema, Interface do Usuário (*Front-End*), Processamento dos Dados (*Back-End*) e Banco de Dados. Além disso, todo o código utilizado neste trabalho está disponível publicamente em um repositório no GitHub [1].

5.1 Arquitetura do Sistema

O sistema foi projetado em uma arquitetura Cliente-Servidor, em que os serviços são consumidos pelo *Front-End* por meio de chamadas de API. A aplicação foi desenvolvida usando a linguagem de programação Python no *Back-End* para fazer o processamento das mensagens, o Framework Angular no *Front-End* para o usuário interagir, o banco vetorial Qdrant para guardar os dados obtidos e o banco de dados SQLite para guardar o histórico do chat com o usuário. Abaixo será detalhado cada parte da aplicação.

Em resumo, o sistema desenvolvido funciona como um assistente virtual acadêmico acessado por meio de um site semelhante aos Chatbots modernos, onde o usuário pode conversar de forma simples e intuitiva. Quando uma mensagem é enviada, o sistema analisa automaticamente o conteúdo para identificar se a pergunta está relacionada ao contexto universitário. Caso esteja dentro desse contexto, o assistente busca informações relevantes em sua base de conhecimento para gerar uma resposta adequada; caso contrário, ele informa que o assunto não faz parte do seu escopo de atendimento. Todo esse processo acontece de forma automática e rápida, permitindo que o usuário tenha uma experiência de conversa natural e organizada.

Para conseguir responder às perguntas, a aplicação utiliza diferentes tecnologias trabalhando em conjunto. O site apresentado ao usuário é responsável apenas pela interface de conversa, enquanto uma parte interna do sistema realiza o processamento das mensagens, consulta os dados armazenados e gera as respostas utilizando inteligência artificial. Além disso, o sistema mantém o histórico das conversas para preservar o contexto do diálogo e utiliza uma base de dados especializada para localizar informações acadêmicas relevantes. Dessa forma, o Chatbot consegue fornecer respostas mais coerentes, contextualizadas e alinhadas às informações oficiais da universidade.

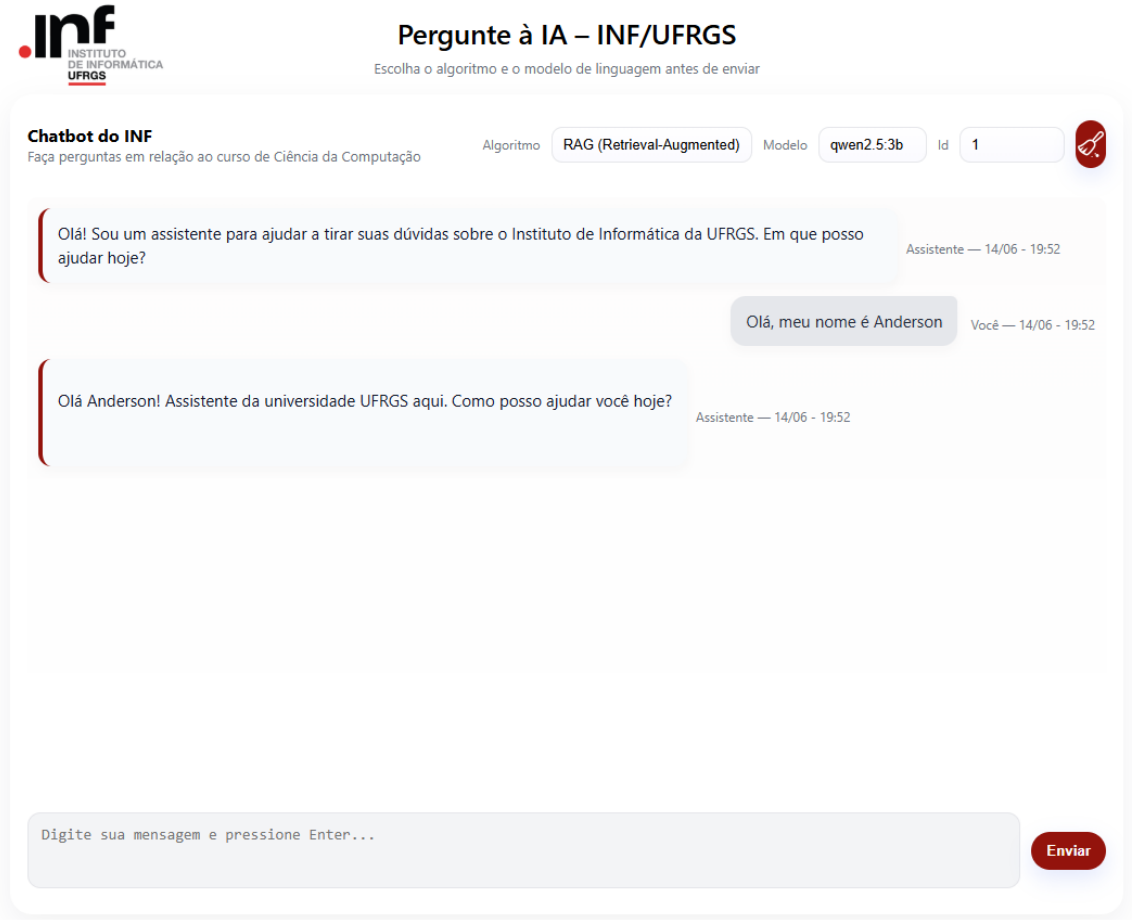
[1] https://github.com/AnderCaporale/tcc-chatbot-inf

5.2 Interface do Usuário (*Front-End*)

Para a interação do usuário e visualização das mensagens, foi desenvolvido um website usando o Framework JavaScript Angular, pois ele proporciona um ambiente completo de desenvolvimento e é capaz de criar aplicações web dinâmicas, interativas e robustas (GEETHA et al., 2022). Seu fácil controle de estados e ciclo de vida de componentes o torna uma ótima opção para o trabalho proposto.

O layout do sistema foi inspirado nos modelos de Chatbot já existentes, como ChatGPT, Google Gemini e Microsoft Copilot. Porém foi adicionado o logo do INF e campos de destaque na cor vermelho-escuro para remeter às cores do Instituto de Informática da UFRGS, como é mostrado na Figura 5.1.

Figura 5.1 – Print da tela inicial do Chatbot mostrando a interface para o usuário interagir.



Fonte: Autor

Ele foi projetado para ser simples e intuitivo, possuindo um campo para digitar o texto, um botão para enviar a mensagem e outro para limpar o chat. Também possui alguns campos auxiliares para selecionar o tipo de estratégia usada ou a LLM que irá gerar as respostas (essas funcionalidades serviram para facilitar a comparação entre modelos)

e o Id para representar diferentes usuários. Além disso, foi adicionado um botão para limpar o histórico do Chat e criar uma nova conversa. Na parte central da página é onde aparecem as mensagens do Assistente e do Usuário, cada uma em um lado da tela, e com a sinalização do autor e do horário.

5.3 Processamento dos Dados (*Back-End*)

O *Back-End* faz a conexão com os bancos de dados e controla o histórico de chat do usuário. Ele é responsável por tratar as mensagens recebidas do *Front-End* e por fazer as chamadas das APIs das SLMs. Além da linguagem Python, foram utilizadas a biblioteca LangChain, para interação com as SLMs e com os bancos de dados; a biblioteca FastAPI, para geração do servidor responsável por tratar as requisições do *Front-End*; e a plataforma Ollama para a execução local dos Modelos de Linguagem.

Na biblioteca LangChain, pode-se criar *pipelines* usando a *LangChain Expression Language* (LCEL), que encadeia funções, de forma que a saída da primeira função sirva de entrada para a próxima e assim sucessivamente. Para isso, deve-se usar o operador especial pipe "|". Como pode ser observado no exemplo da Figura 5.2, no processamento da variável *chain*, a saída do *prompt* é passada para o Modelo de Linguagem, e sua saída é passada para o *Output Parser* que fará o tratamento final antes de retornar a resposta para o usuário. Essas cadeias podem ser tão grandes quanto necessário, e além de ser possível realizar *chains* sequenciais, também é possível criar chains paralelas que executarão ao mesmo tempo ou *chains* condicionais que irá executar uma ou outra dependendo de alguma condição.

Figura 5.2 – Exemplo criação de cadeia de *chains* usando o Framework LangChain em Python.

```
1    chain = prompt | modelo | output_parser
2
```

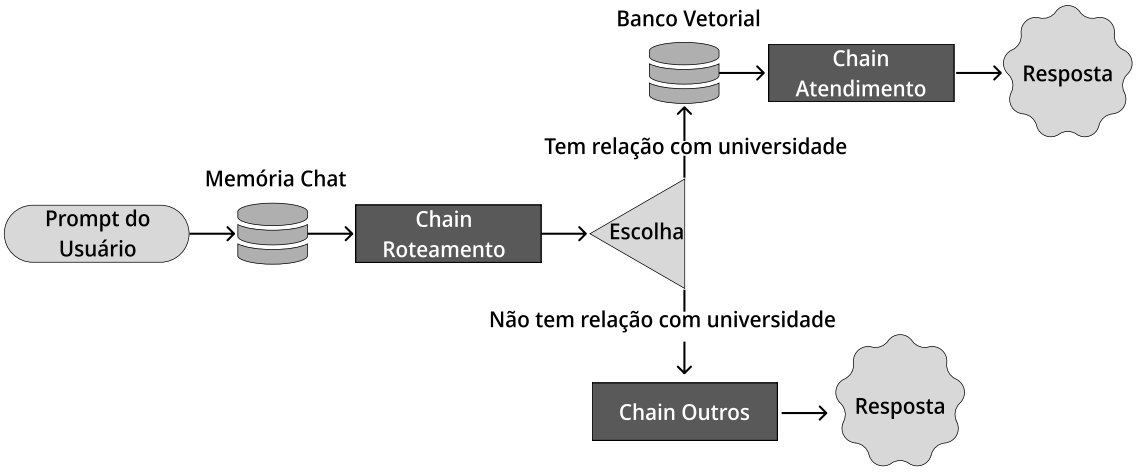
Fonte: Autor

O Chatbot está configurado para responder apenas perguntas relacionadas ao contexto acadêmico, qualquer mensagem que não se encaixe nele será informado que o assunto não está no escopo Chatbot. Entretanto, ele possui um tom amigável e responderá mensagens de saudação. Para classificar se uma mensagem está ou não dentro do escopo do Chatbot, foi utilizada a LLM openai/gpt-oss-20b (OpenAI, 2025). Essa escolha foi feita porque a etapa de classificação não faz parte do objetivo principal deste trabalho. O foco da pesquisa é avaliar e comparar o desempenho de modelos de linguagem de pequeno

porte na geração de respostas corretas quando recebem o contexto necessário por meio do RAG.

A aplicação desenvolvida possui o arquivo *main.py* que gerencia a conexão de todos os componentes (chains). Como ilustrado nos retângulos em cinza escuro da Figura 5.3, foram criadas três *Chains* que interagem com os Modelos de Linguagem. A primeira é a de Roteamento, que irá definir se o *prompt* do usuário tem relação com o contexto acadêmico ou se é sobre outro assunto. Caso seja sobre a universidade, será chamada a *Chain* de Atendimento, que irá responder ao *prompt* do usuário baseado nos textos recuperados do banco vetorial. E caso seja sobre um assunto diverso, irá para a *Chain* de Outros que responderá que a mensagem está fora do escopo. Essa abordagem faz com que seja economizado custo de processamento, pois não é necessário fazer a etapa de busca vetorial e nem aumentar o *prompt* com o contexto antes de passar para a SLM.

Figura 5.3 – Fluxograma do tratamento da mensagem do usuário, mostrando o caminho da mensagem até a geração da resposta.



Fonte: Autor

Para a *Chain* de roteamento foi utilizado um *prompt* que define a LLM como um especialista de classificação, como mostra a Figura 5.4. Essa *Chain* vai fazer a mensagem do usuário ser classificada em 1 caso seja um contexto acadêmico ou em 0 caso contrário. Ela também recebe o histórico da conversa para ter um contexto maior, mas não retorna nenhuma resposta sobre a pergunta, apenas faz a classificação. Trata-se de uma etapa importante para o funcionamento do sistema, pois é a partir desse resultado que é determinado se a pergunta do usuário será processada e respondida ou não, garantindo maior controle sobre o escopo e economizando processamento adicional de consultar e fazer operações com o banco vetorial.

A *Chain* de outros assuntos recebe um *prompt* dizendo para a SLM que ela é um

Figura 5.4 – *Prompt* de configuração da *Chain* de Roteamento usando LangChain.

```
# Criando o ChatPromptTemplate que solicitará ao LLM que ele classifique a entrada do usuário:
sys_prompt_rota = """
    Você é um especialista em classificação. Você receberá perguntas do usuário e precisará classificar. Se a mensagem do usuário for sobre as mensagens do chat, relacionado à dúvidas sobre universidade, professores ou entidades relacionadas à faculdade a opcao deve ser 1, caso contrário deve ser 0.
    Atente-se que a mensagem pode se referir indiretamente a algum dos casos acima, então também deve ser a opção 1.
    Se a mensagem for algo em relação ao histórico do chat, também responda a opção 1.
    Apenas classifique, não responda à pergunta do usuário.
    Importante: Responda apenas com a resposta final, sem descrever seu raciocínio.
    Pergunta Usuário: {input_user}
    """

```

Fonte: Autor

assistente de universidade e só deve responder perguntas sobre esse tema, conforme a a Figura 5.5. Adicionalmente, ela recebe apenas o histórico de mensagens para ter algum contexto das últimas mensagens para poder responder de maneira mais natural e poder desenvolver o bate-papo.

Figura 5.5 – *Prompt* de configuração da *Chain* de temas não relacionados usando LangChain.

```
#Criando o ChatPromptTemplate que vai agir como assistente, mas não irá responder.
sys_prompt_fora_do_tema = """
    Não responda a pergunta. Você está recebendo uma mensagem fora do seu escopo, então informe ao usuário que não é seu objetivo.
    ## Regras:
    1 - Não responda a pergunta.
    2 - Você é um assistente de uma universidade e não deve responder as perguntas.
    3 - Nunca utilize conhecimento próprio para responder.
    4 - Nunca explique o tema perguntado.
    5 - Caso o usuário apenas cumprimente (Olá, Oi, Boa tarde, etc.), responda normalmente e apresente seu objetivo (assistente da universidade UFRGS).
    6 - A pergunta não deve ser respondida.

    Exemplos:

    Usuário:
    Faz qualquer pergunta.
    Resposta:
    Desculpe, posso auxiliar apenas com assuntos relacionados à UFRGS e ao Instituto de Informática (INF).
    """

```

Fonte: Autor

E, por último, a *Chain* de Atendimento que é a parte central da aplicação, ela tem o *prompt* mostrado na Figura 5.6. Essa é a *Chain* mais importante e é nela que as perguntas dos usuários dentro do contexto universitário são processadas e efetivamente respondidas. Nessa etapa, são executadas duas funções importantes em paralelo antes de

encaminhar para o modelo: pegar o histórico do chat para manter a coerência do diálogo; e o contexto retornado pelo banco vetorial. Esses dois dados serão passados para a SLM para fornecer informações relevantes já armazenadas por meio de *Embeddings*. Nessa *chain*, a SLM recebe um *prompt* dizendo que é um assistente de universidade e deve responder as perguntas dos alunos. A SLM recebe regras de não inventar informações e de encaminhar o aluno ao site oficial caso não tenha a resposta necessária. Também deve basear-se no contexto recuperado dos *Embeddings*, mas evitando explicitar que essas informações foram recuperadas, garantindo uma conversa mais fluída e natural.

Figura 5.6 – *Prompt* de configuração da *chain* de Atendimento usando LangChain.

```
# Criando o ChatPromptTemplate que irá responder a pergunta do usuário se estiver no escopo
sys_atendimento_geral_prompt = """
    Você é um assistente de uma universidade e tem como objetivo responder às perguntas dos alunos. Os assuntos serão sobre dúvidas em relação à faculdade UFRGS e ao Instituto de Informática, também chamado de INF.

    ## Regras:
    1 - Nunca invente informação. Responda que desconhece o assunto caso você não souber responder e diga para consultar o site oficial da faculdade.
    2 - Sempre se baseie no contexto que é entregue entre as tags <contexto></contexto>. As informações presentes nestas tags foram obtidas de uma base de conhecimento.
    3 - Evite falar 'no contexto...' ou 'conforme o contexto...' porque o usuário desconhece sobre a presença desse contexto.

    ## Contexto Recuperado:
    <contexto>
    {contexto_obtido}
    </contexto>
    """

```

Fonte: Autor

Assim, garante-se que todo o fluxo de trabalho seja executado de maneira correta e eficiente, evitando que seja feito processamento computacional desnecessário. Além disso, permite manter a aplicação apenas dentro do seu contexto definido para o projeto.

5.4 Banco de Dados

Garantir que as informações sejam armazenadas corretamente é uma etapa fundamental da aplicação. Como explicado na Seção 4.1, os dados são de suma importância para o funcionamento da aplicação. Foram utilizados dois bancos de dados que possuem funções distintas, mas ambos relacionados em fornecer contexto para a LLM.

O banco mais importante da aplicação é o Qdrant que foi utilizado para armazenar e buscar os *Embeddings*. É nesse banco que estão guardadas todas as informações

obtidas sobre o Instituto de Informática. Antes de salvar os textos no banco é necessário transformá-los em *Embeddings*. Para isso, foi utilizado o modelo gemini-embedding-001 (Google, 2026) e a quebra em parágrafos com até 1000 caracteres. Para a sobreposição, foi utilizado um tamanho de 100 caracteres, o que corresponde a 10% do tamanho de cada chunk nesta implementação. Esse valor segue uma prática comum, de acordo com as recomendações da documentação do LangChain e da comunidade (RABHI, 2026).

E para o banco de histórico de chat foi utilizado o SQLite salvo localmente, por se tratar de um banco simples com apenas três colunas e ser executado apenas consultas básicas. Nele consta a coluna **`id`** que salva o id da mensagem, a coluna **`session_id`** que identifica o usuário a quem a mensagem pertence, e a coluna **`message`** que salva o conteúdo da mensagem e se ela foi escrita pelo usuário ou pela IA.

6 RESULTADOS E DISCUSSÕES

Este capítulo apresenta os resultados dos testes realizados com o protótipo do Chatbot do INF. O objetivo é analisar o desempenho dos modelos de linguagem utilizados e verificar como a técnica de Geração Aumentada de Recuperação influencia na precisão das respostas fornecidas pelo sistema. Nas seções seguintes, será avaliado o comportamento do modelo ao responder perguntas utilizando apenas o conhecimento adquirido durante seu treinamento, em comparação com seu desempenho quando recebe um contexto adicional para encontrar as respostas corretas.

Os resultados foram obtidos a partir de 150 perguntas objetivas definidas no protocolo de avaliação da Seção 4.3. Com isso, foi possível comparar as respostas geradas diretamente pelos modelos com aquelas produzidas utilizando o RAG, analisando o desempenho do Chatbot em diferentes áreas de conhecimento institucional. A lista completa das respostas recebidas por cada SLM pode ser encontrada no Apêndice B.

6.1 Comparação entre os modelos

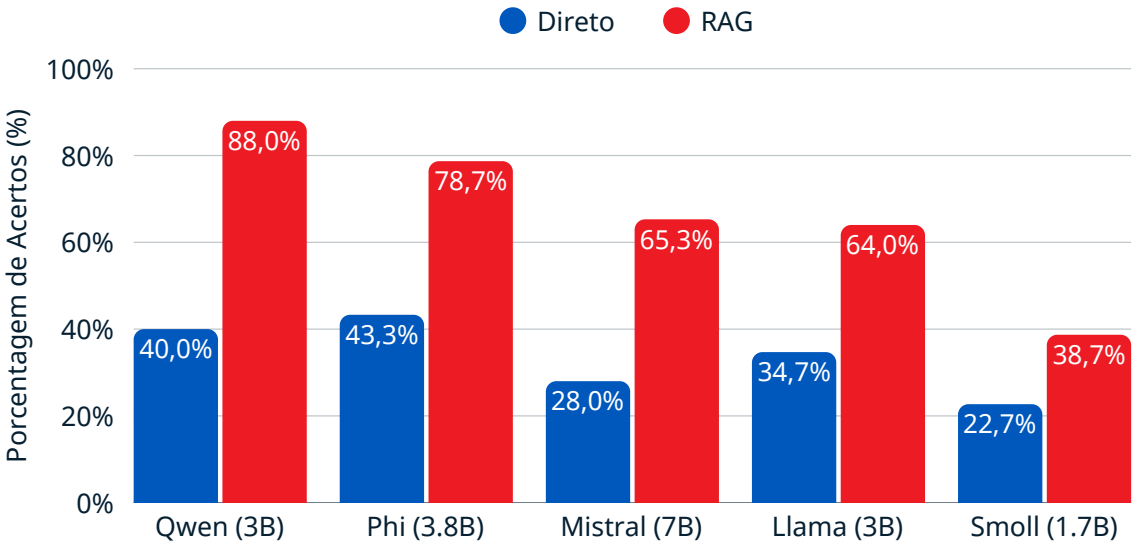
Entre os modelos utilizados, o Qwen 2.5 e o Phi 3.5 foram os grandes destaques positivos. Eles demonstraram uma alta precisão ao analisar as informações contidas no contexto recuperado, acertando detalhes que modelos maiores às vezes deixavam passar. O Llama 3.2 também se mostrou muito equilibrado, sendo excelente para analisar textos recuperados, embora ainda se atrapalhe um pouco com os números dos planos de ensino. Já o Mistral 7B, apesar de ser o maior modelo do grupo, não se mostrou infalível e ocasionalmente falhou em detalhes específicos que os modelos menores e mais recentes conseguiram capturar com mais precisão, além de ter uma forte tendência a responder a alternativa C.

Os resultados do SmolLM2 mostraram que esse modelo pode não ter a capacidade necessária para aproveitar o RAG de forma eficiente. Mesmo com as informações disponíveis, ele continuou cometendo erros em fatos simples que todos os outros modelos acertaram, como a localização do campus ou o nome do atual diretor. Os resultados observados sugerem que existe um tamanho mínimo de parâmetros para que a IA consiga relacionar a pergunta ao texto de apoio de forma útil.

Esses resultados mostram que nem todos os modelos de linguagem são igualmente adequados para aplicações baseadas em RAG. Embora todos tenham acesso ao mesmo

contexto, alguns conseguem interpretar e utilizar essas informações de forma muito mais eficiente, gerando respostas mais precisas e confiáveis. Isso reforça a importância de avaliar o desempenho dos modelos antes de definir qual será utilizado em um chatbot acadêmico.

Figura 6.1 – Gráfico que mostra a diferença dos resultados entre RAG e Resposta Direta.



Fonte: Autor

6.2 O Impacto Quantitativo do RAG na Acurácia

Sem o RAG, os modelos tentam "adivinhar" as respostas com base no que aprenderam no treino, o que gera muitos erros em dados específicos da UFRGS. Quando o RAG foi utilizado, a taxa de acertos deu um grande salto, porque o modelo passou a consultar o documento antes de gerar a resposta final. É nítido que o RAG transforma um modelo que apenas "chuta" com o seu conhecimento interno em um assistente que realmente informa fatos reais baseados nos documentos fornecidos.

No método de Resposta-Direta, modelos como o Llama e o Mistral erraram quase todas as perguntas sobre datas históricas e nomes de gestores. Já com o uso da técnica de busca, esses mesmos modelos acertaram o ano de criação do curso de Computação e até a área física ocupada pela incubadora CEI. Isso demonstra que o RAG é uma peça fundamental para a confiabilidade de um sistema de informação institucional.

Comparando as duas abordagens:

- Resposta Direta: Modelos como o Smoll 1.7B e o Mistral 7B apresentam um desempenho significativamente baixo ao tentar responder apenas com o conheci-

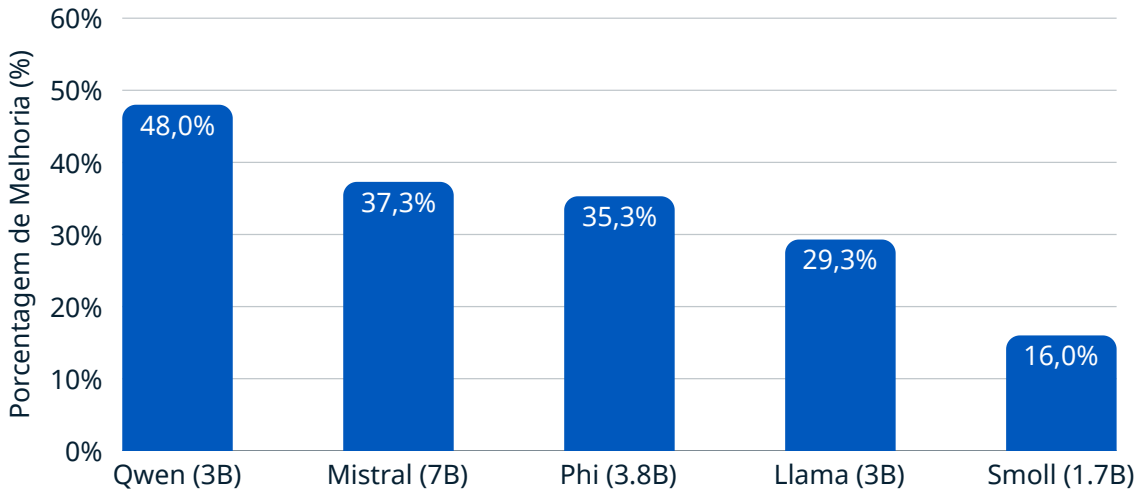
mento de treinamento. Eles frequentemente “alucinam” datas e nomes que parecem aceitáveis, mas estão incorretos no contexto da UFRGS.

- Com RAG: Há um salto de performance onde perguntas antes erradas passam a ser respondidas corretamente de forma consistente, como o ano de fundação do INF ou a área ocupada pelo CEI.

Como se pode observar no gráfico da Figura 6.2, todos os modelos apresentaram melhorias com a utilização do RAG. Em especial, o modelo Qwen ganhou uma performance de 48% na taxa de perguntas respondidas corretamente. Os outros modelos também obtiveram um aumento em suas respostas corretas.

Apesar disso, o modelo que menos teve aumento foi o Smoll. Como mencionado na seção anterior, mesmo com as informações corretas disponíveis, ele não conseguiu processar todas as informações e chegar na resposta correta.

Figura 6.2 – Gráfico que mostra a porcentagem de melhoria nos acertos dos modelos ao utilizar RAG.



Fonte: Autor

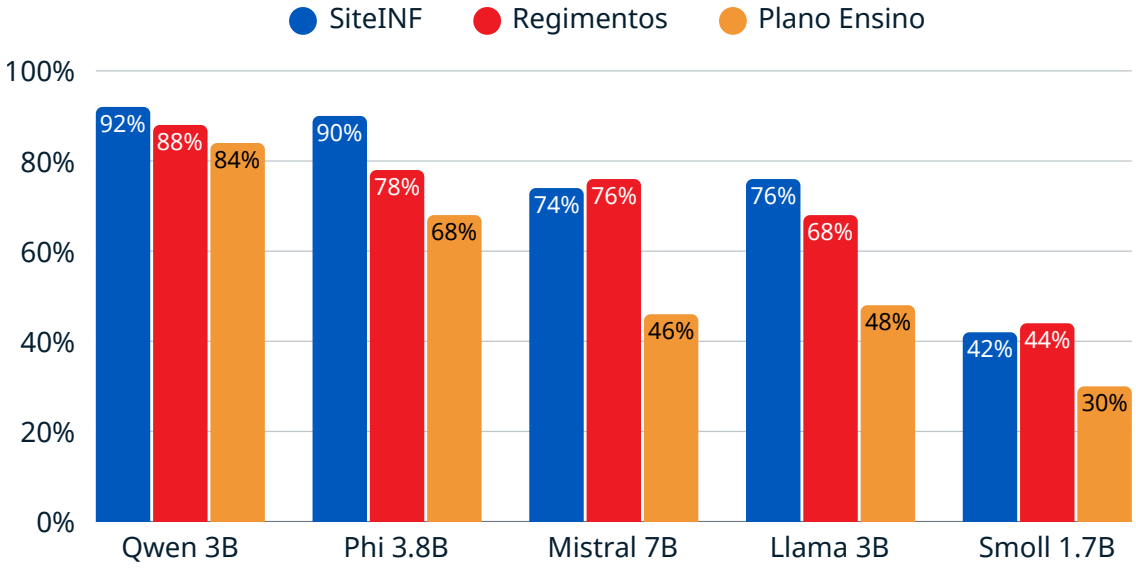
6.3 Análise por Categoria de Conhecimento

Na Figura 6.3 pode-se perceber uma diferença da quantidade de respostas corretas conforme a categoria analisada. Em todos os modelos, as perguntas baseadas no site do INF e nos Regimentos foram as que mais se beneficiaram do RAG, pois tratam de regras e fatos muito específicos que geralmente não fazem parte do treinamento comum de uma IA. Em Regimentos, por exemplo, é improvável o modelo saber o tempo de suspensão de um aluno sem ler a norma, e o RAG permitiu que quase todos os modelos acertassem essa

questão técnica. Ao consultar as informações do PDF das normas disciplinares, o modelo parou de inventar punições ou regras que não existem.

Por outro lado, a categoria de Plano de Ensino foi a mais desafiadora, mesmo com a técnica de busca ativa. Muitos modelos se atrapalharam com os pesos de notas, as fórmulas de médias e os cronogramas, o que indica que eles ainda possuem uma certa dificuldade em raciocínio lógico e matemático sobre o texto recuperado. Outra hipótese é a organização dos PDFs indexados, pois os planos de ensino possuem uma estrutura que pode ser difícil para a vetorização, como dados em tabelas e colunas, ruído de cabeçalhos e rodapés repetitivos que não representam conteúdo significativo. Isso faz com que o contexto recuperado não seja tão preciso, e dificulta a resposta certa da SLM.

Figura 6.3 – Gráfico que mostra a porcentagem de acertos em cada categoria usando RAG.



Fonte: Autor

Portanto, a técnica pareceu não possuir grande eficácia em tarefas de raciocínio lógico/matemático em comparação com a mera recuperação de fatos. E, mesmo que o RAG ajude a encontrar a informação, o modelo ainda precisa de capacidade para interpretar os números corretamente.

Comparando as categorias de conhecimento:

- Informações do site do INF (História e Fatos): O RAG é crucial aqui para evitar erros em datas específicas (ex: 1989 vs 1972).
- Regimentos (Regras Administrativas): Esta é a área onde a precisão é vital. O RAG permite que os modelos identifiquem corretamente sanções disciplinares e requisitos de estágio que são específicos da UFRGS e impossíveis de "adivinhar" sem consulta ao documento.

- Plano de Ensino (Fórmulas e Pesos): Esta categoria parece ser a mais desafiadora. Mesmo com RAG, alguns modelos ainda erram cálculos de pesos e médias (ex: Llama 3.2 errando o peso de uma verificação na disciplina de Arquitetura).

6.4 Análise Geral dos Resultados

A análise dos resultados mostrou que o uso da técnica de *Retrieval-Augmented Generation* (RAG) foi essencial para tornar viável o uso de modelos de linguagem de pequeno porte no ambiente acadêmico. Sem essa técnica, os modelos respondiam apenas com o conhecimento que já possuíam, o que gerava muitas respostas incorretas e informações imprecisas sobre a UFRGS. Além de serem informações extremamente específicas sobre o Instituto de Informática, algumas informações são de data posterior a quando o modelo foi treinado. Por outro lado, ao utilizar o RAG, o melhor resultado foi obtido pelo modelo Qwen 2.5 (3B), que aumentou sua taxa de acertos em 48%, chegando a 88% de precisão. Isso mostra que fornecer informações de contexto não apenas melhora a qualidade das respostas, mas também faz com que um modelo genérico consiga atuar como um assistente confiável para responder dúvidas da comunidade acadêmica.

Na comparação entre os modelos, pode-se observar que o desempenho não depende apenas da quantidade de parâmetros, mas principalmente da capacidade de interpretar o contexto fornecido. Os modelos com cerca de 3 a 3,8 bilhões de parâmetros, como o Qwen e o Phi 3.5, apresentaram os melhores resultados. Já o SmolLM2 (1.7B), mesmo recebendo as mesmas informações por meio do RAG, alcançou apenas 38,7% de acertos. E o modelo Mistral de 7 bilhões de parâmetros, mesmo tendo mais que o dobro de parâmetros que o modelo Qwen, obteve um resultado de acertos inferior.

7 CONCLUSÃO

Este trabalho apresentou o desenvolvimento de um Chatbot baseado em *Small Language Models* utilizando a técnica de *Retrieval-Augmented Generation* para auxiliar alunos do Instituto de Informática da UFRGS na obtenção de informações acadêmicas. Além disso, a solução pode ser executada em computadores comuns, com baixo custo de infraestrutura, oferecendo um canal centralizado para consulta de informações e ajudando a reduzir a demanda por atendimentos administrativos.

As principais contribuições deste trabalho são a centralização de informações acadêmicas do INF em um único ambiente, a redução do tempo necessário para os alunos obterem respostas e a proposição de uma arquitetura simples e escalável. A execução local utilizando Python e Ollama garante maior controle sobre os dados acadêmicos, ajuda a manter a conformidade com a LGPD e evita o envio de informações para serviços em nuvem. Esses pontos são importantes para que o Instituto de Informática possa implementar o sistema de forma segura e com baixo custo de infraestrutura.

A técnica de RAG provou ser essencial para a confiabilidade, transformando o Chatbot de um sistema que "chuta" respostas em um assistente que informa fatos reais baseados em documentos. E quando combinada com as *Small Language Models* (SLMs) juntos são alternativas viáveis aos modelos de grande escala para tarefas de escopo específico em ambientes com limitações de *hardware*. Como o modelo Qwen 2.5, que teve um salto de 48% na acurácia, atingindo 88% de acerto em questões institucionais quando utilizado RAG.

Por fim, os resultados obtidos mostraram que nem todos os modelos de linguagem são igualmente adequados para aplicações baseadas em RAG. Embora todos tenham recebido o mesmo contexto, alguns conseguiram interpretar e utilizar essas informações com muito mais eficiência do que outros. O modelo SmolLM2 (1.7B), por exemplo, não conseguiu entender corretamente o contexto e apresentou erros até em informações simples que estavam no banco de dados. Esses resultados mostram que a escolha do modelo não deve ser feita de forma arbitrária. É fundamental realizar uma avaliação prévia dos modelos disponíveis, pois essa decisão tem impacto direto na qualidade das respostas e na eficácia do chatbot.

7.1 Limitações

Este trabalho apresentou algumas limitações que precisam ser consideradas na análise dos dados e em futuras melhorias do sistema.

Uma das principais limitações foi a falta de acesso às dúvidas reais dos alunos. Tentou-se obter essas informações pelo canal "Chama INF", mas o acesso foi negado por causa das regras da Lei Geral de Proteção de Dados (LGPD), já que as mensagens continham informações pessoais e acadêmicas dos estudantes. Sem esses dados, não foi possível treinar ou testar o sistema usando a linguagem e as dúvidas mais comuns dos usuários.

Outra limitação é que o desempenho dos modelos depende diretamente do computador utilizado nos testes. O protótipo foi executado em uma máquina com processador AMD Ryzen 7 5700X, 32 GB de memória RAM e placa de vídeo NVIDIA GeForce GTX 1070. Essa configuração foi suficiente para os testes iniciais, mas ainda não foi avaliado como o sistema funcionaria com muitos usuários ao mesmo tempo ou em computadores mais simples.

Também pode-se citar que a qualidade dos contextos recuperados do banco vetorial depende diretamente do modelo utilizado para gerar os embeddings. Nesta pesquisa, não foi realizada uma comparação entre diferentes modelos de embeddings, de modo que não foi possível avaliar como essa escolha poderia influenciar a qualidade da recuperação das informações e, consequentemente, o desempenho dos modelos de linguagem.

7.2 Trabalhos futuros

O próximo passo desta pesquisa é levar o protótipo para um ambiente de produção real e realizar testes com usuário. A ideia é implementar o sistema nos servidores do Instituto de Informática da UFRGS, permitindo que os alunos possam utilizar e testar o Chatbot. Essa etapa será importante para verificar como o sistema se comporta com vários acessos ao mesmo tempo, além de permitir a coleta de opiniões dos usuários sobre a utilidade do Chatbot e a clareza das respostas apresentadas na interface desenvolvida em Angular.

Como a arquitetura foi desenvolvida de forma modular usando o Framework LangChain, o sistema permite trocar os modelos de linguagem de maneira rápida e simples. Pesquisas futuras podem testar novos modelos pequenos (SLMs) que forem lançados,

comparando não apenas a qualidade das respostas, mas também fatores como tempo de resposta e consumo de memória em diferentes tipos de *hardware*. Isso pode ajudar a encontrar o melhor equilíbrio entre desempenho do modelo e qualidade das respostas, considerando os recursos de *hardware* disponíveis na instituição.

Atualmente, a coleta e o tratamento dos dados do site e dos regimentos são feitos manualmente. Como as informações acadêmicas mudam com frequência, uma melhoria importante seria criar um sistema de atualização automática. Com isso, o banco vetorial Qdrant poderia ser atualizado automaticamente sempre que houvesse mudanças nos sites oficiais ou a publicação de novos arquivos. Desse modo, o Chatbot conseguiria fornecer informações sempre atualizadas, evitando respostas desatualizadas.

Portanto, os trabalhos futuros apresentados buscam transformar o protótipo desenvolvido em uma solução efetivamente utilizada pela comunidade acadêmica. Espera-se que a solução possa evoluir continuamente, acompanhando os avanços dos modelos de linguagem e as mudanças nas informações institucionais.

REFERÊNCIAS

BOYKIS, V. What are embeddings. **10.5281/zenodo**, v. 8015029, 2023.

CAHN, J. Chatbot: Architecture, design, & development. **University of Pennsylvania School of Engineering and Applied Science Department of Computer and Information Science**, v. 46, 2017.

CHANG, T. A.; BERGEN, B. K. Language model behavior: A comprehensive survey. **Computational Linguistics**, v. 50, n. 1, p. 293–350, 2024.

CHEN, L.; VAROQUAUX, G. What is the role of small models in the llm era: A survey. **arXiv preprint arXiv:2409.06857**, 2024.

GEETHA, G. et al. Interpretation and analysis of angular framework. In: **2022 International Conference on Power, Energy, Control and Transmission Systems (ICPECTS)**. [S.l.: s.n.], 2022. p. 1–6.

Google. **Gemini Embedding API: gemini-embedding-001**. 2026. <https://ai.google.dev/gemini-api/docs/models/gemini-embedding-001>. Acesso em: 6 jul. 2026.

GUPTA, A. et al. **Small Language Models (SLMs) Can Still Pack a Punch: A survey (updated 2026)**. 2026. Disponível em: <https://arxiv.org/abs/2501.05465>.

HESSANI, H. S. **LLM Embeddings Explained: A Visual and Intuitive Guide**. 2025.

Hugging Face TB. **SmolLM2-360M**. 2025. <https://huggingface.co/HuggingFaceTB/SmolLM2-360M>. Accessed: 2026-06-15.

INF, I. **INF Apresentação**. 2026. Disponível em: <https://www.inf.ufrgs.br/site/institucional/apresentacao>.

JURAFSKY, D.; MARTIN, J. H. **Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models**. 3rd. ed. [s.n.], 2026. Online manuscript released January 6, 2026. Disponível em: <https://web.stanford.edu/~jurafsky/slp3/>.

LANGCHAIN, I. **LangChain Documentation**. 2022. Disponível em: <https://docs.langchain.com/oss/python/langchain/rag>.

LEWIS, P. et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. **Advances in neural information processing systems**, v. 33, p. 9459–9474, 2020.

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**. [S.l.: s.n.], 2024. p. 388–393.

LU, Z. et al. Small language models: Survey, measurements, and insights. **arXiv preprint arXiv:2409.15790**, 2024.

MAVROUDIS, V. Langchain. 2024.

Meta AI. **Llama 3.2 3B**. 2024. <https://huggingface.co/meta-llama/Llama-3.2-3B>. Accessed: 2026-06-15.

Microsoft. **Phi-3.5-mini-instruct**. 2024. <https://huggingface.co/microsoft/Phi-3.5-mini-instruct>. Accessed: 2026-06-15.

Mistral AI. **Mistral-7B-v0.1**. 2023. <https://huggingface.co/mistralai/Mistral-7B-v0.1>. Accessed: 2026-06-15.

MONIGATTI, L. **Retrieval-Augmented Generation (RAG): From Theory to LangChain Implementation**. 2023. <https://www.leoniemonigatti.com/blog/retrieval-augmented-generation-langchain.html>. Acesso em: 24 fev. 2025.

NAVEED, H. et al. A comprehensive overview of large language models. **ACM Transactions on Intelligent Systems and Technology**, ACM New York, NY, v. 16, n. 5, p. 1–72, 2025.

OCKERMAN, S. et al. Exploring distributed vector databases performance on hpc platforms: A study with qdrant. In: **Proceedings of the SC'25 Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis**. [S.l.: s.n.], 2025. p. 575–581.

Ollama. **Ollama Documentation**. 2026. <https://docs.ollama.com/>. Acesso em: 14 jun. 2026.

OpenAI. **openai/gpt-oss-20b**. Hugging Face, 2025. Disponível em: <https://huggingface.co/openai/gpt-oss-20b>.

Python Software Foundation. **Python 3 Reference Manual**. 2026. Versão 3.13. Disponível em: <https://www.python.org/>.

Qwen Team. **Qwen2.5: A Party of Foundation Models!** 2024. <https://qwen.ai/blog?id=qwen2.5>. Accessed: 2026-06-15.

RABHI, A. **Optimal RAG Chunking: 8 Strategies & Real Benchmarks**. 2026. <https://ianas.fr/en/blog/2026/04/15/chunking-optimal-rag/>. Acessado em 6 de julho de 2026.

REN, X. et al. **PanGu-Σ: Towards Trillion Parameter Language Model with Sparse Heterogeneous Computing**. 2023. Disponível em: <https://arxiv.org/abs/2303.10845>.

SWACHA, J.; GRACEL, M. Retrieval-augmented generation (rag) chatbots for education: A survey of applications. **Applied Sciences**, v. 15, n. 8, 2025. ISSN 2076-3417. Disponível em: <https://www.mdpi.com/2076-3417/15/8/4234>.

UFRGS. **Ciência da Computação - Currículo**. 2026. <https://www.ufrgs.br/site/ensino/graduacao/ciencia-da-computacao/>. Acesso em: 6 jul. 2026.

UFRGS, I. **Informações básicas aos alunos**. 2026. <https://www.inf.ufrgs.br/site/graduacao/informacoes-basicas-aos-alunos/>. Acesso em: 6 jul. 2026.

VAKAYIL, S. et al. Rag-based llm chatbot using llama-2. In: **2024 7th International Conference on Devices, Circuits and Systems (ICDCS)**. [S.l.: s.n.], 2024. p. 1–5.

APÊNDICE A — PERGUNTAS ELABORADAS

Neste Apêndice, são apresentadas as perguntas utilizadas na avaliação dos modelos de linguagem. Cada linha contém: o número da pergunta, para fins de referência; a classificação da pergunta, conforme descrita na Subseção 4.3.2; o arquivo de origem, com o objetivo de possibilitar eventual verificação; o enunciado da pergunta com suas respectivas alternativas; e, por fim, a alternativa correta.

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
1	SiteINF	Ciência da Computação.txt	Em que ano o Bacharelado em Ciência da Computação foi criado oficialmente? A) 1972; B) 1983; C) 1988; D) 1995; E) 1989;	B
2	SiteINF	Apresentação.txt	Qual a área física ocupada pela incubadora CEI do Instituto de Informática? A) Exatamente 500 m^2; B) Cerca de 800 m^2; C) Menos de 1.000 m^2; D) Superior a 1.000 m^2; E) Exatamente 2.000 m^2;	D
3	SiteINF	Estrutura.txt	Qual entidade estudantil está listada na estrutura do Instituto de Informática? A) Empresa Júnior IDE; B) Diretório Central dos Estudantes; C) União Nacional dos Estudantes; D) Conselho de Reitores; E) Grêmio de Engenharia Civil;	A
4	SiteINF	PPGIE.txt	Qual o objetivo geral do Programa de Pós-Graduação em Informática na Educação (PPGIE)? A) Formar apenas programadores de jogos; B) Pesquisar hardware de baixo custo; C) Substituir professores por robôs; D) Desenvolver sistemas de banco de dados; E) Propiciar estudos aprofundados na área de Informática na Educação;	E
5	SiteINF	Especializações.txt	Qual destes cursos de especialização possui edição prevista para 2024-2026? A) Big Data & Data Science; B) Internet das Coisas; C) Engenharia de Software para Aplicações de Ciência de Dados; D) Segurança Cibernética; E) Gestão de TI;	C

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
6	SiteINF	Informações básicas...	A quem se aplica o Código Disciplinar Discente (CDD) da UFRGS? A) Apenas a alunos de pós-graduação; B) Somente a alunos formandos; C) A todos os discentes com vínculo ativo ou matrícula trancada; D) Exclusivamente a alunos bolsistas; E) Apenas a alunos presenciais;	C
7	SiteINF	Apresentação.txt	Em que data o Instituto de Informática (INF) da UFRGS foi fundado? A) 9 de novembro de 1989; B) 20 de dezembro de 1989; C) 1 de janeiro de 1990; D) 15 de agosto de 1975; E) 9 de novembro de 1970;	A
8	SiteINF	História.txt	Em que ano a Sociedade Brasileira de Computação (SBC) transferiu sua sede para o INF? A) 1989; B) 1991; C) 1996; D) 1974; E) 1998;	E
9	SiteINF	Descrição.txt	Em que ano o Instituto de Informática foi transferido para o Campus do Vale? A) 1983; B) 1991; C) 1996; D) 2003; E) 1970;	B
10	SiteINF	Projetos de Extensão.txt	Qual o foco principal das atividades do projeto LoBoGames? A) Desenvolvimento de apps; B) Programação de robôs; C) Podcasts sobre tecnologia; D) Uso de jogos lógicos de tabuleiro; E) Inclusão digital;	D
11	SiteINF	Apresentação.txt	Com quais países o INF desenvolve notadamente projetos de cooperação internacional? A) China e Japão; B) Argentina e Uruguai; C) Austrália e Canadá; D) Rússia e Índia; E) França, Alemanha e Estados Unidos;	E
12	SiteINF	Direção.txt	Quem é o atual Diretor do Instituto de Informática (gestão 2024-2028)? A) Flávio Rech Wagner; B) Luciano Paschoal Gaspary; C) Philippe Navaux; D) Luís da Cunha Lamb; E) Roberto Tom Price;	B
13	SiteINF	COMGRAD.txt	Quantos eixos estruturantes compõem o Plano de Ação da COMGRAD para 2025? A) Dois; B) Três; C) Quatro; D) Cinco; E) Seis;	D

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
14	SiteINF	PGMICRO.txt	A natureza multidisciplinar do PGMicro exigiu a conjugação de quantas unidades acadêmicas? A) Quatro unidades; B) Duas unidades; C) Oito unidades; D) Uma unidade; E) Seis unidades;	A
15	SiteINF	Departamentos.txt	Quantos departamentos são responsáveis pelas atividades de ensino no INF? A) Um; B) Três; C) Dois; D) Quatro; E) Cinco;	C
16	SiteINF	História.txt	O que foi criado em 1968, marcando o início da história da informática na UFRGS? A) CPD (Centro de Processamento de Dados); B) Bacharelado em Ciência da Computação; C) Curso de Engenharia de Computação; D) Centro de Empreendimentos; E) Laboratório LICIA;	A
17	SiteINF	Projetos de Extensão.txt	Qual projeto foca no recondicionamento de computadores para empréstimo a estudantes? A) Program.ada; B) FalaINF; C) Reconecta UFRGS; D) INF-Games; E) LoBoGames;	C
18	SiteINF	Engenharia de Computação.txt	Calcula-se que qual porcentagem de processadores existentes esteja embutida (embarcada) em equipamentos? A) 50%; B) 98%; C) 75%; D) 25%; E) 10%;	B
19	SiteINF	História.txt	Em que ano o PPGC alcançou a nota 6 na CAPES pela primeira vez? A) 1973; B) 1989; C) 1993; D) 2007; E) 2013;	D
20	SiteINF	NAU-INF.txt	Quem é o atual Coordenador do NAU-INF (Gestão 2024-2027)? A) Renan Maffei; B) Lucio Mauro Duarte; C) Bruno Grisci; D) Giesa Carla da Silva; E) Renan de Queiroz Maffei;	E
21	SiteINF	Ingresso.txt	Quantas vagas anuais totais são oferecidas para Ciência da Computação via Sisu? A) 70; B) 100; C) 22; D) 30; E) 8;	D
22	SiteINF	NDE.txt	Quem é o atual Coordenador do NDE/CIC (Núcleo Docente Estruturante)? A) Rodrigo Machado; B) Leila Ribeiro; C) André Inácio Reis; D) Álvaro Freitas Moreira; E) Eduardo Simões Lopes Gastal;	E

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
23	SiteINF	Perguntas Frequentes.txt	Qual a duração prevista para o curso de Engenharia de Computação? A) 4 anos; B) 4,5 anos; C) 5 anos; D) 6 anos; E) 3,5 anos;	C
24	SiteINF	Licenciatura...txt	Qual o prazo para integralização curricular da Licenciatura em Computação e Robótica? A) 4 anos; B) 2 anos; C) 6 anos; D) 3 anos; E) 5 anos;	A
25	SiteINF	História.txt	Quem projetou o primeiro modem brasileiro em 1978? A) Philippe Navaux; B) Juergen Rochol; C) Clesio Saraiva; D) Raul Weber; E) Otacílio de Souza;	B
26	SiteINF	Ingresso.txt	Quantas vagas novas anuais são oferecidas para Engenharia da Computação? A) 100; B) 60; C) 160; D) 30; E) 70;	B
27	SiteINF	Localização.txt	Em qual cidade está localizado o Instituto de Informática da UFRGS? A) Porto Alegre; B) Curitiba; C) Florianópolis; D) São Paulo; E) Belo Horizonte;	A
28	SiteINF	Centro de Empreendimentos...	Qual o nível de maturidade do CEI na certificação CERNE da Anprotec? A) Nível 1; B) Nível 3; C) Nível 4; D) Nível 5; E) Nível 2;	E
29	SiteINF	Conselho.txt	Quem ocupa a vaga de representante discente da graduação no CONINF? A) Maria Flávia Tondo; B) Alexsander Ribeiro; C) Adriel de Souza; D) Carlos Alberto Júnior; E) Tiago Ferreira;	C
30	SiteINF	Licenciatura...txt	Qual resolução do MEC prevê as diretrizes para a Licenciatura em Computação? A) Resolução nº 4/82; B) Resolução nº 19/2011; C) Resolução nº 001/2013; D) Resolução nº 5, de 16 de novembro de 2016; E) Resolução nº 32/98;	D
31	SiteINF	Projetos de Extensão.txt	Qual o objetivo principal do projeto "Program.ada"? A) Criar algoritmos; B) Desenvolver jogos; C) Capacitar e integrar mulheres na Computação; D) Traduzir manuais; E) Consertar hardware;	C

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
32	SiteINF	PGMICRO.txt	Qual destes laboratórios do Instituto de Física colabora com o PGMICRO? A) Laboratório de Laser; B) Centro de Microscopia; C) Laboratório de Filmes Finos; D) Laboratório de Implantação Iônica; E) Laboratório de Sistemas Digitais;	D
33	SiteINF	Grade Curricular.txt	Qual é a carga horária total do curso de Ciência da Computação? A) 2000 horas; B) 3240 horas; C) 4500 horas; D) 1500 horas; E) 5000 horas;	B
34	SiteINF	Projetos de Extensão.txt	Quem é o coordenador do projeto de extensão "INF-Games"? A) Leticia Machado; B) Erika Cota; C) Anderson Tavares; D) Weverton Cordeiro; E) Joel Luis Carbonera;	E
35	SiteINF	PGMICRO.txt	Em que ano o curso de Doutorado em Microeletrônica foi reconhecido na UFRGS? A) 2003; B) 2007; C) 2002; D) 2006; E) 2010;	A
36	SiteINF	Perguntas Frequentes.txt	Qual o peso da prova de Matemática no Vestibular para Ciência da Computação? A) 3; B) 1; C) 2; D) 4; E) 5;	A
37	SiteINF	Intercâmbio.txt	Em que ano foi criado o laboratório LICIA (parceria com a França)? A) 1970; B) 2005; C) 1996; D) 2020; E) 2011;	E
38	SiteINF	Ciência da Computação.txt	Em que ano o curso de Ciência da Computação foi reestruturado para a forma atual? A) 1983; B) 1989; C) 1991; D) 1995; E) 1999;	D
39	SiteINF	Intercâmbio.txt	O laboratório LICIA é uma parceria entre o INF e qual instituição francesa? A) Sorbonne; B) Laboratoire d'Informatique de Grenoble (LIG); C) École Normale Supérieure; D) INPG; E) Universidade de Paris;	B
40	SiteINF	Comissão de Pesquisa.txt	Quem é a coordenadora da atual Comissão de Pesquisa (2025-2027)? A) Luciana Porcher Nedel; B) Erika Fernandes Cota; C) Karina Kohl Silveira; D) Taisy Silva Weber; E) Viviane Pereira Moreira;	C
41	SiteINF	Apresentação.txt	Qual a nota máxima (obtida pelo PPGC) nas últimas avaliações da CAPES/MEC? A) 4; B) 5; C) 6; D) 10; E) 7;	E

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
42	SiteINF	NAU-INF.txt	O que significa a sigla NAU no contexto do Instituto de Informática? A) Núcleo de Apoio Universitário; B) Novo Agrupamento de Usuários; C) Núcleo de Avaliação da Unidade; D) Núcleo de Automação Urbana; E) Nível de Aprendizado Unificado;	C
43	SiteINF	PPGC.txt	Quantas áreas de concentração integram as atividades de pesquisa do PPGC? A) 5; B) 3; C) 8; D) 10; E) 2;	A
44	SiteINF	Perguntas Frequentes.txt	Qual o total de créditos exigido para a conclusão do curso de Ciência da Computação? A) 152; B) 180; C) 234; D) 216; E) 200;	D
45	SiteINF	PPGIE.txt	Em qual plano o Centro de Excelência em Informática na Educação foi proposto? A) Plano Diretor; B) II Plano Nacional de Informática (II PLANIN); C) Plano Decenal; D) Plano de Expansão; E) I Plano de Metas;	B
46	SiteINF	COMGRAD.txt	Quem é o Coordenador da COMGRAD-CIC para a gestão 2025-2027? A) Marcelo Walter; B) Sérgio Luis Cechin; C) Raul Fernando Weber; D) Álvaro Freitas Moreira; E) Daltro José Nunes;	D
47	SiteINF	PPGC.txt	Desde que ano existe o Mestrado em Ciência da Computação no PPGC? A) 1989; B) 1973; C) 1968; D) 1996; E) 1983;	B
48	SiteINF	Perguntas Frequentes.txt	Onde ocorrem as aulas do curso de Engenharia de Computação? A) Apenas no Campus Centro; B) Apenas no Campus do Vale; C) Campus do Vale e Campus Centro; D) Campus Litoral Norte; E) Prédio 43412 apenas;	C
49	SiteINF	Comissão de Pesquisa.txt	Quantos projetos de pesquisa ativos o Instituto de Informática possui atualmente? A) Exatamente 10; B) Mais de 500; C) Cerca de 20; D) Menos de 50; E) Mais de 100;	E
50	SiteINF	História.txt	Em que ano foi criado oficialmente o Instituto de Informática (INF)? A) 1989; B) 1973; C) 1983; D) 1996; E) 2003;	A

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
51	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	De acordo com os princípios do CDD, qual destes rege a conduta discente? A) Convivência ética; B) Lucratividade comercial; C) Competição individualista; D) Segregação por desempenho; E) Autonomia financeira privada;	A
52	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	Qual é o período mínimo e máximo estabelecido para a sanção de suspensão? A) 1 a 30 dias; B) 10 a 60 dias; C) 3 a 180 dias; D) 30 a 90 dias; E) 5 a 120 dias;	C
53	Regimentos	Diretrizes curriculares... UFRGS.pdf	Como as diretrizes curriculares definem o conceito de currículo na UFRGS? A) Lista de livros obrigatórios; B) Expressão de um projeto pedagógico; C) Tabela de preços de disciplinas; D) Manual de normas de segurança; E) Catálogo de cursos técnicos;	B
54	Regimentos	Normas para acompanha-mento...pdf	O que significa a sigla NCA no contexto de desempenho acadêmico? A) Notas de Concurso Acadêmico; B) Nível de Conhecimento Aplicado; C) Número de Créditos Acumulados; D) Novo Cadastro de Alunos; E) Número de Créditos Aprovados em um período;	E
55	Regimentos	Normas para admissão... de alunos visitantes.pdf	Qual o requisito mínimo de integralização do curso na origem para ser aluno visitante? A) 10%; B) 20%; C) 75%; D) 40%; E) 50%;	D
56	Regimentos	Normas para as atividades com-plementares...pdf	Qual o limite máximo de créditos atribuídos às atividades complementares em relação ao total do curso? A) 5%; B) 10%; C) 15%; D) 20%; E) 50%;	B
57	Regimentos	Normas para o programa de estágio...pdf	Qual a carga horária semanal máxima permitida para o estágio não obrigatório? A) 30 horas; B) 10 horas; C) 20 horas; D) 40 horas; E) 44 horas;	A
58	Regimentos	Regimento.pdf	Qual destes órgãos faz parte da estrutura oficial do Instituto de Informática? A) Conselho de Reitores; B) Diretório Central; C) Empresa de Correios; D) Secretaria de Segurança; E) Centro de Empreendimentos em Informática;	E

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
59	Regimentos	Regulamenta o ordenamento de matrícula...pdf	Qual o valor atribuído ao índice I2 para o grupo de alunos calouros? A) 1; B) 5; C) 7; D) 10; E) 0;	D
60	Regimentos	Regulamentação da dupla diplomação...pdf	A quem compete a homologação dos programas de dupla diplomação após trâmite interno? A) Departamento de Física; B) Reitoria apenas; C) CEPE (Conselho de Ensino, Pesquisa e Extensão); D) Diretório Acadêmico; E) Prefeitura de Porto Alegre;	C
61	Regimentos	Regulamentação do Programa Especial... PEG.pdf	Qual a exigência mínima de professores do quadro da UFRGS em cursos PEG? A) 50%; B) 10%; C) 25%; D) 75%; E) 100%;	A
62	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	A quem se aplica o Código Disciplinar Discente da UFRGS? A) Apenas alunos de graduação; B) Apenas alunos de pós-graduação; C) Somente a alunos formandos; D) Discentes com vínculo ativo ou matrícula trancada; E) Apenas alunos residentes na moradia estudantil;	D
63	Regimentos	Diretrizes curriculares... UFRGS.pdf	Onde o conteúdo ético da formação universitária deve estar presente? A) Apenas em cursos de Direito; B) Perpassando todo o currículo e práticas; C) Somente no vestibular; D) Apenas em seminários extras; E) Exclusivamente no diploma;	B
64	Regimentos	Normas para acompanhamento do desempenho...pdf	O que define o jubilamento de um aluno na UFRGS? A) Reprovação por nota; B) Falta de pagamento; C) Transferência externa; D) Mudança de curso; E) Atingimento do prazo máximo para conclusão;	E
65	Regimentos	Normas para admissão... de alunos visitantes.pdf	Qual a duração máxima permitida para a permanência de um aluno visitante na instituição? A) Um mês; B) Um semestre; C) Dois semestres consecutivos; D) Todo o curso; E) Três anos;	C

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
66	Regimentos	Normas para as atividades complementares...pdf	Quantas horas de atividade de extensão equivalem a 1 crédito complementar? A) 15 horas; B) 30 horas; C) 45 horas; D) 90 horas; E) 60 horas;	E
67	Regimentos	Normas para o programa de estágio...pdf	É dever da Parte Concedente do estágio segurar o estagiário obrigatoriamente contra o quê? A) Acidentes pessoais; B) Desemprego; C) Danos materiais à empresa; D) Erros de programação; E) Roubo de equipamentos;	A
68	Regimentos	Regimento.pdf	Quem atua como Presidente do Conselho do Instituto de Informática (CONINF)? A) O Reitor; B) O Chefe do Departamento; C) O Bibliotecário; D) O Diretor do Instituto; E) O Representante Discente;	D
69	Regimentos	Regulamenta o ordenamento de matrícula...pdf	No cálculo do índice I3 de ordenamento, qual o valor atribuído para o conceito FF? A) 0; B) 1; C) 2; D) 3; E) 6;	B
70	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	Qual autoridade tem a competência exclusiva para aplicar a sanção de desligamento? A) Direção da Unidade; B) Conselho da Unidade; C) Reitoria; D) Ministério da Educação; E) Polícia Federal;	C
71	Regimentos	Regulamenta o aproveitamento de estudos...pdf	Para afastar-se e cursar disciplinas em outra IES, qual o percentual mínimo que o aluno deve ter integralizado? A) 20%; B) 5%; C) 10%; D) 15%; E) 50%;	A
72	Regimentos	Regulamentação do Programa Especial... PEG.pdf	O que é expressamente vedado aos alunos vinculados a cursos do Programa PEG? A) Trancamento de matrícula e transferência interna; B) Uso da biblioteca; C) Participação em extensão; D) Ter aulas no turno da noite; E) Realizar estágios;	A
73	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	Utilizar inteligência artificial para fraudar tarefas acadêmicas é uma infração de que tipo? A) Leve; B) Grave; C) Gravíssima; D) Administrativa; E) Média;	E

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
74	Regimentos	Normas para acompanhamento do desempenho...pdf	Qual o peso atribuído ao conceito D ocorrido pela terceira vez em diante na mesma disciplina? A) Peso 1; B) Peso 1,5; C) Peso 2; D) Peso 3; E) Peso 4;	B
75	Regimentos	Diretrizes curriculares... UFRGS.pdf	Qual metáfora é usada para ilustrar o conhecimento sem hierarquias ou linearidades? A) Pirâmide; B) Escada; C) Rede; D) Edifício; E) Engrenagem;	C
76	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	Atitudes que firam a política contra o assédio na UFRGS são classificadas no mínimo como: A) Infração grave; B) Infração leve; C) Falta de decoro simples; D) Mal-entendido social; E) Infração administrativa média;	A
77	Regimentos	Normas para acompanhamento do desempenho...pdf	Quando o aluno entra oficialmente em regime de observação de desempenho? A) Quando tira um D; B) Após 2 anos de curso; C) Se trancar o semestre; D) Ao ser calouro; E) Quando seu CD é maior que o NCA;	E
78	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	Em quantos dias prescreve o processo disciplinar estudantil na UFRGS? A) 60 dias; B) 120 dias; C) 180 dias; D) 240 dias; E) 365 dias;	D
79	Regimentos	Normas para o programa de estágio...pdf	De acordo com a regulamentação, a realização de estágio na UFRGS gera que tipo de vínculo? A) Empregatício; B) Não acarreta vínculo empregatício; C) Vínculo de servidor público; D) Vínculo vitalício; E) Vínculo de prestação de serviço técnico;	B
80	Regimentos	Regimento.pdf	Qual o prazo máximo para a Direção enviar o Plano de Ação ao Conselho após a posse? A) 30 dias; B) 90 dias; C) 6 meses; D) 1 ano; E) 2 anos;	C
81	Regimentos	Regulamenta o aproveitamento de estudos...pdf	Qual a duração máxima total permitida para o afastamento para estudos externos? A) Um mês; B) Dois semestres letivos; C) Quatro semestres; D) Indeterminada; E) Apenas as férias;	B
82	Regimentos	Regulamenta o ordenamento de matrícula...pdf	Qual o valor atribuído ao índice I1 (seriação aconselhada) para alunos calouros? A) 0; B) 10; C) 100; D) 1; E) SU;	D

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
83	Regimentos	Regulamentação da dupla diplomação...pdf	Através de qual modalidade os alunos em dupla diplomação conservam o vínculo com a UFRGS? A) Matrícula normal; B) Licença prêmio; C) Abandono justificado; D) Matrícula em trâmite; E) Afastamento para Realização de Estudos;	E
84	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	O CDD se aplica a infrações cometidas em quais espaços físicos ou virtuais? A) Apenas salas de aula; B) Somente no Restaurante Universitário; C) Apenas em Porto Alegre; D) Dependências da Universidade e locais de fazer universitário; E) Apenas em prédios tombados;	D
85	Regimentos	Normas para acompanhamento do desempenho...pdf	Quando pode ser concedido um semestre adicional para evitar o jubilamento do aluno? A) Para qualquer aluno que solicitar; B) Se o aluno for monitor; C) Se for possível integralizar o currículo em um único semestre; D) Somente para bolsistas; E) Em caso de doença comprovada;	C
86	Regimentos	Normas para as atividades complementares...pdf	O que é vedado à COMGRAD em relação às atividades complementares na graduação? A) Implementá-las; B) Coordená-las; C) Dar parecer de mérito; D) Determinar atividade complementar como obrigatória; E) Registrar no histórico;	D
87	Regimentos	Normas para o programa de estágio...pdf	Quem atua como supervisor do estagiário obrigatoriamente na Parte Concedente? A) O Reitor; B) Profissional com formação ou experiência na área; C) O Diretor do INF; D) Outro estagiário veterano; E) O Presidente do Grêmio;	B
88	Regimentos	Regimento.pdf	Quem indica a representação discente nos diversos organismos do Instituto de Informática? A) O Diretório Acadêmico da Computação; B) O Chefe do Departamento; C) O Conselho Universitário; D) Os pais dos alunos; E) A empresa Júnior;	A

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
89	Regimentos	Regulamenta o aproveitamento de estudos...pdf	Disciplinas de outra IES podem ser aproveitadas se tiverem equivalência mínima de quê? A) Apenas o nome igual; B) Cor do livro texto; C) Preço da mensalidade; D) Nome do professor; E) Conteúdo programático e carga horária;	E
90	Regimentos	Regulamenta o ordenamento de matrícula...pdf	O que significa o valor "S0"no índice I1 de ordenamento de matrícula? A) Aluno calouro; B) Aluno jubilado; C) Etapa em que o aluno não foi aprovado em todas as obrigatórias; D) Aluno com média 10; E) Matrícula cancelada;	C
91	Regimentos	Regulamentação do Programa Especial... PEG.pdf	Qual o objetivo principal do Programa Especial de Graduação (PEG) na UFRGS? A) Substituir os cursos regulares; B) Cobrar mensalidades; C) Formar técnicos em hardware; D) Criar novos campi em outros países; E) Atender necessidades emergenciais ou inovadoras;	E
92	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	Devassar o conteúdo de correspondência alheia digital ou física é uma infração: A) Leve; B) Grave; C) Gravíssima; D) Média; E) Ética simples;	D
93	Regimentos	Normas para acompanhamento do desempenho...pdf	Qual o peso para o cálculo do Coeficiente de Desperdício em caso de conceito FF (1ª vez)? A) Peso 1; B) Peso 2; C) Peso 3; D) Peso 0; E) Peso 4;	B
94	Regimentos	Diretrizes curriculares... UFRGS.pdf	O caráter político e cultural do currículo na UFRGS visa a consolidação de qual tipo de sociedade? A) Plural e solidária; B) Monárquica; C) Puramente tecnocrata; D) Militarizada; E) Oligárquica;	A
95	Regimentos	Normas para o programa de estágio...pdf	Qual a exigência mínima de Taxa de Integralização (TIM) para realizar estágio não obrigatório? A) 10%; B) 30%; C) 50% da TIM do respectivo curso; D) 70%; E) 100%;	C
96	Regimentos	Regulamenta o ordenamento de matrícula...pdf	Qual valor o índice I2 atribui aos discentes com ingresso por transferência interna? A) 10; B) 6; C) 1; D) 9; E) 5;	B

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
97	Regimentos	Regimento.pdf	O Centro de Empreendimentos em Informática (CEI) é classificado regimentalmente como: A) Departamento de Ensino; B) Empresa Privada; C) Entidade Estudantil; D) Laboratório de Física; E) Órgão Auxiliar do Instituto;	E
98	Regimentos	Regulamenta o aproveitamento de estudos...pdf	De onde devem provir os estudos realizados para fins de aproveitamento oficial na UFRGS? A) De cursos técnicos apenas; B) De cursos de idiomas; C) De experiências de vida; D) De outra instituição de ensino superior; E) De cursos de culinária;	D
99	Regimentos	CÓDIGO DISCIPLINAR DISCENTE.pdf	Onde deve ser registrada obrigatoriamente toda denúncia de provável falta disciplinar? A) Apenas verbalmente; B) Em redes sociais; C) Encaminhada à Direção da Unidade à qual o discente está vinculado; D) No mural do campus; E) No portal de notícias local;	C
100	Regimentos	Regulamentação da dupla diplomação...pdf	Qual o percentual mínimo da carga horária que o aluno deve cursar na UFRGS para obter o diploma? A) 30%; B) 10%; C) 50%; D) 70%; E) 100%;	A
101	PlanoEnsino	INF01032.pdf	Na disciplina de Empreendimento em Informática, qual o peso atribuído ao Trabalho Final no cálculo da média final? A) Peso 2; B) Peso 5; C) Peso 3; D) Peso 7; E) Peso 10;	B
102	PlanoEnsino	INF01043.pdf	De acordo com o plano de Interação Homem-Computador, qual o peso da terceira apresentação (avaliação) no projeto prático? A) 20%; B) 30%; C) 50%; D) 40%; E) 10%;	D
103	PlanoEnsino	INF01046.pdf	Na disciplina de Fundamentos de Processamento de Imagens, quanto tempo do cronograma é dedicado aos Princípios de Análise de Imagens? A) 2 semanas; B) 1 semana; C) 3 semanas; D) 4 semanas; E) 5 semanas;	A

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
104	PlanoEnsino	INF01047.pdf	Qual o limite da carga horária que pode ser ministrada de forma remota na disciplina de Fundamentos de Computação Gráfica? A) 10%; B) 30%; C) 40%; D) 50%; E) 20%;	E
105	PlanoEnsino	INF01048.pdf	Segundo o plano de Inteligência Artificial, qual a fórmula para o cálculo da Nota Final (NF) unindo trabalhos e teoria? A) NF = TP + AT; B) NF = (TP + AT) / 2; C) NF = (TP0.4) + (AT0.6); D) NF = (TP0.6) + (AT0.4); E) NF = AT;	C
106	PlanoEnsino	INF01058.pdf	Na disciplina de Circuitos Digitais, o conceito "A"é atribuído quando a média final (MF) atinge qual valor? A) MF > 9,5; B) MF > 9,2; C) MF > 9,0; D) MF > 8,5; E) MF > 9,8;	B
107	PlanoEnsino	INF01107.pdf	Na disciplina de Introdução à Arquitetura de Computadores, qual o peso do conjunto de verificações (V1, V2 e V3) na média final? A) 0,5; B) 0,6; C) 0,4; D) 0,8; E) 0,2;	D
108	PlanoEnsino	INF01108.pdf	Em Arquitetura e Organização de Computadores I, a média final é calculada através de qual método matemático entre tarefas e trabalhos? A) Média harmônica; B) Média aritmética simples; C) Média geométrica; D) Média ponderada por pesos ímpares; E) Mediana;	A
109	PlanoEnsino	INF01112.pdf	De acordo com o plano de Arquitetura e Organização de Computadores II, qual o peso individual de cada avaliação presencial (AP1 e AP2)? A) Peso 1; B) Peso 3; C) Peso 2; D) Peso 4; E) Peso 5;	C
110	PlanoEnsino	INF01113.pdf	Na disciplina de Organização de Computadores B, qual destes tópicos faz parte da súmula do curso? A) Apenas redes sem fio; B) Exclusivamente banco de dados; C) Apenas sociologia; D) Somente química orgânica; E) Máquinas CISC X RISC;	E
111	PlanoEnsino	INF01120.pdf	Na disciplina de Técnicas de Construção de Programas, qual o percentual de peso da média das provas na formação do conceito final? A) 50%; B) 40%; C) 25%; D) 10%; E) 60%;	A

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
112	PlanoEnsino	INF01121.pdf	Em Modelos de Linguagem de Programação, qual o prazo máximo para a disponibilização das notas após a realização das avaliações? A) 15 dias; B) 20 dias; C) 10 dias; D) 30 dias; E) 7 dias;	B
113	PlanoEnsino	INF01124.pdf	Na disciplina de Classificação e Pesquisa de Dados, o conceito "B"é atribuído em qual intervalo de média final (MG)? A) MG >= 9,0; B) MG >= 6,0; C) MG entre 7,5 e 9,0; D) MG entre 8,0 e 10,0; E) MG < 6,0;	C
114	PlanoEnsino	INF01127.pdf	No curso de Engenharia de Software N, qual o peso total do conjunto de trabalhos práticos na média ponderada? A) Peso 2; B) Peso 4; C) Peso 1; D) Peso 5; E) Peso 3;	D
115	PlanoEnsino	INF01142.pdf	Na disciplina de Sistemas Operacionais I N, qual o peso da média ponderada das atividades e projetos extra-classe (MP)? A) 1/2; B) 3/4; C) 1/4; D) 2/3; E) 1/3;	E
116	PlanoEnsino	INF01145.pdf	Na disciplina de Fundamentos de Banco de Dados, qual o peso atribuído ao Trabalho Prático na nota final? A) 40%; B) 30%; C) 20%; D) 50%; E) 10%;	A
117	PlanoEnsino	INF01147.pdf	Em Compiladores, as atividades do projeto de construção do compilador são divididas em quantas etapas incrementais? A) Três etapas; B) Sete etapas; C) Cinco etapas; D) Dez etapas; E) Duas etapas;	B
118	PlanoEnsino	INF01151.pdf	Para aprovação em Sistemas Operacionais II N, qual a nota mínima obrigatória em cada uma das provas teóricas? A) 5,0; B) 6,0; C) 3,0; D) 4,0; E) 7,0;	C
119	PlanoEnsino	INF01154.pdf	Qual a frequência mínima de presença exigida para aprovação na disciplina de Redes de Computadores N? A) 85%; B) 80%; C) 90%; D) 75%; E) 100%;	D
120	PlanoEnsino	INF01202.pdf	Na disciplina de Algoritmos e Programação - CIC, qual o peso da prova P2 na média final? A) 0,15; B) 0,3; C) 0,5; D) 0,2; E) 0,4;	E

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
121	PlanoEnsino	INF01203.pdf	Na disciplina de Estruturas de Dados, qual o percentual da nota final correspondente a cada uma das três provas (P1, P2 e P3)? A) 25%; B) 20%; C) 30%; D) 15%; E) 10%;	A
122	PlanoEnsino	INF01209.pdf	Em Fundamentos de Tolerância a Falhas, quantos pontos de bônus na média final recebe o aluno com 100% de frequência presencial? A) 1,0 ponto; B) 0,5 ponto; C) 0,2 ponto; D) 2,0 pontos; E) 0,1 ponto;	B
123	PlanoEnsino	INF05005.pdf	Qual ferramenta computacional gratuita é recomendada para o apoio às atividades de Linguagens Formais e Autômatos N? A) Eclipse; B) VS Code; C) JFlap; D) DrRacket; E) Matlab;	C
124	PlanoEnsino	INF05008.pdf	Na disciplina de Fundamentos de Algoritmos, qual o peso das Atividades Autônomas (T) no cálculo da média final? A) 0,35; B) 0,45; C) 0,50; D) 0,20; E) 0,10;	D
125	PlanoEnsino	INF05010.pdf	Em Otimização Combinatória, como é calculada a nova média (m') após a prova de recuperação (nr)? A) m' = nr; B) m' = (nr + n4) / 2; C) m' = (2nr + n4) / 3; D) m' = (nr + 3n4) / 4; E) m' = (3nr + n4) / 4;	E
126	PlanoEnsino	INF05501.pdf	Na disciplina de Teoria da Computação N, qual o peso atribuído à totalização das notas dos trabalhos (MT)? A) 0,4; B) 0,3; C) 0,5; D) 0,2; E) 0,1;	A
127	PlanoEnsino	INF05508.pdf	No curso de Lógica para Computação, qual o valor percentual das Atividades Autônomas na composição da nota final? A) 30%; B) 20%; C) 50%; D) 40%; E) 10%;	B
128	PlanoEnsino	INF05512.pdf	Em Teoria dos Grafos e Análise Combinatória, qual o peso da nota da 2ª prova no cálculo da nota m2? A) 0,5; B) 0,6; C) 0,8; D) 0,2; E) 0,4;	C
129	PlanoEnsino	INF05515.pdf	Na disciplina de Complexidade de Algoritmos - B, qual o peso atribuído à Prova 3 na nota final? A) 3,5; B) 1,5; C) 1,0; D) 4,0; E) 2,0;	D

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
130	PlanoEnsino	INF05516.pdf	Qual o peso do trabalho (T) realizado em grupo na nota final de Semântica Formal N? A) 0,40; B) 0,20; C) 0,25; D) 0,50; E) 0,30;	E
131	PlanoEnsino	MAT01032.pdf	Em Cálculo Numérico A, qual o número máximo de encontros que podem ser realizados na modalidade a distância? A) 7 encontros; B) 3 encontros; C) 15 encontros; D) 10 encontros; E) 1 encontro;	A
132	PlanoEnsino	MAT01353.pdf	Na disciplina de Cálculo e Geometria Analítica I - A, qual a pontuação do teste (T1) na Área 1? A) 7,0 pontos; B) 3,0 pontos; C) 5,0 pontos; D) 1,5 pontos; E) 10,0 pontos;	B
133	PlanoEnsino	MAT01354.pdf	Em Cálculo e Geometria Analítica II - A, para ser aprovado, qual a nota mínima exigida em cada avaliação de área? A) 6,0; B) 5,0; C) 4,5; D) 7,0; E) 3,0;	C
134	PlanoEnsino	MAT01355.pdf	Na disciplina de Álgebra Linear I - A, qual o peso da nota da prova de recuperação geral (R) na média final (MF)? A) 0,3; B) 0,4; C) 0,5; D) 0,6; E) 0,1;	D
135	PlanoEnsino	MAT01375.pdf	Em Matemática Discreta B, se o aluno fizer o exame de recuperação, ele pode obter no máximo qual conceito se M >= 9,0? A) Conceito A; B) Conceito C; C) Conceito D; D) Conceito FF; E) Conceito B;	E
136	PlanoEnsino	MAT02219.pdf	Na disciplina de Probabilidade e Estatística, qual o peso conjunto das três provas parciais na Nota Final? A) 80%; B) 20%; C) 50%; D) 60%; E) 100%;	A
137	PlanoEnsino	QUI99009.pdf	Em Introdução ao Empreendedorismo e Inovação, qual o peso das atividades individuais desenvolvidas no Moodle? A) 50%; B) 30%; C) 20%; D) 10%; E) 75%;	B
138	PlanoEnsino	TRABALHO DE CONCLUSÃO DE CURSO.pdf	Qual o tempo máximo destinado para a apresentação do TCC pelo aluno perante a banca? A) 15 minutos; B) 20 minutos; C) 30 minutos; D) 45 minutos; E) 60 minutos;	C

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
139	PlanoEnsino	INF01108.pdf	Na disciplina de Arquitetura e Organização de Computadores I, o conceito "B"é dado quando a média MF está em qual faixa? A) MF >= 80%; B) 60% <= MF < 70%; C) MF >= 90%; D) 70% <= MF < 80%; E) MF < 60%;	D
140	PlanoEnsino	INF01147.pdf	Em Compiladores, qual o valor máximo limitado para a Nota Mínima de Recuperação (NMR)? A) 6,0; B) 8,0; C) 5,0; D) 7,5; E) 10,0;	E
141	PlanoEnsino	MAT01354.pdf	Em Cálculo e Geometria Analítica II - A, qual a média mínima (M) necessária para o conceito "A"? A) M >= 9,0; B) M >= 8,0; C) M >= 7,5; D) M >= 9,5; E) M = 10,0;	A
142	PlanoEnsino	INF05005.pdf	Para aprovação na recuperação de Linguagens Formais e Autômatos N, qual o aproveitamento mínimo exigido? A) 50%; B) 60%; C) 70%; D) 75%; E) 80%;	B
143	PlanoEnsino	INF01046.pdf	Em Fundamentos de Processamento de Imagens, qual o peso da nota de participação (PA) na média geral? A) 0,2; B) 0,5; C) 0,1; D) 0,3; E) 0,4;	C
144	PlanoEnsino	INF01203.pdf	Na disciplina de Estruturas de Dados, qual o percentual da nota final correspondente ao Trabalho Final (TF)? A) 10%; B) 25%; C) 20%; D) 15%; E) 50%;	D
145	PlanoEnsino	MAT01353.pdf	Para não ser reprovado direto em Cálculo e Geometria Analítica I - A, as notas NA1 e NA2 devem ser no mínimo: A) 6,0; B) 5,0; C) 3,0; D) 4,5; E) 4,0;	E
146	PlanoEnsino	INF01142.pdf	Em Sistemas Operacionais I N, como é calculada a Nota Mínima de Recuperação (NMR)? A) NMR = 12 - NF; B) NMR = 6 - NF; C) NMR = NF / 2; D) NMR = 10; E) NMR = 9 - NF;	A
147	PlanoEnsino	INF01113.pdf	Qual o conceito atribuído para uma média final total inferior a 6,0 em Organização de Computadores B? A) Conceito FF; B) Conceito D; C) Conceito C; D) Conceito B; E) Recuperação;	B

Nº	Classificação	Arquivo de Origem	Pergunta	Alternativa Correta
148	PlanoEnsino	INF01202.pdf	Em Algoritmos e Programação - CIC, qual o peso das atividades práticas (AP) na média final? A) 0,3; B) 0,4; C) 0,15; D) 0,2; E) 0,1;	C
149	PlanoEnsino	QUI99009.pdf	Na disciplina de Introdução ao Empreendedorismo e Inovação, qual o peso da presença e participação? A) 10%; B) 30%; C) 50%; D) 20%; E) 40%;	D
150	PlanoEnsino	INF05515.pdf	Qual o peso atribuído à nota das Atividades Autônomas (AA) em Complexidade de Algoritmos - B? A) 0,4; B) 0,3; C) 0,2; D) 0,05; E) 0,1;	E

APÊNDICE B — RESPOSTAS RECEBIDAS

Abaixo estão as alternativas respondidas por cada modelo com os dois métodos utilizados (Direto ou com RAG). Para facilitar a visualização, as respostas corretas utilizando RAG estão com o fundo destacado em verde claro.

Nº	Alternativa Correta	Qwen (3B)		Mistral (7B)		Phi (3.8)		Llama (3B)		Smoll (1.7)	
		Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG
1	B	B	B	C	B	E	B	E	B	A	B
2	D	D	D	C	D	C	D	C	D	A	C
3	A	A	A	C	A	C	A	C	E	A	B
4	E	E	E	E	E	E	E	E	E	E	E
5	C	A	C	C	C	C	C	E	C	A	C
6	C	C	C	C	C	C	C	C	C	A	C
7	A	D	A	C	A	A	A	E	A	A	A
8	E	C	E	C	B	B	B	C	E	A	B
9	B	B	B	C	B	B	B	E	B	C	A
10	D	E	D	C	C	E	D	E	D	A	E
11	E	E	E	C	E	E	E	E	E	E	C
12	B	A	B	C	B	D	B	E	B	E	A
13	D	C	E	C	C	C	D	C	C	A	C
14	A	B	A	C	A	C	A	E	A	C	C
15	C	B	C	C	C	C	C	C	C	C	C
16	A	A	A	C	A	A	A	E	A	A	A
17	C	C	C	C	C	C	C	C	C	A	C
18	B	B	B	C	B	B	B	E	B	A	A
19	D	D	D	C	C	E	D	E	E	C	B
20	E	E	E	C	E	E	E	E	E	E	E
21	D	E	D	C	C	E	D	C	C	A	A
22	E	E	E	C	D	E	D	C	E	A	A
23	C	A	C	C	C	C	B	C	B	A	C
24	A	D	A	A	A	D	A	D	A	A	C
25	B	C	B	C	C	D	A	C	B	A	C
26	B	C	B	C	C	E	B	C	E	A	C

Nº	Alternativa Correta	Qwen (3B)		Mistral (7B)		Phi (3.8)		Llama (3B)		Smoll (1.7)	
		Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG
27	A	A	A	A	A	A	A	E	A	A	A
28	E	E	E	C	E	C	E	E	E	A	A
29	C	A	C	C	C	A	C	E	E	A	A
30	D	B	D	A	D	D	D	D	D	A	D
31	C	C	C	C	C	C	C	C	C	A	C
32	D	B	B	C	C	D	D	E	C	E	C
33	B	B	B	C	B	B	B	B	B	A	A
34	E	E	E	C	E	E	E	E	E	C	C
35	A	B	D	C	C	B	A	C	D	A	C
36	A	B	A	C	C	B	C	C	C	C	C
37	E	C	E	C	E	B	E	E	E	A	C
38	D	B	D	C	C	B	D	C	D	C	D
39	B	E	D	C	B	B	B	E	B	E	B
40	C	A	C	A	C	E	C	C	C	E	A
41	E	B	E	C	E	E	E	B	E	A	E
42	C	A	C	A	C	E	C	E	C	A	A
43	A	B	A	C	C	C	A	B	B	A	A
44	D	C	D	C	D	E	D	C	D	A	C
45	B	B	B	C	B	B	B	E	B	E	B
46	D	D	D	C	D	A	D	E	E	A	D
47	B	A	B	C	B	E	B	E	B	A	A
48	C	C	C	C	C	C	C	C	C	A	C
49	E	D	E	C	E	B	E	C	E	A	A
50	A	B	A	C	A	B	A	E	A	A	A
51	A	A	A	A	A	A	A	E	E	A	C
52	C	C	C	C	C	D	C	C	C	A	C
53	B	B	B	C	B	B	B	B	B	A	B
54	E	C	E	A	E	E	E	E	E	A	C
55	D	C	D	A	D	A	D	C	D	A	B
56	B	B	B	A	C	D	B	C	E	C	C
57	A	B	A	C	A	A	A	C	A	A	C

Nº	Alternativa Correta	Qwen (3B)		Mistral (7B)		Phi (3.8)		Llama (3B)		Smoll (1.7)	
		Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG
58	E	E	E	C	E	E	E	E	E	E	A
59	D	B	D	C	E	E	E	C	E	A	E
60	C	B	C	C	C	C	C	C	C	A	C
61	A	D	A	C	C	A	A	C	E	A	C
62	D	D	D	C	D	D	D	D	D	A	D
63	B	B	B	B	B	B	B	B	B	A	B
64	E	E	E	E	E	E	E	E	E	A	E
65	C	B	C	C	C	E	C	E	C	A	C
66	E	B	A	C	E	B	A	E	E	A	C
67	A	A	A	C	A	D	A	C	A	E	C
68	D	A	D	C	D	D	D	D	D	E	D
69	B	E	A	C	A	B	E	E	E	C	C
70	C	A	C	C	C	E	C	C	C	E	D
71	A	E	A	A	A	B	A	C	E	C	C
72	A	A	A	A	A	A	A	E	E	A	E
73	E	C	C	B	C	B	C	E	E	E	B
74	B	B	D	C	D	D	D	C	D	C	D
75	C	C	C	C	C	C	C	C	C	E	C
76	A	A	A	C	A	E	A	C	E	A	E
77	E	B	E	C	C	E	E	E	E	A	D
78	D	C	D	C	D	C	D	C	D	A	D
79	B	B	B	A	B	B	B	E	B	E	B
80	C	B	C	C	C	D	C	C	C	A	D
81	B	D	B	C	B	D	B	D	B	A	B
82	D	D	B	C	A	D	B	D	B	A	A
83	E	A	E	C	E	D	E	E	E	E	E
84	D	D	D	C	D	D	D	D	D	A	D
85	C	E	C	C	C	E	C	E	C	A	C
86	D	D	D	C	D	C	D	E	D	E	E
87	B	B	B	C	B	B	B	B	B	A	B
88	A	A	A	C	A	A	A	C	E	A	C

Nº	Alternativa Correta	Qwen (3B)		Mistral (7B)		Phi (3.8)		Llama (3B)		Smoll (1.7)	
		Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG
89	E	E	E	E	E	E	E	E	E	E	E
90	C	C	C	C	C	A	A	E	E	A	A
91	E	E	E	A	E	E	E	E	E	E	E
92	D	C	D	C	B	C	C	C	C	A	C
93	B	D	B	C	C	A	C	C	C	C	C
94	A	A	A	A	A	A	A	E	A	E	A
95	C	C	C	C	C	C	C	C	C	A	C
96	B	C	D	C	C	C	C	C	E	A	C
97	E	E	E	C	E	E	E	C	E	E	C
98	D	D	D	C	D	D	D	E	D	A	D
99	C	C	C	C	C	C	C	C	C	E	C
100	A	E	A	A	C	E	D	E	E	A	C
101	B	C	B	C	C	C	B	C	B	C	C
102	D	C	D	C	C	D	D	C	C	C	C
103	A	B	C	A	C	C	C	E	C	C	C
104	E	D	E	C	E	E	E	E	E	A	A
105	C	B	C	C	C	C	C	B	C	A	C
106	B	D	B	C	C	E	C	C	C	A	C
107	D	C	C	A	C	D	D	C	C	A	A
108	A	D	A	C	C	A	B	B	E	A	A
109	C	C	C	C	C	A	C	C	C	A	C
110	E	E	E	C	E	E	E	E	E	A	A
111	A	B	C	A	C	E	C	C	C	A	C
112	B	D	B	A	B	E	E	C	C	E	A
113	C	B	C	C	C	C	C	C	E	E	C
114	D	B	D	C	C	E	D	C	B	A	D
115	E	E	E	C	D	D	E	E	D	A	C
116	A	C	A	C	C	A	A	C	C	C	C
117	B	A	B	C	B	C	B	C	B	C	B
118	C	B	C	C	C	B	C	E	E	A	B
119	D	D	D	C	D	E	D	C	D	A	C

Nº	Alternativa Correta	Qwen (3B)		Mistral (7B)		Phi (3.8)		Llama (3B)		Smoll (1.7)	
		Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG	Direto	RAG
120	E	B	B	A	C	A	E	C	B	A	C
121	A	C	A	A	C	A	A	C	C	A	C
122	B	C	B	A	B	A	B	E	B	A	C
123	C	C	C	C	C	C	C	C	C	C	C
124	D	B	E	C	C	A	E	C	D	C	C
125	E	E	E	A	E	E	A	E	E	A	C
126	A	D	A	C	C	A	B	C	B	C	C
127	B	D	B	A	C	A	B	C	B	A	C
128	C	B	C	C	C	A	C	C	C	A	A
129	D	C	D	C	C	A	D	C	C	E	D
130	E	D	E	C	C	D	E	C	E	A	C
131	A	A	A	C	C	E	D	E	B	A	D
132	B	C	B	C	B	A	B	C	B	C	B
133	C	B	C	C	C	A	A	E	C	A	C
134	D	D	D	C	C	A	B	C	B	A	B
135	E	B	A	A	A	A	A	E	E	E	C
136	A	B	A	A	C	A	A	C	C	E	D
137	B	D	B	C	C	A	C	C	C	A	C
138	C	C	C	C	C	C	C	C	C	A	C
139	D	D	D	C	D	D	D	E	E	E	B
140	E	C	E	C	E	A	E	E	E	E	A
141	A	B	A	C	A	E	A	C	C	E	C
142	B	D	B	A	C	E	D	E	D	A	B
143	C	D	C	A	C	A	E	C	C	E	C
144	D	B	C	C	C	E	E	C	C	A	C
145	E	E	E	A	A	A	E	E	E	A	C
146	A	B	A	A	A	E	A	C	E	A	A
147	B	E	D	C	B	D	B	E	E	A	D
148	C	D	C	A	C	A	C	C	C	C	C
149	D	B	D	C	C	E	D	C	D	C	C
150	E	E	E	C	C	A	E	C	E	A	C