UNIVERSIDADE ESTADUAL PAULISTA "JÚLIO DE MESQUITA FILHO"
FACULDADE DE CIÊNCIAS - CAMPUS BAURU
DEPARTAMENTO DE COMPUTAÇÃO
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

VINICIUS CASIMIRO DA SILVEIRA

QUESTION-ANSWERING COM MODELOS DE LINGUAGEM BASEADA NA ABORDAGEM DE ESTUDO COM FLASHCARDS

BAURU
Novembro/2025

VINICIUS CASIMIRO DA SILVEIRA

QUESTION-ANSWERING COM MODELOS DE LINGUAGEM BASEADA NA ABORDAGEM DE ESTUDO COM FLASHCARDS

Trabalho de Conclusão de Curso do Curso de Bacharelado em Ciência da Computação da Universidade Estadual Paulista "Júlio de Mesquita Filho", Faculdade de Ciências, Campus Bauru.
Orientador: Prof. Mestre Pedro Henrique Paiola
Coorientador: Mestre Gabriel Lino Garcia

BAURU
Novembro/2025

S587q

Silveira, Vinicius Casimiro da
Question-answering com modelos de linguagem baseada na abordagem de estudo com flashcards / Vinicius Casimiro da Silveira. --
, 2025
48 f.

Trabalho de conclusão de curso (-) - Universidade Estadual Paulista (UNESP), Faculdade de Ciências, Bauru,
Orientador: Pedro Henrique Paiola
Coorientador: Gabriel Lino Garcia

1. Ciência da Computação. 2. Processamento de linguagem natural (Computação). 3. Sistemas de recuperação da informação. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Dados fornecidos pelo autor(a).

Vinicius Casimiro da Silveira

Question-Answering com Modelos de Linguagem baseada na abordagem de estudo com flashcards

Trabalho de Conclusão de Curso do Curso de Bacharelado em Ciência da Computação da Universidade Estadual Paulista "Júlio de Mesquita Filho", Faculdade de Ciências, Campus Bauru.

Banca Examinadora

Prof. Mestre Pedro Henrique Paiola
Orientador
Departamento de Computação
Faculdade de Ciências
Universidade Estadual Paulista "Júlio de Mesquita Filho"

Dra. Simone das G. Domingues Prado
Departamento de Computação
Faculdade de Ciências
Universidade Estadual Paulista "Júlio de Mesquita Filho"

Dr. Clayton Reginaldo Pereira
Departamento de Computação
Faculdade de Ciências
Universidade Estadual Paulista "Júlio de Mesquita Filho"

Bauru, 10 de Novembro de 2025.

Aos meus pais, Raimundo Filho e Angela Maria, que abdicaram de seus próprios sonhos para que eu tivesse a chance de realizar os meus.

Agradecimentos

Agradeço, primeiramente, a Deus, por ter sido minha fortaleza, guiando meus passos e renovando minhas forças e minha fé ao longo de toda esta jornada. Sem Ele, a conclusão deste trabalho não seria possível.

Dedico esta conquista à minha família, meu alicerce. Aos meus pais, Raimundo Filho Lopes Casimiro e Angela Maria da Silveira Casimiro, expresso minha mais profunda gratidão. Seu amor, sacrifício e apoio incondicional tornaram este sonho uma realidade. Ao meu irmão, Willami Casimiro da Silveira, pela amizade e compreensão, que mesmo no silêncio próprio de sua personalidade, me deu seu apoio. Obrigado por serem minha maior inspiração e por nunca me deixarem duvidar do meu potencial, sem vocês nunca chegaria onde cheguei. Agradeço também a todos os meus familiares, pelo carinho e incentivo constantes.

Um agradecimento especial à minha namorada, Sofia Azevedo Rosa, por sua imensa paciência, compreensão e companheirismo. Seu apoio nos momentos de ausência e estresse foi um pilar fundamental para que eu mantivesse o foco e a tranquilidade necessários para concluir esta etapa, além de todos os momentos lado a lado que nos fizeram crescer na faculdade. Sem você, a faculdade seria muito mais difícil. Também agradeço à sua família, Joel Rosa e Claudia Lovison, que me acolheram e trataram como filho, sempre presentes em todos os dias difíceis.

Também agradeço ao meu amigo e professor Rodrigo Sousa Nascimento, que sempre esteve presente durante o colegial, mudou minha visão de mundo com seus conselhos e puxões de orelha e moldou quem sou hoje. Mesmo longe, ele nunca deixou de acreditar em mim. Sem as olimpíadas, nunca teria me desafiado como fiz e não estaria aqui. Além disso, agradeço a todos os professores que tive a oportunidade de escutar e aprender algo, em especial, o professor Clayton Reginaldo Pereira, com quem tive a oportunidade de trabalhar junto e sempre pude contar.

Por fim, agradeço imensamente aos meus orientadores. Ao Prof. Mestre Pedro Henrique Paiola e ao Mestre Gabriel Lino Garcia, pela orientação precisa, pelas contribuições que viabilizaram a execução deste projeto e pela paciência em todo o processo.

A todos que, diretamente ou indiretamente, fizeram parte desta trajetória, meus sinceros agradecimentos.

"O aspecto mais triste da vida atualmente é que a ciência reúne conhecimento mais rápido do que a sociedade reúne sabedoria."

(Isaac Asimov)

Resumo

Grandes Modelos de Linguagem (LLMs) sofrem com conhecimento estático e alucinações. Embora a Geração Aumentada por Recuperação (RAG) atenue isso ao injetar contexto externo, sua abordagem tradicional de segmentação linear (*chunking*) frequentemente introduz ruído contextual. Como alternativa, este trabalho investiga uma arquitetura RAG baseada em *flashcards*, unidades atômicas de Pergunta-Resposta inspiradas na prática de recuperação, utilizando uma indexação assimétrica que vetoriza apenas a pergunta. Para validar esta abordagem, foi realizado um experimento comparando três grupos (LLM Puro, RAG Tradicional e RAG Flashcards) com o modelo Phi-4, em uma tarefa de Múltipla Escolha (MCQA) sobre 293 itens do *dataset* BLUEX (Biologia, Geografia, História). O RAG Tradicional ($88,74\%$) superou marginalmente o RAG Flashcards ($84,30\%$), ambos com ganho mínimo sobre o LLM Puro ($82,94\%$), sugerindo alto conhecimento paramétrico do modelo na tarefa. Contudo, a principal contribuição foi validada na análise de eficiência: o RAG Flashcards alcançou desempenho quase idêntico ao tradicional consumindo apenas $39,69\%$ do custo computacional (média de 1.173 vs 2.957 *tokens* por consulta). Conclui-se que, apesar de não superior em acurácia neste cenário, a abordagem em *flashcards* oferece um balanço de custo-benefício superior à segmentação linear.

Palavras-chave: Geração Aumentada por Recuperação (RAG); Modelos de Linguagem; Flashcards; Prática de Recuperação.

Abstract

Large Language Models (LLMs) suffer from static knowledge and hallucinations. Although Retrieval-Augmented Generation (RAG) mitigates this by injecting external context, its traditional linear segmentation (*chunking*) approach often introduces contextual noise. As an alternative, this work investigates a RAG architecture based on flashcards, atomic Question-Answering units inspired by retrieval practice, using asymmetric indexing that vectorizes only the question. To validate this approach, an experiment was conducted comparing three groups (Pure LLM, Traditional RAG, and Flashcard RAG) with the Phi-4 model, in a Multiple Choice Question Answering (MCQA) task on 293 items from the BLUEX dataset (Biology, Geography, History). The accuracy results were not as expected, with Traditional RAG (88.74%) slightly outperforming Flashcards RAG (84.30%), both with minimal gains over Pure LLM (82.94%), suggesting high parametric knowledge of the model in the task. However, the main contribution was validated in the efficiency analysis: Flashcards RAG achieved almost identical performance to the traditional one while consuming only 39.69% of the computational cost (average of 1.173 vs. 2.957 tokens per query). It is concluded that, although not superior in accuracy in this scenario, the flashcard approach offers a dramatically superior cost-benefit balance to linear segmentation.

Keywords: Retrieval-Enhanced Generation (RAG); Language Models; Flashcards; Retrieval Practice.

Lista de figuras

Lista de tabelas

Lista de abreviaturas e siglas

BERT	*Bidirectional Encoder Representations from Transformers*
CNN	Rede Neural Convolucional
DPR	*Dense Passage Retriever*
GPU	Unidade de processamento gráfico
GPT	*Generative Pre-trained Transformer*
IA	Inteligência Artificial
JSON	*JavaScript Object Notation*
LLaMA	*Large Language Model Meta AI*
LLM	*Large Language Model*
MCQA	*Multiple choice question answering*
MMLU	*Massive Multitask Language Understanding*
PLN	Processamento de Linguagem Natural
QA	*Question Answering*
RAG	*Retrieval-Augmented Generation*
RNN	Rede Neural Recorrente
SLM	*Small Language Model*
SQuAD	*Stanford Question Answering Dataset*
T5	*Text-to-Text Transfer Transformer*

Sumário

1 Introdução

Nos últimos anos, os Grandes Modelos de Linguagem (*Large Language Models* - LLMs) tornaram-se protagonistas em aplicações de Inteligência Artificial (IA), impulsionados pela consolidação do Transformer como arquitetura de referência no Processamento de Linguagem Natural (PLN) (VASWANI et al., 2017). Esses modelos demonstram elevado desempenho em tarefas de geração e compreensão de texto. No entanto, uma das aplicações de maior interesse e desafio são perguntas e respostas, do inglês *Question Answering* (QA), onde a precisão factual é fundamental. É justamente nesse ponto que os LLMs expõem limitações significativas: a natureza estática do conhecimento incorporado durante o treinamento e a consequente tendência a produzir informações desatualizadas ou imprecisas, fenômeno descrito como *alucinação*, comprometem a capacidade de responder com precisão, atualidade e rastreabilidade.

Em resposta a esse quadro, a linha de pesquisa em *Retrieval-Augmented Generation* (RAG) propõe integrar mecanismos de recuperação de informações a modelos generativos (LEWIS et al., 2020). Na arquitetura RAG padrão, documentos-fonte externos são primeiro segmentados em fragmentos de texto (*chunks*), que são então vetorizados e armazenados em um banco de dados. Dada uma consulta, o sistema recupera os *chunks* semanticamente mais relevantes e os fornece como contexto ao LLM, que deve usá-los para formular a resposta. Embora represente um avanço significativo, permitindo atualizar o conhecimento sem re-treinamento e promovendo transparência, a eficácia prática desse modelo depende de como o conhecimento é representado e organizado. Na prática, recuperações redundantes ou ambíguas, resultado de *chunks* mal definidos ou semanticamente difusos, podem saturar o contexto e comprometer a coerência global da resposta, o que se reflete diretamente no desempenho em tarefas de QA.

Assim sendo, este trabalho parte da hipótese de que princípios cognitivos da aprendizagem humana, em especial a prática de recuperação, em inglês *retrieval practice*, podem orientar uma estruturação mais eficaz do conhecimento para uso em RAG. Evidências experimentais clássicas mostram que o ato de recordar ativamente uma informação, em oposição à sua mera releitura, consolida a memória de longo prazo, produz ganhos superiores de retenção e favorece a transferência de conhecimento para novos contextos Roediger Henry L. e Butler (2011).

A metodologia de *flashcards*, que operacionaliza a prática de recuperação por meio de unidades atômicas de estudo e revisão, representa, portanto, um candidato natural para a organização de bases de conhecimento utilizadas em sistemas RAG aplicados a QA. Essa técnica é amplamente reconhecida por sua eficácia na fixação e recuperação de informações, e, quando adaptada a um contexto computacional, pode permitir que um sistema de QA

recupere o conhecimento de forma mais precisa, segmentada e cognitiva, análoga à forma como humanos consolidam memórias.

Com base nessa motivação, esta dissertação investiga se a organização do conteúdo de um sistema RAG em formato de *flashcards* digitais pode otimizar o desempenho de LLMs em tarefas de QA, quando comparados a um RAG normal. Para validar essa hipótese, o contexto abordado será o de questões de vestibulares brasileiros, um domínio que exige conexões multidisciplinares e raciocínio complexo.

A proposta consiste, portanto, em comparar a acurácia das respostas geradas por um modelo de linguagem quando este é auxiliado por duas arquiteturas de recuperação distintas: um RAG tradicional, alimentado por documentos longos, e um RAG estruturado com *flashcards*.

Ao unir os avanços técnicos do RAG (LEWIS et al., 2020) com os princípios cognitivos da prática de recuperação (ROEDIGER HENRY L.; BUTLER, 2011), este trabalho busca aproximar o aprendizado de máquina do aprendizado humano. O objetivo é explorar como estratégias que ajudam humanos a consolidar a memória podem ser usadas para projetar sistemas de IA que precisam consultar conhecimento externo e, principalmente, como essa abordagem impacta a qualidade e a confiabilidade final das respostas.

1.1 Problema

Apesar dos avanços trazidos pelos LLMs e pelo paradigma RAG, ainda persiste um desafio fundamental na forma como o conhecimento é recuperado, representado e integrado às respostas geradas. Os modelos RAG tradicionais, embora combinem memória paramétrica e não paramétrica, frequentemente enfrentam dificuldades em selecionar informações verdadeiramente relevantes e em articular o conteúdo recuperado de forma coesa e precisa. Como consequência, é comum que apresentem redundâncias, inconsistências semânticas e até alucinações factuais, especialmente em tarefas de QA, que exigem exatidão e contextualização rigorosa (LEWIS et al., 2020).

Pesquisas recentes têm aprofundado a compreensão dessas limitações. O estudo de Liu et al. (2023) mostra que, mesmo em modelos capazes de lidar com grandes janelas de contexto, a informação mais relevante tende a ser negligenciada quando posicionada longe do início ou do final do texto. Esse efeito evidencia que, embora os LLMs consigam processar longos contextos, sua capacidade de atenção efetiva é desigual e decai em função da posição das informações dentro da sequência. Isso implica que aumentar o tamanho da janela de contexto não é, por si só, suficiente para melhorar a qualidade das respostas, uma vez que a relevância e a estrutura do conteúdo recuperado desempenham papel mais determinante do que o volume de texto fornecido.

Complementarmente, Zhong et al. (2024) demonstram que a granularidade com que

o conhecimento é dividido e recuperado, ou seja, o tamanho e a coesão semântica dos trechos ou *chunks* utilizados na indexação, influencia diretamente o desempenho de sistemas RAG. O estudo propõe uma abordagem de *Mix-of-Granularity*, na qual diferentes níveis de segmentação são combinados de forma adaptativa para otimizar a qualidade da recuperação e reduzir redundâncias contextuais. Os resultados indicam que a simples fragmentação linear dos documentos, prática comum em implementações RAG, pode prejudicar a precisão factual das respostas e comprometer a integração semântica entre trechos relacionados.

Essas constatações reforçam que o principal gargalo dos sistemas RAG não está apenas na capacidade do modelo de gerar texto, mas sobretudo na forma como as informações são organizadas e estruturadas antes da inferência. A recuperação textual tradicional tende a operar sobre segmentos longos e heterogêneos, que nem sempre correspondem às unidades conceituais de um tema, tornando a seleção de evidências relevante uma tarefa imprecisa. Essa desorganização pode saturar a janela de atenção, reduzir a cobertura de conteúdo relevante e comprometer a coerência global da resposta.

1.2 Justificativa

Uma vez estabelecido o gargalo técnico dos sistemas RAG (Seção 1.1), que se concentra na granularidade e organização do conhecimento, a presente pesquisa justifica-se por propor uma solução inspirada em princípios cognitivos e validada por tendências recentes na própria engenharia de RAG.

A literatura técnica recente começa a convergir para soluções alinhadas à hipótese deste trabalho. O estudo de Raina e Gales (2024), por exemplo, propõe reorganizar as bases de conhecimento em unidades conceituais mínimas, denominadas *atomic units*, projetadas para responder a perguntas específicas de forma direta. A noção de *atomic units* aproxima-se diretamente da lógica dos *flashcards*, ao sugerir que o conteúdo seja armazenado e recuperado em blocos menores, semanticamente autossuficientes e alinhados a uma consulta.

Esse paralelo entre estruturas cognitivas e computacionais encontra respaldo também na literatura da psicologia da aprendizagem. De acordo com Roediger Henry L. e Butler (2011), o ato de recuperar informações de forma ativa, em oposição à mera releitura, fortalece significativamente a consolidação da memória e a capacidade de transferência do conhecimento para novos contextos. Sabendo que, a prática de recuperação, é a base conceitual da metodologia de *flashcards*, que se mostra uma das estratégias mais eficazes para retenção e generalização de conhecimento a longo prazo. Assim, integrar princípios da prática de recuperação à estruturação do conhecimento em sistemas RAG não apenas representa uma inovação técnica, mas também uma aproximação entre aprendizagem humana e inteligência artificial.

Dessa forma, a relevância desta pesquisa se sustenta em dois eixos complementares. No eixo científico, o trabalho busca contribuir para o avanço dos estudos sobre organização e

representação de conhecimento em sistemas RAG, oferecendo uma alternativa inspirada na cognição humana para o problema da granularidade e da contextualização. No eixo prático, pretende-se demonstrar como a aplicação de estruturas do tipo *flashcard* pode melhorar a precisão, a coerência e a rastreabilidade das respostas em tarefas de QA. Em síntese, a dissertação propõe um ponto de convergência entre os princípios da memória humana e os desafios técnicos da geração aumentada por recuperação, explorando como a modelagem inspirada em *retrieval practice* pode tornar os sistemas de IA mais explicáveis, confiáveis e eficientes.

1.3 Objetivos

1.3.1 Objetivo geral

O objetivo geral deste trabalho é investigar o impacto da estruturação de conhecimento em *flashcards* na otimização do desempenho de sistemas de RAG aplicados a tarefas de QA. Busca-se verificar se a organização do conhecimento em unidades atômicas melhora a precisão factual, a relevância contextual e a coerência das respostas geradas por modelos RAG em comparação com arranjos convencionais baseados em fragmentos textuais lineares.

1.3.2 Objetivos específicos

Para atingir o objetivo geral, este estudo propõe os seguintes objetivos específicos:

1. Desenvolver um conjunto de *flashcards* a partir de conteúdos teóricos de uma disciplina específica, utilizando diferentes modelos de linguagem, priorizando a concisão, a clareza e a cobertura dos conceitos-chave necessários à compreensão do conteúdo;

2. Adaptar bancos de dados afim de filtrar apenas as questões das matérias escolhidas, assegurando a correspondência entre perguntas e contextos conceituais relacionados ao domínio temático escolhido;

3. Implementar um sistema de RAG que utilize os *flashcards* desenvolvidos como base de conhecimento estruturado, de modo a auxiliar o modelo de linguagem na tarefa de QA e permitir a recuperação segmentada e contextual de informações;

4. Comparar a performance do modelo de linguagem nas tarefas de QA adaptadas, utilizando métricas como acurácia e custo computacional, considerando três configurações experimentais distintas: (a) Apenas o modelo, sem nenhuma técnica aplicada, (b) modelo integrado a um RAG baseado em documentos longos e (c) modelo integrado a um RAG estruturado com *flashcards*, mensurando o impacto de cada abordagem sobre a precisão e a coerência das respostas;

5. Analisar os benefícios e limitações do uso de *flashcards* como fonte de conhecimento estruturado para modelos de linguagem, discutindo sua aplicabilidade prática em sistemas de QA voltados ao contexto educacional ou assistivo, e as implicações para o desenvolvimento de ferramentas cognitivamente inspiradas em aprendizado automático.

O cumprimento desses objetivos permitirá compreender se princípios derivados da cognição humana, particularmente os que regem a prática de recuperação e o uso de *flashcards*, podem servir como base para o desenvolvimento de sistemas RAG mais eficientes, explicáveis e alinhados com mecanismos de aprendizado de natureza humana.

2 Fundamentação teórica

O presente capítulo tem como objetivo mostrar os principais temas relacionados aos LLMs, ao método de RAG e à aplicação da metodologia de *flashcards* como estratégia de aprimoramento da recuperação de conhecimento.

Nos últimos anos, os LLMs têm se consolidado como um dos pilares da inteligência artificial moderna, impulsionando avanços significativos em tarefas de PLN. Esses modelos, treinados sobre extensos conjuntos de dados textuais, são capazes de gerar respostas coerentes e contextualizadas, mas ainda apresentam limitações quanto à fidelidade das informações produzidas e à capacidade de manter uma memória de longo prazo.

Nesse contexto, o método RAG surge como uma alternativa promissora ao combinar a capacidade generativa dos modelos de linguagem com mecanismos de busca em bases de conhecimento externas, permitindo respostas mais fundamentadas e atualizadas. Entretanto, a eficiência desse processo depende diretamente da forma como as informações são representadas, organizadas e relembradas durante a recuperação, o que evidencia um espaço de melhoria relacionado à gestão da memória contextual.

Com base em princípios cognitivos do aprendizado humano, a metodologia de *flashcards* tem se mostrado eficaz na consolidação do conhecimento por meio da repetição espaçada e da associação ativa entre conceitos. Assim, sua aplicação em sistemas RAG busca aproximar o comportamento dessas arquiteturas de estratégias de estudo humano, reforçando conexões entre contexto e conteúdo, e promovendo uma retenção de informações mais estruturada.

Dessa forma, esta fundamentação teórica está organizada em quatro partes principais:

- os fundamentos e o funcionamento dos grandes modelos de linguagem;
- os sistemas de *question answering*;
- a arquitetura e as limitações do método *Retrieval-Augmented Generation*;
- os aspectos cognitivos e computacionais relacionados à utilização de *flashcards* como mecanismo de otimização da recuperação de conhecimento.

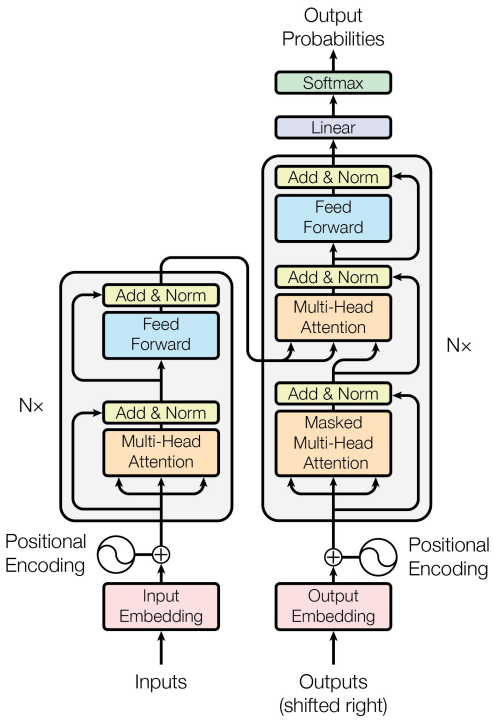
2.1 Arquitetura Transformer e o Mecanismo de Atenção

O artigo *Attention Is All You Need*, proposto por Vaswani et al. (2017), introduziu a arquitetura Transformer, responsável por uma das maiores revoluções no campo do *deep learning* aplicado ao PLN. Diferentemente das abordagens anteriores, baseadas em redes neurais recorrentes (RNNs) ou convolucionais (CNNs), o Transformer elimina a necessidade de

recorrência ao utilizar exclusivamente mecanismos de atenção, permitindo modelar relações de dependência entre *tokens* de forma totalmente paralelizável e eficiente.

A arquitetura, mostrada na figura 1, segue o paradigma *encoder-decoder*, em que o codificador transforma uma sequência de entrada em representações contínuas e o decodificador gera uma sequência de saída com base nessas representações. O componente central do Transformer é o mecanismo de autoatenção (*self-attention*), que calcula a importância relativa de cada *token* em relação aos demais da sequência, permitindo que o modelo capture dependências de longo alcance sem o custo computacional das RNNs.

Figura 1 – Arquitetura original do Transformer



Fonte: Vaswani et al. (2017)

Matematicamente, a atenção é definida como uma função que mapeia um conjunto de *queries* (Q), *keys* (K) e *values* (V) para uma saída ponderada, dada pela equação 1.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

em que d_k representa a dimensionalidade das chaves (*keys*). Essa forma, denominada *atenção por produto escalar escalonado* (*Scaled Dot-Product Attention*), estabiliza o treinamento ao evitar gradientes explosivos para grandes dimensões de *embedding*.

O Transformer introduz ainda o conceito de *atenção multi-cabeça* (*multi-head attention*), no qual múltiplos mecanismos de atenção são executados em paralelo, permitindo que o modelo aprenda representações de diferentes subespaços semânticos simultaneamente. Essa estratégia

amplia a capacidade do modelo em compreender relações complexas entre palavras e contextos, algo essencial para tarefas como tradução automática, sumarização e *question answering*.

Outra contribuição fundamental é o *positional encoding*, que incorpora informações de ordem às representações vetoriais dos *tokens*, uma vez que o modelo não possui recorrência explícita. Vaswani et al. (2017) propõem uma codificação senoidal que permite ao modelo generalizar para comprimentos de sequência não observados durante o treinamento.

2.1.1 Evolução do Transformer

A partir da introdução do Transformer, a área de PLN mudou atingindo novos patamares. O mecanismo de atenção proposto por Vaswani et al. (2017) tornou-se a base para uma nova geração de arquiteturas capazes de compreender e gerar linguagem natural com profundidade semântica e contextual antes inatingível. O principal diferencial dessas arquiteturas é a capacidade de modelar dependências de longo alcance de forma paralela e escalável, eliminando as limitações sequenciais das redes recorrentes.

Com essa base, surgiram modelos como o BERT (*Bidirectional Encoder Representations from Transformers*), proposto por Devlin et al. (2018), que introduziu o aprendizado bidirecional de contexto para tarefas de compreensão textual, e o GPT (*Generative Pre-trained Transformer*), desenvolvido pela OpenAI (RADFORD; NARASIMHAN, 2018), que enfatizou o aprendizado não supervisionado e a geração de texto coerente a partir de predições sequenciais. Posteriormente, arquiteturas como o T5 (*Text-to-Text Transfer Transformer*) (RAFFEL et al., 2019), Large Language Model Meta AI (LLaMA) (TOUVRON et al., 2023) e Mistral (JIANG et al., 2023) consolidaram o uso de grandes corpora textuais e treinamento em larga escala.

Contudo, a principal distinção entre essas arquiteturas recentes e seus predecessores, como o BERT e o T5, não reside apenas em aspectos conceituais ou arquiteturais, mas fundamentalmente em sua escala. Enquanto os modelos anteriores, hoje frequentemente classificados como *Small Language Models* (SLMs), operavam na ordem de centenas de milhões de parâmetros, os modelos contemporâneos atingem dezenas ou mesmo centenas de bilhões.

De acordo com Wei et al. (2022), esse aumento exponencial na escala não resultou apenas em melhorias graduais de desempenho, mas desencadeou um fenômeno qualitativamente distinto, denominado capacidades emergentes (*emergent abilities*). Essas capacidades referem-se a habilidades que não estão presentes em modelos menores e que não podem ser previstas pela simples extrapolação de seu desempenho. Em termos práticos, isso significa que determinadas competências, por exemplo, raciocínio lógico, compreensão contextual complexa ou capacidade de seguir instruções, manifestam-se apenas quando o modelo ultrapassa um determinado limiar de escala, caracterizando uma mudança de fase no comportamento do sistema.

Dessa forma, enquanto modelos menores, como o BERT, necessitavam de *fine-tuning*

específico para executar tarefas como sumarização ou tradução, os LLMs modernos passaram a apresentar comportamento *zero-shot*, isto é, a habilidade de generalizar para tarefas não vistas durante o treinamento. Essa transição evidencia a relação direta entre escala e emergência de comportamento, reforçando a ideia de que aumentos quantitativos em parâmetros e dados podem gerar mudanças qualitativas nas capacidades cognitivas dos modelos, conforme demonstrado empiricamente por Wei et al. (2022).

Os LLMs representam, portanto, uma evolução direta da proposta de Vaswani et al. (2017), onde o salto em escala (bilhões de parâmetros) e o pré-treinamento massivo resultaram em modelos com capacidades emergentes generalistas (WEI et al., 2022). Essas redes demonstraram uma aptidão sem precedentes para aprender relações semânticas complexas e adaptar-se a diferentes contextos, tornando-se ferramentas versáteis em aplicações *zero-shot* que envolvem raciocínio, geração e compreensão de informação.

Contudo, apesar dessa notável capacidade de raciocínio e geração, o poder dos LLMs permanece intrinsecamente ligado ao conhecimento adquirido durante seu treinamento. Essa dependência de uma base de dados interna e estática expõe limitações fundamentais, como a rápida obsolescência do conhecimento e a propensão a gerar informações factualmente incorretas, fenômeno conhecido como alucinação (ZHANG et al., 2023). Foi para solucionar esta lacuna, a separação entre o raciocínio do modelo e o conhecimento factual do mundo, que uma nova arquitetura foi proposta.

2.2 Sistemas de Question Answering

Os sistemas de QA têm como objetivo principal permitir que máquinas compreendam e respondam perguntas formuladas em linguagem natural, fornecendo respostas precisas, concisas e semanticamente adequadas (HIRSCHMAN; GAIZAUSKAS, 2001). Diferentemente dos mecanismos de busca tradicionais, que retornam uma lista de documentos ou trechos relevantes, os sistemas de QA buscam apresentar uma resposta direta, extraída ou gerada a partir de fontes textuais.

Historicamente, o desenvolvimento de sistemas de QA evoluiu de abordagens baseadas em regras e recuperação de informações para modelos estatísticos e, mais recentemente, para arquiteturas neurais profundas impulsionadas pelos LLMs. Segundo Rajpurkar et al. (2016), o avanço de bases de dados de benchmark, como o *Stanford Question Answering Dataset* (SQuAD), possibilitou a avaliação padronizada de modelos em tarefas de compreensão textual, consolidando o QA como um problema de referência para medir a capacidade de raciocínio e generalização dos modelos de linguagem.

De forma geral, os sistemas de QA podem ser classificados em três categorias principais: fechados, abertos e gerativos. No QA fechado, o modelo responde com base em um conjunto de documentos previamente delimitado, frequentemente utilizado em domínios específicos,

como jurídico ou médico. No QA aberto, a resposta é buscada em grandes bases textuais ou na web, exigindo que o sistema execute tarefas de recuperação, seleção e síntese de informação. Já o QA abstrativo, amplamente utilizado em LLMs e no paradigma RAG, envolve a geração de respostas completas e contextuais, integrando informações recuperadas de fontes externas ao conhecimento paramétrico do modelo (LEWIS et al., 2020).

Além dessas categorias funcionais, uma formatação de tarefa crucial para a avaliação de modelos é o *Multiple-Choice Question Answering* (MCQA). Neste formato, o desafio do LLM não é a geração de texto livre, mas sim a seleção da resposta correta a partir de um conjunto finito de alternativas fornecidas. A principal vantagem do MCQA, e a razão de sua adoção em *datasets* de *benchmark* influentes como o *Massive Multitask Language Understanding* (MMLU) (HENDRYCKS et al., 2020), é a sua capacidade de permitir uma avaliação objetiva, quantitativa e escalável. A avaliação é baseada em métricas de acurácia, em vez de julgamentos qualitativos e subjetivos sobre a qualidade do texto gerado.

O processo de QA moderno é geralmente estruturado em três etapas: interpretação da pergunta, recuperação de passagens relevantes e formulação da resposta. Na primeira etapa, a entrada em linguagem natural é transformada em uma representação semântica compreensível pelo modelo, permitindo identificar entidades, intenções e contextos. A segunda etapa envolve a busca de trechos relevantes em uma base de conhecimento, utilizando métodos de recuperação lexical ou vetorial. Finalmente, na etapa de geração ou extração, o modelo sintetiza uma resposta a partir dos documentos selecionados. No contexto de modelos como o RAG, essas etapas são executadas de maneira integrada, de modo que o mecanismo de atenção do modelo combine informações recuperadas com seu próprio contexto interno de conhecimento.

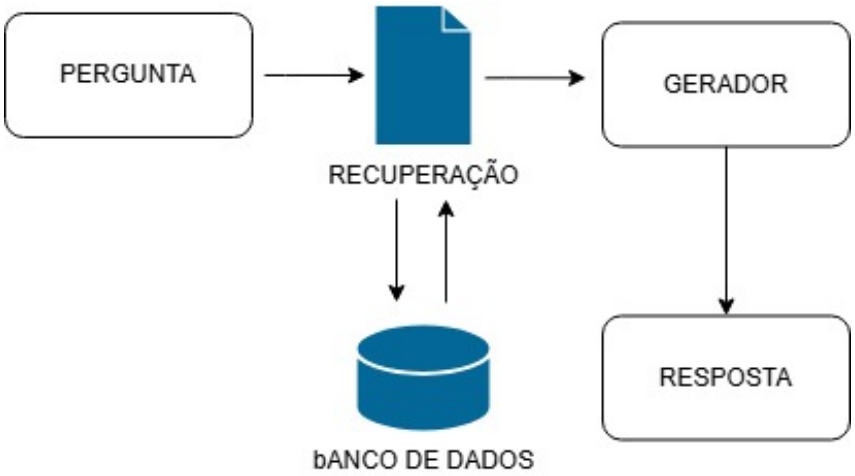
Os avanços recentes em QA, impulsionados por arquiteturas como o Transformer (VASWANI et al., 2017), têm permitido que os modelos capturem relações complexas entre pergunta e contexto, tornando as respostas mais coerentes e sensíveis ao significado. Entretanto, a principal limitação observada em tarefas de QA generativo reside na fidelidade factual das respostas, uma vez que os modelos podem apresentar informações incorretas, incompletas ou irrelevantes. Nesse cenário, o RAG surge como uma solução intermediária, ao combinar recuperação explícita de documentos com geração condicionada (LEWIS et al., 2020), permitindo respostas mais fundamentadas e auditáveis.

2.3 Retrieval-Augmented Generation

O modelo RAG, proposto por Lewis et al. (2020), constitui um marco no desenvolvimento de arquiteturas de geração de linguagem natural orientadas à recuperação de conhecimento. O principal objetivo do RAG é integrar, de forma diferenciada, a capacidade generativa de grandes modelos de linguagem com a busca ativa em bases de conhecimento externas, permitindo que o sistema produza respostas mais precisas, fundamentadas e atualizadas.

Segundo Lewis et al. (2020), as abordagens anteriores de geração de texto em tarefas intensivas em conhecimento dependiam exclusivamente da memória paramétrica do modelo, isto é, do conhecimento aprendido durante o treinamento, o que limitava sua capacidade de responder sobre informações recentes ou específicas. Para superar essa limitação, o RAG introduz a noção de uma memória não paramétrica, representada por um índice de documentos capaz de ser consultado em tempo de execução. Dessa forma, a geração de texto passa a ser condicionada tanto ao conhecimento interno do modelo quanto às informações recuperadas de forma dinâmica, como mostra a figura 2.

Figura 2 – Estrutura básica do RAG



Fonte: Produzido pelo autor.

A arquitetura do RAG é composta por dois módulos principais: o recuperador e o gerador. O recuperador é responsável por selecionar, a partir de uma base de conhecimento, os documentos mais relevantes à consulta de entrada, utilizando representações densas aprendidas por modelos como o DPR (*Dense Passage Retriever*). Já o gerador é um modelo de linguagem, que produz a resposta final com base no texto da consulta e nos documentos recuperados (LEWIS et al., 2020).

Em termos formais, o processo pode ser descrito como uma distribuição condicional, dado pela equação 2.

$$p(y|x) = \sum_{z \in \mathcal{D}} p(y|x,z)\, p(z|x) \tag{2}$$

Onde x representa a entrada (por exemplo, uma pergunta), z denota os documentos recuperados da base $\mathcal{D}$, e y é a resposta gerada. O modelo combina a probabilidade de um documento ser relevante, $p(z|x)$, com a probabilidade de gerar a resposta condicionada a esse documento, $p(y|x,z)$. Essa formulação torna o RAG um sistema diferencialmente treinável de ponta a ponta, em que o aprendizado otimiza tanto a recuperação quanto a geração de forma conjunta.

2.3.1 Vantagens e contribuições

A principal contribuição do RAG é permitir que modelos de linguagem acessem e utilizem conhecimento atualizado sem necessidade de re-treinamento completo. Além disso, o mecanismo de recuperação explícita melhora a transparência do processo, uma vez que as fontes consultadas podem ser rastreadas e auditadas. Os experimentos apresentados por Lewis et al. (2020) em tarefas como *Open-Domain Question Answering*, *Jeopardy* e *Natural Questions* demonstram que o RAG supera os resultados de modelos puramente paramétricos, como o BART, em métricas de precisão e cobertura de conhecimento.

Outra vantagem relevante é a modularidade da arquitetura, que permite atualizar o índice de documentos sem alterar os pesos do modelo generativo. Essa característica torna o RAG especialmente adequado para aplicações em domínios dinâmicos, como notícias, legislação, e bases científicas em constante expansão. Além disso, o sistema reduz significativamente o fenômeno de *alucinação*, já que o texto gerado tende a se basear em evidências explícitas extraídas do conjunto de documentos recuperados.

2.3.2 Desafios e limitações

Apesar dos avanços, o RAG ainda enfrenta uma série de desafios estruturais que motivam pesquisas contínuas na área. Um dos principais é a forte dependência da qualidade da recuperação: se o conjunto de documentos recuperados for impreciso, redundante ou irrelevante, o gerador poderá produzir respostas incorretas, vagas ou factualmente inconsistentes. Esse problema é amplificado em tarefas que exigem precisão contextual, como o *question answering*, nas quais a relevância do conteúdo recuperado é determinante para a coerência e a fidelidade factual da resposta final. Outro obstáculo importante é o custo computacional associado à indexação, manutenção e busca em bases extensas, o que limita a escalabilidade de sistemas RAG em ambientes de grande volume de dados (LEWIS et al., 2020).

Além disso, há o problema recorrente da redundância e da inconsistência entre documentos recuperados, que pode levar a contradições internas nas respostas geradas. Lewis et al. (2020) sugerem que abordagens híbridas, combinando recuperação densa e esparsa, podem mitigar parte dessas limitações, uma vez que cada técnica contribui com diferentes tipos de sinal semântico para o processo de recuperação. No entanto, o desafio da seleção e integração eficaz do conteúdo recuperado permanece aberto e é objeto de intensa investigação em estudos recentes.

Um dos avanços teóricos mais relevantes nessa direção é o trabalho de Liu et al. (2023), que analisa de forma sistemática o comportamento de LLMs em contextos longos. O estudo evidencia que, mesmo quando os modelos possuem capacidade técnica para processar grandes janelas de contexto, sua atenção efetiva não é distribuída de maneira uniforme ao longo do texto. Especificamente, informações posicionadas na porção intermediária do contexto tendem

a ser subutilizadas. Esse efeito implica que o aumento do tamanho do contexto não garante, por si só, uma melhora na precisão ou relevância das respostas — pelo contrário, contextos excessivamente extensos podem dispersar a atenção do modelo e dificultar a identificação das passagens realmente significativas para a tarefa. Assim, torna-se evidente que a qualidade da recuperação não depende apenas da quantidade de informação fornecida, mas também da forma como essa informação é organizada e posicionada dentro do espaço de contexto do modelo.

De modo complementar, o estudo de Zhong et al. (2024) introduz uma discussão fundamental sobre o impacto da granularidade na representação e recuperação do conhecimento em sistemas RAG. Os autores demonstram que a segmentação inadequada dos textos, seja em fragmentos muito longos, que diluem os conceitos principais, ou muito curtos, que perdem coesão semântica, compromete significativamente a eficiência da recuperação e a qualidade das respostas geradas. Para abordar esse problema, propõem o modelo *Mix-of-Granularity*, uma estratégia adaptativa que combina múltiplos níveis de granularidade na segmentação dos documentos, permitindo que o sistema selecione automaticamente a densidade informacional mais apropriada para cada consulta. Essa abordagem melhora a precisão da recuperação e reduz a redundância, ao mesmo tempo em que preserva o contexto semântico necessário para a geração de respostas coesas.

As conclusões de Liu et al. (2023) e Zhong et al. (2024) convergem para um ponto central: a eficiência dos sistemas RAG não depende unicamente da capacidade do modelo de linguagem, mas da arquitetura de organização do conhecimento subjacente à recuperação. Enquanto o primeiro estudo ressalta as limitações de atenção em janelas de contexto extensas, o segundo enfatiza a importância de ajustar o tamanho e a estrutura dos fragmentos de informação utilizados na indexação. Dessa forma, observa-se que o desempenho global do RAG está condicionado não apenas ao poder computacional ou à quantidade de dados, mas à forma como o conhecimento é fragmentado, representado e reintegrado durante a inferência.

Esses textos reforçam a relevância de investigações que explorem novas estratégias de representação do conhecimento, especialmente aquelas que conciliam eficiência computacional com coerência semântica. Nesse sentido, abordagens inspiradas em princípios cognitivos de aprendizagem, como o uso de *flashcards* digitais, emergem como alternativas promissoras para enfrentar o problema da granularidade e da dispersão de atenção, ao permitir uma recuperação mais direcionada, segmentada e cognitivamente significativa.

2.4 Fundamentos Cognitivos dos Flashcards

A metodologia de *flashcards* é uma das aplicações práticas mais conhecidas do princípio cognitivo da *retrieval practice*, amplamente estudado no campo da psicologia da aprendizagem e da memória. Segundo Roediger Henry L. e Butler (2011), a aprendizagem não ocorre apenas

durante os momentos de estudo passivo, mas é substancialmente reforçada durante o processo ativo de recuperação da informação, ou seja, quando o indivíduo tenta recordar conscientemente o conteúdo aprendido sem consultá-lo diretamente.

Tradicionalmente, considerava-se que os testes e revisões tinham como principal função avaliar o quanto foi aprendido. Contudo, as evidências experimentais apresentadas por Roediger Henry L. e Butler (2011) contradizem essa visão, demonstrando que o ato de recordar informações exerce um papel crucial na consolidação da memória de longo prazo. Esse fenômeno, descreve como o esforço cognitivo envolvido na recuperação fortalece as representações neurais do conteúdo, promovendo retenções significativamente maiores quando comparadas à simples releitura do material.

Os experimentos relatados pelos autores mostram que a prática de recuperação, mesmo sem *feedback* imediato, gera melhorias substanciais no desempenho de longo prazo, e que a combinação com o fornecimento posterior da resposta correta potencializa ainda mais o aprendizado. Essa dinâmica cognitiva baseia-se na ideia de que cada tentativa de recordar uma informação atua como um novo evento de aprendizado, reforçando as conexões sinápticas relacionadas ao conteúdo evocado.

Além disso, Roediger Henry L. e Butler (2011) apontam que a prática de recuperação favorece não apenas a memorização literal de informações, mas também o desenvolvimento de um conhecimento flexível e transferível, capaz de ser aplicado em diferentes contextos. Essa característica é particularmente relevante em ambientes educacionais e computacionais, pois permite que o conhecimento aprendido seja recuperado e utilizado em situações que demandam raciocínio, generalização e resolução de problemas.

Dentro desse contexto, o uso de *flashcards* digitais como ferramenta de aprendizagem ativa traduz experimentalmente os princípios de Roediger Henry L. e Butler (2011) em uma aplicação prática e sistematizada. Cada cartão funciona como uma unidade atômica de conhecimento, estimulando o aluno a recuperar uma informação e, em seguida, fornecer uma resposta, criando um ciclo contínuo de teste, feedback e reforço.

Recentemente, essa noção de organização em unidades atômicas tem encontrado paralelos no campo da inteligência artificial, especialmente no contexto de representação e recuperação de conhecimento. O trabalho de Raina e Gales (2024) propõe a estruturação de bases de conhecimento em unidades atômicas de conhecimento, segmentos mínimos semanticamente coesos, projetados para otimizar a recuperação e integração em sistemas RAG. Segundo os autores, dividir o conhecimento em unidades conceitualmente completas, mas compactas, reduz a redundância contextual e melhora a precisão da recuperação, permitindo que o modelo combine evidências de forma mais eficiente durante a geração. Esse conceito computacional reflete, de maneira análoga, o papel cognitivo dos *flashcards*, nos quais o conhecimento é organizado em blocos discretos que favorecem tanto a recordação ativa quanto a recombinação contextual.

A convergência entre Roediger Henry L. e Butler (2011) e a proposta de Raina e Gales (2024) sustenta a hipótese central deste trabalho: a organização do conhecimento em *unidades atômicas* pode servir como um elo entre cognição humana e aprendizado de máquina. Assim como os *flashcards* promovem consolidação e flexibilidade na memória humana, estruturas computacionais baseadas em fragmentos conceituais autossuficientes podem aprimorar a eficiência e a interpretabilidade de modelos de linguagem.

No contexto desta dissertação, o conceito de *retrieval practice* é, portanto, reinterpretado como uma estratégia cognitiva a ser incorporada a sistemas de RAG. A proposta consiste em estruturar a base de conhecimento do RAG em unidades semelhantes a *flashcards*, promovendo um processo de recuperação mais segmentado, ativo e direcionado, análogo ao modo como seres humanos consolidam memórias. Essa integração busca aproximar a arquitetura computacional de um modelo de aprendizado humano baseado em revisão e recordação ativa, potencializando a precisão, a coerência e a retenção contextual das respostas geradas.

3 Metodologia

Este capítulo descreve a metodologia adotada para o desenvolvimento da pesquisa, contemplando as etapas, os procedimentos e os recursos empregados para atingir os objetivos propostos. A metodologia tem como propósito estabelecer um caminho sistemático que permita a verificação empírica da hipótese central deste trabalho: a de que a organização do conhecimento em *flashcards* digitais pode otimizar o desempenho de sistemas de RAG em tarefas de QA.

A abordagem metodológica adotada é de natureza experimental e exploratória, com abordagem quantitativa e qualitativa. A pesquisa combina técnicas de engenharia de sistemas inteligentes, experimentação computacional e análise comparativa de resultados. Assim, busca-se tanto mensurar o impacto objetivo do uso de *flashcards* na precisão das respostas quanto interpretar os efeitos dessa estruturação sobre a capacidade de generalização e retenção contextual dos modelos testados.

Inicialmente, são apresentados as ferramentas e materiais utilizadas, incluindo o LLM empregado, as bases de dados, os procedimentos de preparação e geração de *flashcards*, e o processo de adaptação das questões para o formato de QA. Em seguida, descrevem-se as etapas de implementação do sistema RAG em duas configurações distintas, uma baseada em documentos longos e outra em *flashcards*, bem como as métricas e métodos de avaliação empregados para comparar o desempenho entre as abordagens.

3.1 Materiais e Preparação dos Dados

3.1.1 Seleção do modelo gerador

Para todas as tarefas de geração desta pesquisa, foi adotado o modelo Phi-4 (ABDIN et al., 2024). Este modelo integra a família Phi de *Small Language Models* (SLMs) da Microsoft, com 14 bilhões de parâmetros em uma arquitetura *decoder-only* baseada no Transformer.

A escolha do Phi-4 justifica-se por sua adequação às restrições práticas e aos objetivos empíricos deste estudo. Sendo um SLM, sua escala intencionalmente menor otimiza os parâmetros computacionais, como custo de inferência e viabilidade de implantação local, permitindo a execução robusta dos experimentos. Esta eficiência computacional é combinada com um desempenho competitivo que retém capacidades avançadas de compreensão e raciocínio, configurando o balanço ideal entre viabilidade e performance exigido por esta pesquisa.

3.1.2 Base de dados

Para a avaliação comparativa das duas abordagens de RAG (tradicional e *flashcards*), bem como do *baseline* (LLM-Puro), foi utilizado como base o conjunto de dados **BLUEX** (ALMEIDA et al., 2023). Este *dataset* é composto por questões objetivas de vestibulares brasileiros da Unicamp e USP, datadas entre 2018 e 2024, e inclui rótulos que identificam a matéria abordada em cada pergunta.

A estrutura original do BLUEX, composta por questões objetivas, foi adotada como a metodologia central de avaliação. Optou-se pela tarefa MCQA em vez de uma tarefa de geração aberta. Esta decisão se justifica pela natureza objetiva da avaliação, que permite uma mensuração quantitativa direta e inequívoca via acurácia, comparando a resposta do modelo com o gabarito oficial.

O BLUEX conta com questões datadas entre 2018 à 2024 dos vestibulares da Unicamp e USP, rotuladas com a matéria abordada na pergunta. Como apenas os domínios explorados nos materiais serão aproveitados, dois filtros foram aplicados ao conjunto de dados para alinhá-lo aos objetivos deste estudo: foram selecionadas apenas as questões rotuladas com os domínios de Biologia, Geografia e História, garantindo o alinhamento com as bases de conhecimento (cadernos da UNESP) descritas na Seção 3.1.3. Também foram removidas todas as questões que continham imagens, gráficos ou tabelas, visto que o *pipeline* de RAG implementado neste trabalho é estritamente textual.

Após a aplicação desses filtros, o conjunto de dados final utilizado para os experimentos totalizou **293 questões**.

O *dataset* BLUEX (ALMEIDA et al., 2023) é composto de diversos campos. Embora o *dataset* completo contenha uma rica gama de metadados, este trabalho foca nos campos relevantes para a extração e adaptação das questões.

Os principais campos da estrutura original são:

- `question`: O enunciado textual principal da questão.
- `id`: Um identificador único, composto pelo nome do vestibular, o ano e o número da questão.
- `alternative`: Um objeto ou lista contendo os textos das alternativas de múltipla escolha.
- `answer`: A chave, exemplo 'A', 'B', entre outros, que identifica a alternativa correta.

3.1.3 Escolha da disciplina

A escolha do domínio de conhecimento (a disciplina) é um componente crítico desta metodologia, pois o objetivo é avaliar a eficácia do mecanismo de recuperação e não a

capacidade de raciocínio intrínseco do LLM. Para isolar a variável "qualidade do contexto recuperado", foram estabelecidos os seguintes critérios para a seleção do material:

- Critério de exclusão: Domínios de Alta Subjetividade. Foram evitadas disciplinas cujas questões de vestibular dependem primariamente de interpretação aberta, como por exemplo, Língua Portuguesa. Nesses casos, o desempenho do LLM estaria mais atrelado ao seu conhecimento interno do que ao contexto factual fornecido pelo RAG. Isso dificultaria a mensuração do impacto real dos *flashcards*.
- Critério de exclusão: Domínios de Ciências Exatas. Também foram descartados domínios que exigem raciocínio simbólico e cálculos matemáticos complexos, como Matemática e Física. As LLMs, por sua natureza arquitetural, apresentam limitações conhecidas nessas tarefas. Portanto, a ocorrência de respostas errôneas seria primariamente atribuída às limitações da LLM, independentemente da qualidade do contexto recuperado, o que invalida a comparação.
- Critério de inclusão: Domínio densidade Factual e Conceitual. O domínio ideal deve ser densamente factual e conceitual. A disciplina escolhida precisaria ser rica em terminologias, definições, processos e correlações entre diferentes áreas, permitindo que o conhecimento fosse estruturado em *flashcards* de forma eficiente.

Atendendo a esses critérios, Geografia, Biologia e História foram escolhidos. Optou-se por essas três opções, já que se baseiam fortemente na memorização e correlação de um vasto conjunto de conceitos. A natureza predominantemente factual dessas disciplinas as tornam, portanto, ideal para a metodologia de *flashcards*, permitindo representar o conhecimento em unidades informativas, concisas e objetivas, alinhando-se perfeitamente com a hipótese desta dissertação.

3.1.4 Fonte de Conteúdo

O material de estudo utilizado como base de conhecimento foi a 2ª Edição dos Cadernos dos Cursinhos Pré-Universitários da Universidade Estadual Paulista "Júlio de Mesquita Filho" (Unesp), especificamente as matérias de biologia (CALDEIRA, 2016) e, geografia e história (ALMEIDA; MAGNONI, 2016), disponibilizado publicamente pela instituição.

Os cadernos demonstrado na figura 3 reúnem conteúdos programáticos típicos de vestibulares brasileiros e foram adotados por sua abrangência e alinhamento com a tarefa de QA educacional. O conteúdo textual foi extraído e normalizado (remoção de artefatos de formatação e padronização) para posterior indexação.

Figura 3 – Capa dos livros *Ciências da Natureza: Biologia*, volume 3. e *Ciências humanas: filosofia, geografia, história e sociologia*, volume 6.

Fonte: Unesp. Disponível em: <https://www2.unesp.br/portal#!/servico_ses/materiais-didaticos>. Acesso em: 19 out. 2025.

3.1.5 Estratégias de experimentos

A partir do mesmo conteúdo-fonte, foram criados dois índices vetoriais distintos para representar as duas abordagens de RAG e uma linha de base, onde apenas o LLM é utilizado, para futura comparação.

3.1.5.1 Grupo A: LLM puro

A abordagem comum foi utilizada como linha de base, dessa forma, o modelo de linguagem recebe a pergunta e responde a partir de sua memória paramétrica, como sugere a figura 4.

Para estabelecer a linha de base do desempenho e mensurar o conhecimento interno do modelo, foi definido este grupo de controle. Nesta configuração, o *pipeline* de RAG é desativado. O modelo de linguagem (LLM) recebe a consulta de entrada, enunciado e alternativas, e deve formular sua resposta utilizando exclusivamente seu conhecimento paramétrico, sem acesso a qualquer contexto factual recuperado de fontes externas. Esta abordagem, ilustrada na Figura 4, isola a capacidade intrínseca do LLM na tarefa de MCQA.

Figura 4 – Estrutura utilizada como *baseline*

Fonte: Elaborada pelo autor.

3.1.5.2 Grupo B: RAG Tradicional

Para fins de comparação, o texto integral do caderno foi processado por uma estratégia de segmentação textual, comumente referida como *chunking*.

A tecnologia empregada para este fim opera de forma hierárquica, tentando primeiro dividir o documento por separadores que indicam uma unidade semântica,como quebras de parágrafo, e, subsequentemente, por separadores mais granulares (como quebras de linha ou espaços), até que os fragmentos (*chunks*) resultantes atinjam um tamanho máximo pré-definido.

Esta abordagem é definida por dois conceitos principais:

- Tamanho do Fragmento (*Chunk Size*): Define o comprimento máximo de cada fragmento, medido em caracteres.
- Sobreposição (*Chunk Overlap*): Um mecanismo onde uma porção final de um fragmento é repetida no início do fragmento seguinte. O objetivo desta técnica é mitigar a perda de contexto e garantir a continuidade semântica que pode ocorrer nas bordas da segmentação.

3.1.5.3 Grupo C: RAG com Flashcards

Como alternativa estruturada, e representando a hipótese central deste trabalho, o conhecimento foi organizado em formato de *flashcards* (Figura 5).

O processo de geração sintética, seguiu um fluxo de duas etapas (geração e indexação), conforme ilustrado na Figura 6.

Como alternativa estruturada, o conhecimento foi organizado em formato de *flashcards*. O processo de criação consistiu em, primeiramente, isolar o conteúdo textual correspondente a cada tópico do sumário do caderno, por exemplo: Citologia, Genética. Em seguida, o modelo de linguagem Phi-4 foi instruído, a partir do prompt na Figura 7 a acessar este conteúdo e gerar um baralho com *flashcards* concisos, focados nos conceitos-chave daquele tópico. Por fim, cada *flashcard* gerado, contendo um par (pergunta, resposta), foi tratado como um documento de texto independente e indexado na *vector store*. O campo {format_instructions} no prompt instrui o LLM a formatar a saída como JSON, respeitando a estrutura da Figura 5.

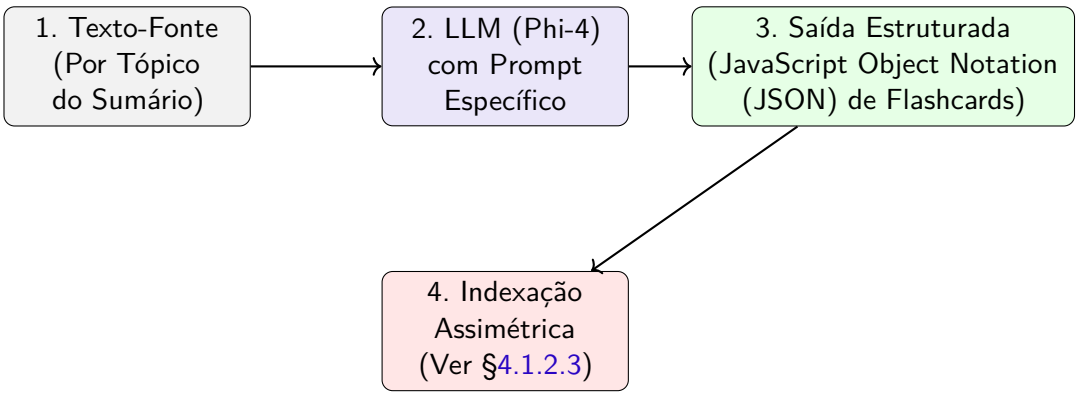
Figura 5 – Estrutura utilizada para os flashcards

```python
class Flashcard(BaseModel):
    id: int
    front: str = Field(description="A pergunta clara e objetiva de até 100 caracteres")
    back: str = Field(description="A resposta direta e objetiva para a pergunta")

    def __str__(self):
        return f"""
        Flashcard {self.id}:
        Question: {self.front}
        Answer: {self.back}
        ------------------------------
        """
```

Fonte: Elaborada pelo autor.

Figura 6 – Diagrama do pipeline de geração sintética de *flashcards*.



Fonte: Elaborada pelo autor.

3.2 Configuração da Arquitetura RAG

Para garantir uma comparação justa, ambos os grupos experimentais utilizaram os mesmos componentes de *embedding*, armazenamento e geração, alterando-se apenas a base de conhecimento.

3.2.1 Representação Semântica e Armazenamento Vetorial

Para que os fragmentos de texto (sejam *chunks* ou *flashcards*) possam ser recuperados por relevância, eles precisam ser convertidos para uma representação numérica que capture seu significado semântico.

Esta etapa é realizada por um modelo de *embeddings*. Este tipo de modelo é uma rede neural projetada para mapear unidades de texto, como sentenças ou parágrafos, para vetores

Figura 7 – Prompt de sistema e de usuário para a geração dos *flashcards*.

```
"""
Você é um criador de flashcards educacionais. Sua tarefa é gerar flash-
cards a partir do tema fornecido utilizando apenas o contexto dado.

---

{format_instructions}

Cada flashcard deve seguir o seguinte formato:

- Pergunta: clara, objetiva e de até 100 caracteres.
- Resposta: a resposta direta e objetiva para a pergunta.

Gere ao menos {n_de_cartões} flashcards sobre o seguinte tema: {tema}.
Certifique-se de que as perguntas cubram conceitos fundamentais, aplica
ções práticas e curiosidades importantes.
"""
```

Fonte: Código do autor.

de alta dimensão. A propriedade fundamental desses vetores é que textos com significados semanticamente próximos resultarão em vetores que estão geometricamente próximos no espaço vetorial.

Os vetores gerados para cada fragmento de texto são então indexados e persistidos em um banco de dados vetorial (*vector store*). O *vector store* se trata de um sistema de gerenciamento de dados otimizado para armazenar e consultar eficientemente grandes volumes de vetores, permitindo operações de busca por similaridade, para encontrar os fragmentos mais relevantes para uma determinada consulta.

3.2.2 Recuperação e Geração

O fluxo de consulta do sistema RAG é iniciado por uma consulta de entrada (*query*), no caso, a submissão de uma questão proveniente de um dos *datasets*. Esta pergunta é primeiro convertida em uma representação vetorial pelo mesmo modelo de *embeddings* descrito na seção anterior, garantindo que a consulta e os documentos estejam no mesmo espaço vetorial.

Em seguida, o componente de recuperação (*retriever*) executa a busca por similaridade semântica. Este processo é o núcleo do RAG e difere fundamentalmente de uma busca por palavra-chave. Em vez de procurar termos idênticos, o sistema compara matematicamente o vetor da pergunta com os vetores dos documentos armazenados no banco de dados. A similaridade é medida pela proximidade geométrica entre esses vetores através do cálculo de similaridade de cossenos, dado pela equação 3. Quanto mais próximos os vetores, maior a

relevância semântica entre a pergunta e o documento, mesmo que não compartilhem as mesmas palavras.

$$\text{similaridade}(A, B) = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} \tag{3}$$

Onde:

- A representa o vetor da consulta do usuário;
- B representa o vetor do documento armazenado;

O sistema, então, recupera um número pré-definido, k, dos documentos mais relevantes do índice correspondente, seja o índice de *chunks* ou o de *flashcards*. Estes k documentos recuperados são concatenados e formatados dentro de um *prompt*, que serve como instrução para o Modelo de Linguagem gerador.

Por fim, o LLM sintetiza a resposta final baseando-se exclusivamente no contexto fornecido por esses documentos. A comparação de desempenho entre as duas abordagens (tradicional vs. *flashcards*) incidirá sobre a assertividade das respostas geradas.

3.2.3 Avaliação

Para garantir uma avaliação objetiva e quantitativa do desempenho dos três grupos experimentais, a tarefa de *question answering* foi definida como MCQA.

Esta abordagem foi selecionada em detrimento de uma tarefa de QA aberto (geração de texto livre) por permitir o uso de uma métrica de sucesso inequívoca: a acurácia. Em vez de avaliar qualitativamente uma resposta gerada, o modelo de linguagem foi instruído a selecionar uma das alternativas fornecidas, permitindo a comparação direta e automática entre a resposta do modelo e o gabarito do *dataset* BLUEX.

Para cada uma das 293 questões do *dataset* BLUEX filtrado, a consulta enviada ao *pipeline* foi construída concatenando o enunciado da questão com suas respectivas alternativas.

3.2.3.1 Extração da resposta

Para permitir uma avaliação quantitativa e objetiva, o modelo de linguagem foi instruído a não gerar texto livre, mas sim a selecionar uma das alternativas fornecidas. A fim de forçar essa seleção, a saída do modelo foi restringida. O LLM foi instruído a retornar sua resposta em um formato estritamente estruturado, contendo apenas a identificação da alternativa que julgava correta (por exemplo, a letra 'a'), o que permitiu a comparação direta e automática com o gabarito.

3.2.3.2 Métricas de Avaliação

O desempenho foi medido por duas métricas principais:

1. Acurácia: Define a porcentagem de questões onde a alternativa escolhida pelo LLM coincidiu com O gabarito do *dataset* BLUEX.
2. Custo Computacional: Foi registrado o número médio de *tokens* (total, entrada e saída) por consulta para avaliar a eficiência de cada abordagem.

4 Experimentos e resultados

Este capítulo apresenta os experimentos realizados e os resultados obtidos no desenvolvimento e avaliação dos modelos propostos. Inicialmente, descrevem-se as condições experimentais, incluindo os conjuntos de dados, os parâmetros adotados e o ambiente de execução. Em seguida, são detalhados os procedimentos de teste, as métricas de desempenho utilizadas e os experimentos de comparação entre os diferentes arranjos RAG implementados. Por fim, são discutidos os resultados alcançados, suas implicações e as principais observações decorrentes da análise.

4.1 Configuração experimental

Nesta seção, são definidos os hiperparâmetros, ferramentas e o ambiente de execução utilizados para implementar e avaliar os modelos de RAG.

4.1.1 Ambiente de Execução e Ferramentas

A execução de modelos de linguagem é uma tarefa que demanda alto custo computacional, exigindo *hardware* especializado. Por esta razão, os experimentos foram conduzidos na plataforma Kaggle, cujo ambiente configurável oferece acesso a Unidades de Processamento Gráfico (GPUs), acelerando significativamente o tempo de inferência. Dentro deste ambiente, foi empregada a plataforma Ollama para gerenciar e servir localmente o modelo de linguagem. A orquestração de todo o *pipeline* de RAG, desde a indexação até a geração, foi implementada em Python, utilizando primariamente a biblioteca LangChain para definir a lógica de recuperação e para o *parsing* estruturado da saída do modelo. Por fim, o ChromaDB foi empregado como o *vector store* para a persistência e consulta por similaridade dos vetores de *embeddings*.

4.1.2 Parâmetros dos Grupos de Teste

Conforme os objetivos específicos, três configurações experimentais (Grupos A, B e C) foram estabelecidas para comparação, todas utilizando o mesmo modelo de linguagem e parâmetros de geração para garantir uma avaliação justa. Para avaliar os grupos de teste, as respostas são salvas em uma estrutura de tabela padrão (Tabela 1), que permite a uniformização dos dados.

- O modelo gerador usado foi o Phi4:14b;
- Como temperatura, parâmetro que controla a aleatoriedade da saída gerada pelo modelo de IA, 0 foi utilizado, de forma a garantir resultados determinísticos.

Tabela 1 – Dados gerado a partir das respostas da LLM

Campo	Descrição
`question_id`	Identificador da questão no BLUEX.
`LLM_response`	Resposta dada pela LLM geradora.
`LLM_alternative`	Alternativa escolhida pela LLM geradora.
`retrieved_docs`	Documentos recuperados ou vazio.
`response_metadata`	Metadados da resposta (modelo gerador, *timestamp*, etc.).
`usage_metadata`	Metadados de uso (quantidade de tokens na entrada, saída e total).

Fonte: Elaborado pelo autor.

Para garantir a padronização da tarefa de MCQA em todos os três grupos experimentais, a *query* foi inserida em um *template* de *prompt* padronizado. Este *prompt* tem o papel fundamental de instruir o modelo gerador sobre seu papel (ex: "vestibulando"), a tarefa e, crucialmente, o formato de saída estruturado (JSON) exigido.

Pequenas, mas metodologicamente significantes, variações no *prompt* de sistema foram utilizadas para diferenciar os grupos com e sem RAG, conforme detalhado no Figura 8, Onde o campo {query} diz respeito à pergunta de vestibular e {documento_relevantes} são os textos recuperados pelo *retrieval* para experimentação com RAG. Para os grupos de RAG (A e B), o modelo foi explicitamente instruído a basear-se **exclusivamente** no contexto recuperado, enquanto para o Grupo C (LLM-Puro), a instrução foi para que respondesse diretamente, sem menção a contexto externo.

4.1.2.1 Grupo A: LLM Puro (Sem RAG)

Nesta configuração, o LLM foi instruído a responder às questões sem receber qualquer contexto externo, medindo assim seu conhecimento interno.

4.1.2.2 Grupo B: RAG Tradicional (*chunks*)

Para esse grupo, o conteúdo textual extraído dos cadernos Caldeira (2016), Almeida e Magnoni (2016) foi processado através da estratégia de segmentação linear (*chunking*). Os hiperparâmetros desta configuração foram definidos empiricamente, conforme detalhado abaixo:

- Tamanho do Fragmento (`chunk_size`): Foi definido um tamanho máximo de **800 caracteres**. Este valor foi selecionado por representar um balanço entre manter unidades semanticamente coesas (como parágrafos curtos) e a granularidade da indexação.
- Sobreposição (`chunk_overlap`): Foi configurada uma sobreposição de **80 caracteres** entre fragmentos adjacentes. Este mecanismo é crucial para mitigar a perda de continuidade semântica que pode ocorrer nas bordas da segmentação.

Figura 8 – Prompt de sistema e de usuário para geração de respostas.

Prompt de Sistema (Grupo A)

```
"""
System:
Você é um estudante que está realizando o vestibular, assim, sempre que
lê uma pergunta responde em português e com alguma das alternativas se-
guindo as instruções

Human:
pergunta: {query}

{format_instructions}
"""
```

Prompt de Sistema (Grupos B e C)

```
"""
System:
Você é um estudante que está realizando o vestibular, assim, sempre que
lê uma pergunta responde em português e com alguma das alternativas se-
guindo as instruções

Human:
pergunta: {query}

Documentos: {documentos_relevantes}

{format_instructions}
"""
```

Fonte: Código do autor.

- Recuperação (*top-k*): Foi definido $k = 3$. Desta forma, o sistema recupera os três fragmentos (chunks) semanticamente mais relevantes para a consulta. A escolha de $k > 1$ visa fornecer ao LLM um contexto mais amplo e robusto, não se restringindo a um único fragmento de informação e permitindo a síntese de dados de múltiplas fontes.

4.1.2.3 Grupo C: RAG com Flashcards

No grupo de experimento, a base de conhecimento foi indexada seguindo a estrutura dos *flashcards*, onde apenas a pegunta do cartão é indexada e a resposta é guardada em um dicionário separado. Utilizando o Phi-4, cada tópico das matérias tiveram 5 flashcards específicos.

Para o grupo experimental, a base de conhecimento foi estruturada em *flashcards*, envolvendo duas etapas: geração e indexação assimétrica.

Os *flashcards* foram gerados sinteticamente para cada tópico do sumário dos cadernos Caldeira (2016), Almeida e Magnoni (2016), o modelo de linguagem `Phi-4` foi instruído a criar um *deck* de 5 *flashcards* concisos, contendo pares `pergunta` e `resposta` sobre os conceitos-chave daquele tópico.

A indexação dos cartões adotou uma abordagem assimétrica, projetada para otimizar a relevância da recuperação. O processo foi o seguinte:

- Para cada *flashcard*, apenas o texto da pergunta foi vetorizado e armazenado no índice do ChromaDB.
- O texto da resposta foi armazenado como metadado associado àquele vetor, mas não foi incluído no cálculo de similaridade semântica.

Esta decisão se justifica pela premissa de que a busca semântica é mais eficaz ao comparar a consulta do usuário, com outras perguntas , em vez de compará-la diretamente com respostas, isso traz maior velocidade durante a indexação e busca por similaridade.

Durante a inferência, o sistema recupera os $k = 3$ *flashcards* cujas *perguntas* são semanticamente mais similares à consulta de entrada. O *retriever* então retorna o conteúdo completo (a "resposta"armazenada nos metadados) desses *flashcards* para serem usados como contexto pelo LLM.

4.2 Resultados e Discussão

Nesta seção, os dados quantitativos coletados nos experimentos são apresentados e analisados.

4.2.1 Resultados Quantitativos (Acurácia)

A Tabela 2 apresenta a acurácia final obtida por cada um dos três grupos experimentais após a execução sobre o *dataset* de 293 questões, Onde p.p. significa pontos percentuais.

Tabela 2 – Comparativo de Acurácia no *dataset* BLUEX (MCQA).

Grupo Experimental	Acurácia (%)	Ganho vs. LLM-Puro (p.p.)
Grupo A (LLM-Puro)	$82,94\%$	N/A
Grupo B (RAG-tradicional)	$88,74\%$	$+5,80$ p.p.
Grupo C (RAG-Flashcards)	$84,30\%$	$+1,36$ p.p.

Fonte: Elaborado pelo autor.

4.2.1.1 Análise da Acurácia

Os resultados da Tabela 2 apresentam dois achados centrais. Primeiramente, a hipótese central de que os *flashcards* (Grupo C) superariam o RAG tradicional (Grupo B) não foi confirmada neste experimento. O Grupo B obteve a maior acurácia ($88,74\%$), superando o Grupo C ($84,30\%$).

Em segundo lugar, o impacto do RAG (em ambas as formas) na acurácia foi significante. O Grupo A (LLM-Puro) alcançou uma acurácia de $82,94\%$, indicando que o conhecimento paramétrico do Phi-4 já era suficiente para responder corretamente à vasta maioria das questões de vestibular (que estavam no formato de múltipla escolha). O RAG-tradicional (Grupo B) adicionou apenas $+5,80$ pontos percentuais, e o RAG-Flashcards (Grupo C) apenas $+1.36$ pontos percentuais.

4.2.2 Resultados Quantitativos (Custo Computacional)

A eficiência é crucial para a viabilidade do sistema. A Tabela 3 compara o custo médio de *tokens* por consulta para cada grupo.

Tabela 3 – Comparativo de Custo Computacional (Média de *Tokens* por Consulta).

Grupo Experimental	***Tokens* (Entrada)**	***Tokens* (Saída)**	**Total (Média)**
Grupo A (LLM-Puro)	226.075	90.828	$1.081,58$
Grupo B (RAG-tradicional)	763.218	103.331	$2.957,51$
Grupo C (RAG-Flashcards)	235.208	108.771	$1.173,99$

Fonte: Elaborado pelo autor.

A Tabela 3 demonstra que o RAG-Tradicional é drasticamente mais caro que as outras abordagens, com um custo médio $2,51\times$ maior que o RAG-Flashcards e $2,73\times$ maior que o LLM-Puro. Isso se deve à recuperação de *chunks* longos, que inflam o contexto de entrada do LLM.

Embora a média de *tokens* seja importante, ela não revela o custo-benefício. Uma métrica mais robusta é o custo por resposta correta, que analisa a eficiência do sistema em relação à sua acurácia. Este valor responde quantos *tokens* foram gastos, em média, para cada resposta correta que o sistema entregou. A Tabela 4 detalha essa métrica.

Tabela 4 – Análise de Custo-Benefício (*Tokens* por Resposta Correta).

Grupo Experimental	**Acurácia (%)**	**Custo por Resposta Correta**
Grupo A (LLM-Puro)	$82,94\%$	$1.304,13$ *tokens*
Grupo B (RAG-Tradicional)	$88,74\%$	$3.332,88$ *tokens*
Grupo C (RAG-Flashcards)	$84,30\%$	$1.392,63$ *tokens*

A Tabela 4 oferece a visão mais clara do experimento:

- O RAG-Tradicional é computacionalmente caro. Para cada acerto, ele custou $3.332,88$ *tokens*, um valor $2,73\times$ maior que o *baseline*.
- O RAG-Flashcards demonstrou um balanço superior. Ele entregou uma melhoria na acurácia sobre o Grupo A, acrescentando um custo baixo de *tokens*: seu custo por acerto ($1.392,63$) foi apenas $6,7\%$ maior que o do LLM-Puro, validando sua eficiência.

4.2.2.1 Análise de Custo-Benefício

Os dados da Tabela 3 reformulam a conclusão do experimento.

O Grupo B, embora tenha tido a maior acurácia, também teve o maior custo computacional, exigindo uma média de 2866 *tokens* por consulta. Isso se deve à recuperação de $k = 3$ *chunks* de 800 caracteres cada (totalizando ≈ 2.960 caracteres de contexto).

O Grupo C foi dramaticamente mais eficiente. Ele usou, em média, 1.174 *tokens* por consulta, o que representa apenas $39,69\%$ do custo do RAG-tradicional. Isso se deve à natureza atômica dos *flashcards* recuperados, que são semanticamente densos, mas textualmente curtos.

4.2.3 Discussão dos Experimentos

A análise cruzada de acurácia (Tabela 2) e custo computacional (Tabelas 3 e 4) permite uma conclusão matizada, que revela nuances sobre o valor de cada arquitetura RAG.

A hipótese principal, de que a estruturação em *flashcards* otimizaria a acurácia, não foi confirmada neste experimento. O Grupo B obteve a maior acurácia ($88,74\%$), superando o Grupo C ($84,30\%$). Notavelmente, o ganho de ambas as abordagens RAG sobre o Grupo A, que atingiu $82,94\%$, foi significativo, $+5,80$ p.p. e $+1,36$ p.p., respectivamente. Este resultado, por si só, é um achado relevante: sugere que o modelo Phi-4 possui um forte conhecimento paramétrico sobre os domínios de vestibulares dadas matérias utilizadas durante experimentação. A natureza da tarefa de MCQA, que fornece as alternativas ao modelo, provavelmente facilitou o acerto via conhecimento paramétrico, diminuindo a dependência de um contexto de recuperação preciso, diferentemente do que ocorreria em uma tarefa de QA aberta.

No entanto, a hipótese de otimização foi fortemente validada no eixo da eficiência computacional. O RAG com *flashcards* alcançou seu desempenho de $84,30\%$ utilizando uma média de 1.174 *tokens* por consulta. Em contraste, o RAG Tradicional exigiu uma média de 2.958 *tokens* para atingir seus $88,74\%$. Em termos práticos, o RAG-Flashcards utilizou apenas $41,6\%$ do custo computacional para entregar um resultado de acurácia quase idêntico (uma diferença de apenas $1,31$ p.p.).

Esta análise de custo-benefício torna-se ainda mais evidente ao se observar a métrica de *tokens* por resposta correta. Para cada resposta correta que entregou, o LLM-Puro custou 1.304 *tokens* e o RAG-Flashcards custou 1.393 *tokens*. Em contrapartida, o RAG-Tradicional

custou 3.333 *tokens* por acerto. Ou seja, o RAG Tradicional foi $2,73\times$) que o LLM-Puro para cada acerto, um custo alto. O RAG-Flashcards, por outro lado, foi mais eficiente, ao balancear qualidade e custo quando comparado ao LLM-Puro.

Conclui-se que, para a tarefa de MCQA em domínios de conhecimento geral, a arquitetura RAG-Flashcards apresenta um equilíbrio de custo-benefício muito superior. Embora o RAG Tradicional tenha vencido em acurácia absoluta, ele o fez através da "força bruta", injetando um contexto longo e caro ($k = 3$ *chunks* de 800 caracteres) que se mostrou computacionalmente ineficiente. O RAG-Flashcards, com sua abordagem "atômica"e indexação assimétrica, provou ser a arquitetura mais equilibrada, mitigando o alto custo de *tokens* do RAG tradicional sem sacrificar o desempenho. Assim, em sistemas onde o custo importa, a arquitetura a partir de flashcards tem grandes vantagens.

5 Conclusão

Este Trabalho de Conclusão de Curso investigou se a estruturação de conhecimento de um sistema RAG em formato de *flashcards*, abordagem inspirada na *retrieval practice* da ciência cognitiva (ROEDIGER HENRY L.; BUTLER, 2011), poderia otimizar o seu desempenho. A hipótese central era que esta organização atômica resolveria as limitações de granularidade e ruído contextual encontradas no RAG tradicional (LIU et al., 2023; ZHONG et al., 2024).

Para testar esta hipótese, foi realizado um experimento comparando três grupos: (A) LLM Puro (sem RAG), (B) RAG Tradicional (Chunks) e (C) RAG Experimental (Flashcards), em uma tarefa de MCQA sobre o *dataset* BLUEX.

Os resultados de acurácia (Tabela 2) não confirmaram a hipótese principal. Onde o RAG-Tradicional ($88,74\%$) obteve um desempenho ligeiramente superior ao RAG-Flashcards ($84,30\%$). Notavelmente, o ganho de ambas as abordagens RAG sobre o LLM-Puro ($82,94\%$) foi marginal. Este achado sugere que o modelo Phi-4 já possuía um vasto conhecimento paramétrico sobre o domínio de vestibulares, e a tarefa de MCQA não foi suficientemente desafiadora para exigir um contexto de recuperação superior.

Contudo, a principal contribuição do trabalho foi revelada na análise de eficiência (Tabela 3). O RAG-Flashcards alcançou seu desempenho, utilizando uma média de 1.174 *tokens* por consulta. O RAG-Tradicional, para atingir um resultado similar, exigiu 2.957 *tokens*.

Em suma, este trabalho contribuiu ao validar empiricamente que a organização do conhecimento é um pilar fundamental não apenas para a precisão, mas também para a eficiência de sistemas RAG. A abordagem de *flashcards*, inspirada na cognição humana, demonstrou ser um caminho promissor para a criação de sistemas de IA mais leves, rápidos e com um balanço superior de custo-benefício.

A conclusão desta monografia é, portanto, uma validação da hipótese sob a ótica da otimização de custo-benefício. A estruturação em *flashcards*, especialmente com a indexação assimétrica, provou ser uma arquitetura drasticamente mais eficiente, alcançando o mesmo nível de desempenho do RAG tradicional com apenas $39,69\%$ do custo computacional.

Esta descoberta é relevante para aplicações práticas, sugerindo que para domínios onde a eficiência e o custo de inferência são críticos, a organização de conhecimento em unidades atômicas é uma estratégia de engenharia de RAG superior à segmentação linear.

5.1 Limitações

Apesar dos resultados, este estudo apresenta limitações claras. A principal foi a escolha da tarefa de MCQA, que se mostrou insuficiente para penalizar o alto conhecimento paramétrico do Phi-4 e testar rigorosamente o impacto do contexto. Adicionalmente, o tamanho do *dataset* de avaliação, 293 questões, é considerado pequeno, o que limita a generalização estatística dos resultados; diferenças de acurácia de ≈2-5 p.p. podem não ser estatisticamente significantes em um conjunto tão restrito.

5.2 Trabalhos Futuros

Com base nestas conclusões e limitações, os trabalhos futuros devem se concentrar em testar a hipótese central em cenários onde ela é mais provável de ser validada:

1. Avaliação com QA Aberto: Repetir o experimento de forma que exija a geração de respostas abertas, sem o auxílio das alternativas. Este cenário forçaria os modelos a dependerem inteiramente da qualidade do contexto recuperado, testando de forma mais rigorosa o impacto da granularidade.

2. Domínios de Conhecimento Especializado: Aplicar as metodologias de RAG-tradicional e RAG-Flashcards em um domínio de conhecimento privado ou de nicho (ex: documentos jurídicos, manuais técnicos internos), onde o conhecimento paramétrico do LLM-Puro é muito menor.

3. Hibridização de Técnicas: Investigar se a técnica de indexação assimétrica, utilizada com sucesso nos *flashcards* 4.2.2, poderia ser aplicada também aos *chunks* tradicionais (ex: gerar uma "pergunta"para cada *chunk* e indexar apenas a pergunta), buscando unir a eficiência dos *flashcards* com a cobertura dos documentos longos.

Referências

ABDIN, M.; ANEJA, J.; BEHL, H.; BUBECK, S.; ELDAN, R.; GUNASEKAR, S.; HARRISON, M.; HEWETT, R. J.; JAVAHERIPI, M.; KAUFFMANN, P.; LEE, J. R.; LEE, Y. T.; LI, Y.; LIU, W.; MENDES, C. C. T.; NGUYEN, A.; PRICE, E.; ROSA, G. de; SAARIKIVI, O.; SALIM, A.; SHAH, S.; WANG, X.; WARD, R.; WU, Y.; YU, D.; ZHANG, C.; ZHANG, Y. *Phi-4 Technical Report*. 2024.

ALMEIDA, L. L. de; MAGNONI, M. d. G. M. *Ciências humanas : filosofia, geografia, história e sociologia (Cadernos dos Cursinhos Pré-Universitários da UNESP, Vol. 6)*. São Paulo, SP, Brasil: Universidade Estadual Paulista "Júlio de Mesquita Filho" – Pró-Reitoria de Extensão (PROEX), 2016. 2ª edição. Disponível em: <https://www2.unesp.br/Home/servico_ses/caderno_biologia.pdf>. Acesso em: 24 de ago. de 2025.

ALMEIDA, T. S.; LAITZ, T.; BONáS, G. K.; NOGUEIRA, R. *BLUEX: A benchmark based on Brazilian Leading Universities Entrance eXams*. 2023.

CALDEIRA, A. M. d. A. *Ciências da Natureza – Biologia (Cadernos dos Cursinhos Pré-Universitários da UNESP, Vol. 3)*. São Paulo, SP, Brasil: Universidade Estadual Paulista "Júlio de Mesquita Filho" – Pró-Reitoria de Extensão (PROEX), 2016. 2ª edição. Disponível em: <https://www2.unesp.br/Home/servico_ses/caderno_biologia.pdf>. Acesso em: 24 de ago. de 2025.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2018.

HENDRYCKS, D.; BURNS, C.; BASART, S.; ZOU, A.; MAZEIKA, M.; SONG, D.; STEINHARDT, J. *Measuring Massive Multitask Language Understanding*. 2020.

HIRSCHMAN, L.; GAIZAUSKAS, R. Natural language question answering: The view from here. *Natural Language Engineering*, v. 7, p. 275 – 300, 12 2001.

JIANG, A. Q.; SABLAYROLLES, A.; MENSCH, A.; BAMFORD, C.; CHAPLOT, D. S.; CASAS, D. de las; BRESSAND, F.; LENGYEL, G.; LAMPLE, G.; SAULNIER, L.; LAVAUD, L. R.; LACHAUX, M.-A.; STOCK, P.; SCAO, T. L.; LAVRIL, T.; WANG, T.; LACROIX, T.; SAYED, W. E. *Mistral 7B*. 2023.

LEWIS, P.; PEREZ, E.; PIKTUS, A.; PETRONI, F.; KARPUKHIN, V.; GOYAL, N.; KüTTLER, H.; LEWIS, M.; YIH, W.-t.; ROCKTäSCHEL, T.; RIEDEL, S.; KIELA, D. Retrieval-augmented generation for knowledge-intensive nlp tasks. In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2020. (NIPS '20). ISBN 9781713829546.

LIU, N. F.; LIN, K.; HEWITT, J.; PARANJAPE, A.; BEVILACQUA, M.; PETRONI, F.; LIANG, P. *Lost in the Middle: How Language Models Use Long Contexts*. 2023.

RADFORD, A.; NARASIMHAN, K. Improving language understanding by generative pre-training. In: . [S.l.: s.n.], 2018.

RAFFEL, C.; SHAZEER, N.; ROBERTS, A.; LEE, K.; NARANG, S.; MATENA, M.; ZHOU, Y.; LI, W.; LIU, P. J. *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*. 2019.

RAINA, V.; GALES, M. *Question-Based Retrieval using Atomic Units for Enterprise RAG*. 2024.

RAJPURKAR, P.; ZHANG, J.; LOPYREV, K.; LIANG, P. *SQuAD: 100,000+ Questions for Machine Comprehension of Text*. 2016.

ROEDIGER HENRY L., I.; BUTLER, A. C. The critical role of retrieval practice in long-term retention. *Trends in Cognitive Sciences*, Elsevier, v. 15, n. 1, p. 20–27, jan. 2011. ISSN 1364-6613.

TOUVRON, H.; LAVRIL, T.; IZACARD, G.; MARTINET, X.; LACHAUX, M.-A.; LACROIX, T.; ROZIèRE, B.; GOYAL, N.; HAMBRO, E.; AZHAR, F.; RODRIGUEZ, A.; JOULIN, A.; GRAVE, E.; LAMPLE, G. *LLaMA: Open and Efficient Foundation Language Models*. 2023.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L.; POLOSUKHIN, I. *Attention Is All You Need*. 2017.

WEI, J.; TAY, Y.; BOMMASANI, R.; RAFFEL, C.; ZOPH, B.; BORGEAUD, S.; YOGATAMA, D.; BOSMA, M.; ZHOU, D.; METZLER, D.; CHI, E. H.; HASHIMOTO, T.; VINYALS, O.; LIANG, P.; DEAN, J.; FEDUS, W. *Emergent Abilities of Large Language Models*. 2022.

ZHANG, Y.; LI, Y.; CUI, L.; CAI, D.; LIU, L.; FU, T.; HUANG, X.; ZHAO, E.; ZHANG, Y.; XU, C.; CHEN, Y.; WANG, L.; LUU, A. T.; BI, W.; SHI, F.; SHI, S. *Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models*. 2023.

ZHONG, Z.; LIU, H.; CUI, X.; ZHANG, X.; QIN, Z. *Mix-of-Granularity: Optimize the Chunking Granularity for Retrieval-Augmented Generation*. 2024.