Uma Arquitetura de RAG com Busca Semântica e Filtros Estruturados para Perguntas e Respostas no Domínio Jurídico

Matheus F. C. Brakes[1], **David O. C. Ferreira**[1], **Josiel P. C. Silva**[1], **Artur M. A. Novais**[1], **João P. C. Presa**[1], **Sávio S. T. de Oliveira**[1]

[1] Instituto de Informática – Universidade Federal de Goiás (UFG)
Goiânia – GO – Brasil

{artur.matos, oneil, josielpantaleao}@discente.ufg.br

{joaopaulop, brakes_fares}@egresso.ufg.br

savioteles@ufg.br

Resumo. *Sistemas de busca jurídica enfrentam desafios com dados heterogêneos e consultas complexas que mesclam texto e metadados. Apresentamos um framework de RAG híbrido composto por um pipeline de indexação otimizado e um orquestrador de recuperação. A indexação emprega saneamento textual, segmentação ancorada e enriquecimento com pré-contexto, enquanto a recuperação utiliza expansão multi-consulta e uma busca que combina filtros de metadados com busca vetorial, incluindo um fallback para garantir a cobertura. Nossos experimentos mostram que a otimização dos chunks reduziu o custo de tokens em 66,8% e, simultaneamente, aumentou o Recall@10 em 181,1%. Adicionalmente, a busca híbrida com filtros melhorou o MRR@10 em 27,5% e reduziu a latência em 24,1%, demonstrando ganhos sinérgicos em custo, acurácia e velocidade.*

Abstract. *Retrieval-Augmented Generation (RAG) systems in the legal domain must handle noisy, heterogeneous corpora and complex queries mixing semantic and structured criteria. We propose a hybrid RAG framework featuring (i) an optimized indexing pipeline with aggressive cleaning, section-anchored chunking, and contextual pre-pending to enhance semantic density, and (ii) a retrieval orchestrator that employs multi-query expansion and a hybrid search agent. This agent dynamically combines metadata filters with vector search and includes a safe fallback mechanism to prevent over-filtering. Our experiments show that the indexing optimizations reduced the average token count per chunk by 66.8% while increasing Recall@10 by 181.1%. Furthermore, the hybrid filtering layer improved MRR@10 by an additional 27.5% and reduced end-to-end latency by 24.1%. The proposed framework demonstrates synergistic improvements in accuracy, cost, and latency, presenting a robust solution for reliable legal Q&A systems.*

1. Introdução

O volume crescente de documentos jurídicos digitais, abrangendo legislação, jurisprudência e atos administrativos, impõe sobrecarga informacional a profissionais do direito,

servidores públicos e cidadãos. Navegar com eficiência por esse acervo para obter respostas precisas é desafiador devido à heterogeneidade de formatos, ruído de digitalizações antigas, metadados incompletos e variações redacionais acumuladas ao longo de décadas. Além disso, entendimentos evoluem com o tempo, gerando divergências legítimas entre períodos e ampliando a incerteza interpretativa no processo de busca e análise [Chalkidis et al. 2020, Manning et al. 2008].

Abordagens baseadas apenas em correspondência lexical sofrem com sinonímia, polissemia e jargão jurídico, frequentemente retornando resultados incompletos ou pouco pertinentes. Em consultas compostas por múltiplos elementos estruturados e textuais, pequenas variações de redação impactam desproporcionalmente o ranqueamento e a cobertura [Manning et al. 2008, Chalkidis et al. 2020]. Nesse contexto, *Retrieval-Augmented Generation* (RAG) combina recuperação e geração para responder a perguntas a partir de evidências, mas seu desempenho é limitado pela qualidade da etapa de recuperação e pela seleção eficaz de contexto [Lewis et al. 2020, Gao et al. 2024, Liu et al. 2024].

Este trabalho aborda a necessidade de ir além de "navegar" pelo acervo, permitindo que usuários formulem perguntas e obtenham respostas que resumem, expliquem e contextualizem decisões pretéritas. Propomos um framework de RAG com busca semântica-estruturada, que integra recuperação semântica vetorial, filtragem facetada por metadados e expansão multi-consulta, apoiado por um pipeline de tratamento textual e segmentação ancorada em seções. A ideia central é aumentar a cobertura sem sacrificar a precisão, reduzindo a latência percebida na obtenção de evidências úteis e controlando o custo de contexto.

As contribuições são: (i) uma arquitetura de indexação semântica enriquecida com metadados do domínio jurídico; (ii) uma orquestração de recuperação que combina multi-consulta e filtragem facetada de forma compatível com a intenção do usuário; e (iii) uma avaliação quantitativa que considera simultaneamente custo de tokens, tempo e métricas de recuperação (Recall@k e MRR@k), incluindo ablações para isolar o efeito de cada componente. A Seção 2 revisita fundamentos e trabalhos correlatos, a Seção 3 detalha o framework proposto, a Seção 4 apresenta resultados e discussão, e a Seção 5 traz conclusões e perspectivas futuras.

2. Fundamentos e Trabalhos Correlatos

Esta seção resume os fundamentos teóricos e trabalhos correlatos que sustentam o estudo. Abordamos (i) a arquitetura de *Retrieval-Augmented Generation* (RAG) e a busca semântica densa, (ii) estratégias avançadas de recuperação, como expansão de consultas e filtragem por metadados, e (iii) métricas e práticas de avaliação em domínios especializados, com ênfase no contexto jurídico.

2.1. Recuperação Aumentada por Geração (RAG) em domínios especializados

A arquitetura RAG combina um componente de recuperação com um modelo de linguagem gerativo, permitindo que respostas sejam ancoradas em evidências do corpus [Lewis et al. 2020]. A etapa de recuperação funciona como um funil de evidências; quando falha em trazer os trechos corretos, o gerador tende a produzir respostas imprecisas ou *alucinações* [Gao et al. 2024]. Essa dependência é ainda mais crítica em domínios especializados, como o jurídico, nos quais precisão terminológica e contextualização são

primordiais [Manning et al. 2008, Chalkidis et al. 2020]. Além disso, efeitos de posicionamento em janelas longas (*lost in the middle*) reforçam a necessidade de um *retriever* que priorize, no topo, as evidências mais relevantes [Liu et al. 2024].

2.2. Busca semântica densa e adaptação de domínio

A busca semântica via *embeddings* densos projeta textos em um espaço vetorial no qual proximidade reflete similaridade semântica. Em consultas verbosas, paráfrases ou quando há divergência de vocabulário entre usuário e corpus, métodos densos tendem a superar recuperadores puramente lexicais [Karpukhin et al. 2020, Reimers and Gurevych 2019]. Contudo, modelos genéricos podem ter desempenho subótimo em domínios especializados; técnicas de adaptação de domínio, como pré-treino contínuo e *fine-tuning* contrastivo, elevam a qualidade da recuperação [Gururangan et al. 2020, Chalkidis et al. 2020]. Avanços como interação tardia (*late interaction*) [Khattab and Zaharia 2020] e recuperadores não supervisionados robustos [Izacard et al. 2022] ampliam o estado da arte. Neste trabalho, empregamos modelos de embeddings densos de alta performance, adequados para o processamento de textos no contexto jurídico brasileiro.

2.3. Estratégias avançadas de recuperação: multi-consulta e filtragem estruturada

Para mitigar a lacuna entre a formulação do usuário e o registro da informação, adotamos *expansão por multi-consulta* (*multi-query rewriting*): a partir de uma pergunta inicial, o sistema gera variações autocontidas que preservam numerais e nomeações críticas (por exemplo, número do processo e ano) e as distribui entre elementos estruturados identificados. Evidências recentes mostram que reformular a pergunta em múltiplas variações aumenta a diversidade de candidatos e melhora o *recall* em RAG [Li et al. 2024]. Em paralelo, aplicamos filtragem por metadados (e.g., ANO, ÓRGÃO, RELATOR) para compor uma busca semântico-estruturada com navegação facetada e composição lógica coerente (conjunção quando os filtros descrevem o mesmo item; alternativa quando expressam opções), prática consolidada em *search UIs* [Hearst 2009]. Para contextualizar a abordagem de expansão de consulta empregada, remetemos também ao panorama clássico de *query expansion* [Carpineto and Romano 2012].

2.4. Avaliação de sistemas de recuperação e RAG

A avaliação deve contemplar acurácia da recuperação e custos computacionais. Métricas clássicas de Recuperação da Informação, como Recall@k e MRR@k, aferem cobertura e ordenação [Manning et al. 2008]. Em domínios específicos como o jurídico, *benchmarks* especializados são essenciais para validade externa [Chalkidis et al. 2020, Chalkidis et al. 2022]. Diante da escassez de recursos anotados para subdomínios, a construção de conjuntos sintéticos de alta qualidade, acompanhada de validação criteriosa, torna-se abordagem útil para medir desempenho em cenários realistas. Além de acurácia, é boa prática reportar latências (tempo de recuperação e tempo total) e consumo de tokens (prompt, contexto e resposta), possibilitando análise do *trade-off* entre desempenho e eficiência. Experimentos de ablação e intervalos de confiança ajudam a isolar o impacto de cada componente e a sustentar conclusões robustas.

3. Framework de RAG com busca híbrida

Para endereçar a heterogeneidade de fontes e a complexidade das consultas no domínio jurídico, desenvolvemos um framework de RAG com busca semântica-estruturada. A

abordagem se apoia em dois pilares: (i) um pipeline de indexação semântica enriquecida, que garante qualidade textual e padronização de metadados na origem, e (ii) um orquestrador de recuperação e síntese, que interpreta a intenção do usuário e executa recuperação densa combinada com filtros facetados de metadados. O primeiro pilar reduz ruído e consolida metadados do domínio; o segundo produz consultas robustas a variações de formulação e aplica filtros estruturais de forma inteligente.

3.1. Arquitetura da indexação

A arquitetura de indexação, detalhada na Figura 1, foi projetada para converter um acervo documental heterogêneo, composto por fontes nato-digitais e digitalizadas, em um índice vetorial semanticamente rico e estruturado. O processo inicia-se com um mecanismo de ingestão que aplica Reconhecimento Óptico de Caracteres (OCR) de forma condicional, atuando como *fallback* apenas para documentos baseados em imagem, a fim de otimizar a eficiência computacional.

O texto extraído passa por uma fase robusta de saneamento que combina a remoção de ruído estrutural via expressões regulares, como URLs, assinaturas eletrônicas e informações de identificação pessoal (PII), com a normalização de artefatos textuais, como espaçamentos irregulares, hifenizações indevidas e caracteres espúrios. Paralelamente, metadados essenciais do domínio (ano, órgão, tipo de documento, etc.) são extraídos, validados e padronizados em um esquema unificado, estabelecendo a base para a filtragem facetada precisa na etapa de recuperação [Hearst 2009].

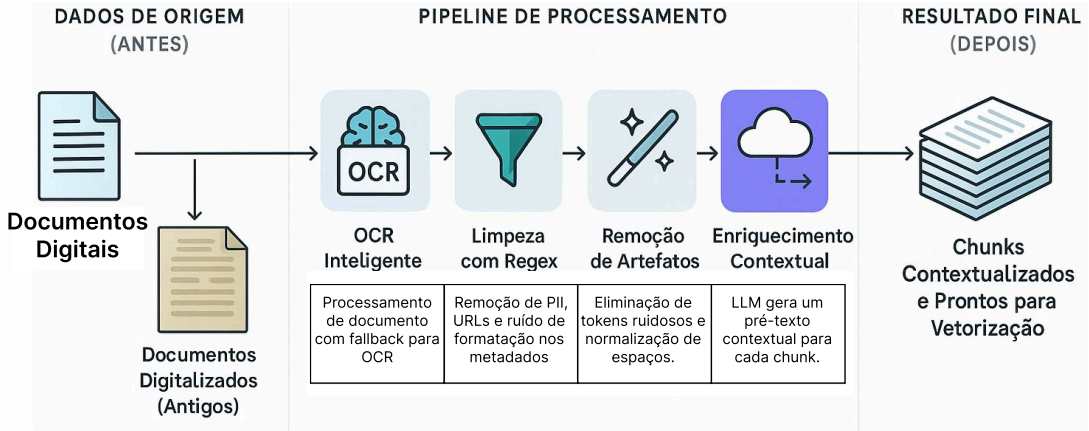

Figura 1. Pipeline: ingestão com OCR sob demanda, saneamento, segmentação ancorada, pré-contexto, vetorização e indexação com metadados.

Após o saneamento, o texto é submetido a um processo de segmentação estratégica. Em vez de uma divisão ingênua baseada em tamanho fixo, adotamos uma segmentação ancorada em seções estruturais do documento (e.g., ementa, dispositivo, capítulos), utilizando janelas de tamanho moderado com sobreposição para garantir a continuidade contextual [Manning et al. 2008]. O passo mais importante do enriquecimento ocorre em seguida: cada *chunk* gerado recebe um pré-texto programático, um cabeçalho curto que resume seu conteúdo e sua localização estrutural no documento de origem. Essa técnica é fundamental para mitigar o problema de perda de contexto em janelas longas [Liu et al. 2024], tornando cada *chunk* uma unidade de informação mais autossuficiente e semanticamente densa. Finalmente, os *chunks* enriquecidos são vetorizados por um

modelo de *embeddings* [Reimers and Gurevych 2019] e persistidos em um índice que associa cada vetor a seus respectivos metadados estruturados, prontos para a orquestração da busca híbrida.

3.2. Orquestração de recuperação e síntese (RAG)

O processo de orquestração em tempo de execução, ilustrado na Figura 2, foi projetado para decompor e resolver consultas complexas que mesclam intenções semânticas e restrições estruturadas. O fluxo descreve o ciclo completo desde a consulta do usuário até a entrega de uma resposta final ancorada em evidências.

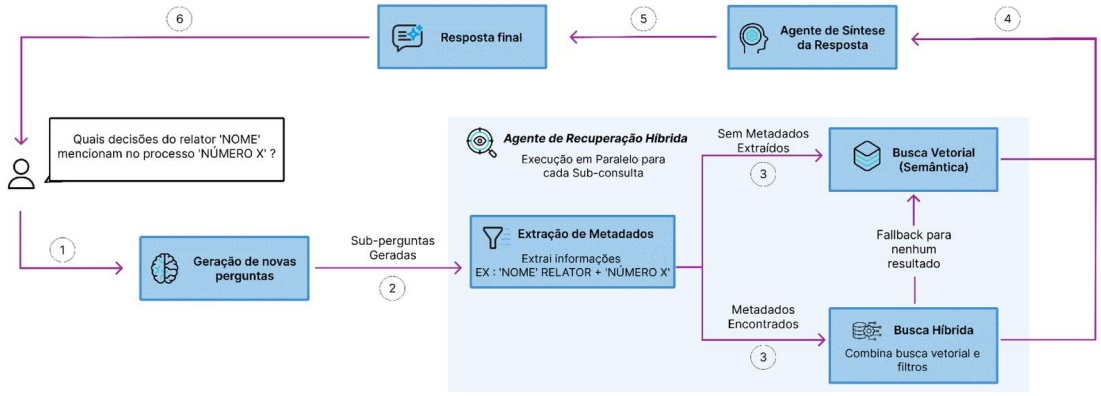

Figura 2. Fluxo RAG: multi-consulta paralela, extração/normalização de metadados, decisão AND/OR, busca híbrida com *fallback*, deduplicação/ordenação, seleção dinâmica de contexto e síntese com confidencialidade.

O ciclo inicia na **etapa 1**, quando o usuário envia uma pergunta ao sistema (por exemplo: "Quais decisões do relator NOME mencionam no processo NÚMERO X?"). Em resposta, o sistema executa a **etapa 2** (Geração de novas perguntas), em que a consulta original é expandida para um conjunto de subconsultas autocontidas. Essa técnica de multi-consulta visa aumentar o *recall* e a robustez, cobrindo reformulações plausíveis da intenção do usuário [Carpineto and Romano 2012].

A estratégia de busca é, então, selecionada dinamicamente com base no resultado dessa extração na **etapa 3**. Caso metadados sejam identificados, o sistema aciona a Busca Híbrida, que combina a busca vetorial com filtros de metadados pré-indexados. Esta abordagem visa aumentar a precisão ao restringir o espaço de busca apenas a documentos que satisfazem as restrições estruturais. Para mitigar o risco de *over-filtering* (onde filtros muito restritivos não retornam resultados), implementamos o seguinte mecanismo: se a busca híbrida falhar, o sistema recorre automaticamente à busca vetorial, garantindo a cobertura. Na ausência de entidades estruturadas na consulta, o sistema já utiliza diretamente a busca vetorial, otimizando a recuperação para questões abertas e puramente semânticas.

Os *chunks* recuperados por todas as subconsultas são então consolidados na **etapa 4**, quando são deduplicados e ordenados. As evidências mais relevantes são enviadas ao *Agente de Síntese da Resposta* na **etapa 5**. Esse agente, um LLM, gera uma resposta coesa e estritamente ancorada nos trechos recuperados. Por fim, na **etapa 6**, a resposta final é

entregue ao usuário. O conteúdo gerado inclui citações para garantir a rastreabilidade e respeita as políticas de confidencialidade aplicáveis.

4. Resultados

O saneamento textual com normalização e remoção de artefatos reduziu de forma substancial o custo de tokens por *chunk* e elevou a qualidade da recuperação no *top*-k. Em paralelo, a orquestração com filtros de metadados aumentou a precisão e reduziu a latência fim-a-fim, ao restringir o espaço de busca sem perda de cobertura graças ao *fallback* vetorial.

4.1. Ambiente experimental

Os experimentos utilizam o acervo institucional após saneamento de OCR, exclusão de cabeçalhos e rodapés repetitivos, assinaturas e URLs, com segmentação em *chunks* ancorados por seção e inclusão de um pré-contexto breve em todos os *chunks* antes da vetorização. A recuperação emprega multi-consulta adaptativa combinada a filtros de metadados com decisão automática entre conjunção ou alternativa, além de *fallback* para busca vetorial quando necessário. Documentos marcados como sigilosos são excluídos em buscas abertas ou mascarados quando retornam por filtros.

Houve também mudança de política de segmentação: anteriormente, cada documento era mantido em um único *chunk* e só era dividido quando excedia o limite do modelo de *embeddings*. Atualmente, documentos com até 680 tokens são indexados inteiros; acima desse limiar, são divididos em *chunks* menores e cada *chunk* recebe um breve resumo de contexto. Essa política explica dois efeitos complementares: (i) forte redução do custo por *chunk* na indexação e na inferência; (ii) estabilidade do total por documento quando se somam todos os *chunks*, pois o pré-contexto, mesmo curto, é adicionado a cada fragmento. Em termos de custo por pergunta, o barateamento por *chunk* domina, pois menos texto irrelevante é processado a cada recuperação.

4.2. Resultados da busca híbrida: quantidade de tokens, tempo, acurácia

A Tabela 1 revela o efeito da nova política de segmentação no nível do *chunk*. Observa-se uma redução consistente em todas as métricas. O total de *tokens* do acervo, quando agregado por *chunk*, diminuiu 66,78%, passando de aproximadamente 325 milhões para 108 milhões. Esse resultado é refletido diretamente na média de *tokens* por *chunk*, que caiu de 1.403 para 466. A redução na mediana foi menos acentuada, porém ainda importante $(-39,67\%)$, indicando que a política anterior gerava uma distribuição de tamanhos mais assimétrica, com muitos documentos longos mantidos como um único *chunk* massivo, inflando a média. A nova abordagem, ao impor limites de segmentação mais estritos, produz *chunks* mais uniformes e significativamente mais curtos, o que impacta diretamente o custo de vetorização (indexação) e, mais importante, o custo de contexto enviado ao LLM a cada consulta.

Em contrapartida, a Tabela 2 serve como um controle para garantir que a redução de custo por *chunk* não se deu pela simples eliminação de conteúdo. Ao agregar a contagem de *tokens* por documento (somando todos os seus *chunks* constituintes), o total permanece praticamente estável, com um aumento de apenas 0,3%. Este leve acréscimo é explicado pela adição do pré-texto contextual a cada um dos fragmentos gerados

Tabela 1. Orçamento de tokens por chunk (antes vs. depois).

Métrica	Antes	Depois	Δ abs.	Δ rel.
Total de tokens do acervo (por chunk)	325.517.100	108.126.934	$-217.390.166$	$-66{,}78\%$
Média de tokens por chunk	1.403,71	466,27	$-937{,}44$	$-66{,}78\%$
Mediana de tokens por chunk	731,00	441,00	$-290{,}00$	$-39{,}67\%$

a partir de um mesmo documento original. Portanto, um documento que antes era um *chunk* único e agora é dividido em três *chunks* receberá o cabeçalho contextual três vezes. Esse aumento pequeno no armazenamento total em troca de uma redução massiva no custo operacional por consulta é vantajoso para as aplicações. As tabelas, em conjunto, demonstram que a nova arquitetura de indexação otimiza a unidade de recuperação sem sacrificar a integridade informacional do acervo.

Tabela 2. Orçamento de tokens por documento (documentos comuns, soma dos chunks).

Métrica	Antes	Depois	Δ abs.	Δ rel.
Total de tokens (docs comuns)	323.291.627	324.342.400	$+1.050.756$	$+0{,}3\%$
Média de tokens por documento	1.468,90	1.473,67	$+4{,}77$	$+0{,}3\%$
Mediana por documento	777,00	869,00	$+92{,}00$	$+11{,}8\%$

As Tabelas 1 e 2 mostram que a combinação de limpeza e pré-contexto produz *chunks* mais curtos e informativos ($-66{,}78\%$ em média por *chunk*). Ao somar *chunks* por documento, o total permanece praticamente estável devido ao pré-contexto em todos os fragmentos e ao leve aumento do número de *chunks* por documento. Ainda assim, o ganho por *chunk* é o que impacta diretamente o custo por consulta, pois reduz o texto irrelevante enviado ao *retriever* e à LLM.

A Tabela 3 evidencia que a otimização da indexação se traduz em ganhos expressivos de acurácia: o Recall@10 aumentou de 0,1099 para 0,3089 (+181,1%) e o MRR@10 de 0,0727 para 0,1914 (+163,3%). Esses ganhos decorrem da combinação entre saneamento (remoção de ruído de OCR, cabeçalhos etc.), adição de pré-contexto e segmentação mais focada, que geram vetores mais fiéis ao conteúdo relevante e reduzem a diluição de evidência em *chunks* muito longos.

Tabela 3. Acurácia (k = 10) com impacto isolado do saneamento de *chunks*.

Métrica	Antes	Depois	Δ abs.	Δ rel.
MRR@10	0,0727	0,1914	$+0{,}1187$	$+163{,}3\%$
Recall@10	0,1099	0,3089	$+0{,}1990$	$+181{,}1\%$

A Tabela 4 apresenta uma ablação ao comparar o fluxo completo do *framework* deste trabalho, que utiliza um agente de busca híbrida com filtros, contra uma linha de base que emprega apenas a busca vetorial. Os resultados demonstram que a incorporação

de filtros de metadados gera ganhos simultâneos tanto em acurácia quanto em eficiência. No que tange à qualidade da recuperação, o **Recall@10** aumentou 15,6%, indicando que a busca híbrida é mais eficaz em localizar documentos relevantes. O ganho é ainda mais pronunciado no **MRR@10**, que cresceu 27,5%. Este resultado é particularmente importante, pois demonstra que o agente não apenas encontra os documentos corretos com mais frequência, mas também os posiciona mais perto do topo da lista de resultados, um fator determinante para a qualidade da resposta final gerada pelo LLM.

Tabela 4. Fluxo com filtros vs. sem filtros (k = 10).

Métrica	**Sem filtros (vetorial)**	**Com filtros (agente)**	**Δ abs.**	**Δ rel.**
Recall@10	0,449	0,519	$+0{,}070$	$+15{,}6\%$
MRR@10	0,363	0,463	$+0{,}100$	$+27{,}5\%$
Tempo fim-a-fim médio (s)	34,62	26,27	$-8{,}35$	$-24{,}1\%$
Tempo fim-a-fim p50 (s)	30,56	22,81	$-7{,}74$	$-25{,}3\%$
Tempo fim-a-fim p95 (s)	65,04	48,28	$-16{,}76$	$-25{,}8\%$
Tempo servidor médio (s)	34,42	22,29	$-12{,}13$	$-35{,}2\%$

Além dos ganhos em acurácia, a Tabela 4 revela uma redução na latência do sistema. O tempo de resposta fim-a-fim médio diminuiu 24,1%, de 34,62 para 26,27 segundos. Essa melhoria é consistente em toda a distribuição, com reduções de 25,3% na mediana (p50) e 25,8% no 95º percentil (p95), o que indica que o sistema se torna não apenas mais rápido, mas também mais previsível, com as consultas mais lentas sendo significativamente aceleradas. A eficiência do mecanismo de filtragem é confirmada pela redução de 35,2% no tempo médio de servidor. Isso ocorre porque a aplicação de filtros (*pre-filtering*) restringe o espaço de busca de forma eficiente, diminuindo a quantidade de candidatos que precisam ser processados e ranqueados pela camada semântica, que é computacionalmente mais custosa.

4.3. Discussão dos resultados

Os resultados experimentais demonstram, de forma complementar, a eficácia da nossa abordagem em duas frentes: a otimização da unidade de recuperação (o *chunk*) e a orquestração inteligente da busca. Primeiramente, as Tabelas 1 a 3 revelam um cenário de ganho duplo na etapa de indexação. Conseguimos reduzir o custo operacional por *chunk*, ao mesmo tempo em que aumentamos a acurácia da recuperação base.

Sobre essa base, a busca híbrida não apenas melhora a acurácia como reduz a latência em cerca de 25%. Consultas com identificadores explícitos, como números de acórdão, ilustram esse efeito: a busca puramente vetorial tende a recuperar trechos que apenas mencionam o identificador, enquanto o agente híbrido restringe o universo a documentos que satisfazem filtros estruturados (tipo, número, ano) antes de aplicar a similaridade semântica.

No conjunto, os experimentos indicam que *chunks* mais densos e coerentes, combinados com filtragem estruturada e *fallback* vetorial, produzem um *framework* que equilibra de forma eficaz acurácia, custo e velocidade. Todos os experimentos foram executados em uma máquina DGX institucional equipada com GPU NVIDIA A100, que hospeda

a API do Tribunal de Contas. Nesse ambiente, a carga local concentra-se na geração de *embeddings* e na busca vetorial, com baixa utilização efetiva de GPU; assim, os ganhos de latência refletem principalmente o desenho da arquitetura, e não mudanças de capacidade de hardware.

5. Conclusão e Trabalhos futuros

Este trabalho apresentou um framework de RAG com busca híbrida no domínio jurídico, combinando saneamento agressivo de texto, segmentação em *chunks* com pré-contexto e uma camada de recuperação que integra filtros de metadados e expansão multi-consulta. A avaliação quantitativa mostrou reduções expressivas no custo de contexto, melhorias consistentes em Recall@10 e MRR@10 e diminuição da latência fim-a-fim, indicando bom equilíbrio entre acurácia, custo e velocidade.

Consultas por número de acórdão ilustram a importância dos filtros. Sem eles, o *retriever* denso é sensível a meras menções do identificador e pode priorizando trechos pouco relevantes. Na busca híbrida, a extração de metadados e a composição lógica filtram por tipo, número e ano, concentrando a busca no item correto e reduzindo falsos positivos, mantendo cobertura pelo *fallback* vetorial. Assim, Saneamento de *chunks* e filtragem por metadados funcionam, complementando para a confiabilidade da recuperação.

As principais limitações observadas decorrem da heterogeneidade do acervo, da qualidade variável dos metadados, da dependência de heurísticas automáticas de relevância e da variação de latência pela complexidade das consultas. Contudo, as evidências indicam que saneamento de *chunks* e filtragem por metadados são alavancas complementares para reduzir custo e aumentar a confiabilidade das respostas.

Apesar das limitações, o framework é reutilizável em domínios afins como saúde, legislativo, auditoria, educação, entre outros. Tais contextos compartilham documentos longos, heterogeneidade, vocabulário técnico e dependência de metadados, exigindo adaptação mínima do framework, como na taxonomia de metadados, o saneamento às especificidades do domínio e as regras de ancoragem seccional.

Em relação a trabalhos futuros, várias direções relevantes podem enriquecer o sistema. Pretende-se ampliar a extração de conhecimento estruturado, incluindo rotulagem temática automática com base em taxonomias jurídicas controladas, melhoria da resolução de entidades nomeadas (como relator ou órgão) e extração de elementos críticos como resultado dispositório, fase processual e status decisório (revogação, reforma). O desenvolvimento de um grafo de citações também faz parte da agenda, permitindo mapear relações entre documentos e enriquecer consultas complexas.

Outra linha essencial envolve avaliar não apenas a recuperação, mas também a qualidade final das respostas geradas pelo LLM.Para isso, pretendemos adotar métodos de avaliação qualitativa com *LLM-as-a-judge*, usando métricas como *faithfulness*, completude, precisão jurídica e clareza textual, complementadas por feedback do usuário.

Adicionalmente, planeja-se incorporar *benchmarks* públicos (LexGLUE, Legal-BERT, Contriever-Legal) para permitir comparações externas e avaliar a transferibilidade do framework além do acervo institucional.

Finalmente, melhorias adicionais em detecção automática de sigilo, enriquecimento semântico e ancoragem estrutural mais precisa devem ampliar a capacidade de fil-

tragem, facilitar a desambiguação e habilitar novas consultas, consolidando uma solução RAG escalável, auditável e adequada a domínios especializados.

Referências

Carpineto, C. and Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Computing Surveys*, 44(1).

Chalkidis, I., Jana, A., Hartung, D., Bommarito, M., Androutsopoulos, I., Katz, D. M., and Aletras, N. (2022). LexGLUE: A benchmark dataset for legal language understanding in english. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Chalkidis, I., Kamateri, E., Lazaridou, K., Aletras, N., Katakalou, M., and Krithara, A. (2020). LEGAL-BERT: The Muppets straight out of Law School. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., and Wang, H. (2024). Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., and Smith, N. A. (2020). Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of ACL*.

Hearst, M. A. (2009). *Search User Interfaces*. Cambridge University Press.

Izacard, G., Caron, M., Hosseini, L., Riedel, S., Bojanowski, P., Joulin, A., and Grave, E. (2022). Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research (TMLR)*.

Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W. (2020). Dense passage retrieval for open-domain question answering. In *Proceedings of EMNLP*.

Khattab, O. and Zaharia, M. (2020). ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of SIGIR*.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Ott, M., tau Chen, W., Conneau, A., and others (2020). Retrieval-augmented generation for knowledge-intensive NLP. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Li, Z., Wang, J., Jiang, Z., Mao, H., Chen, Z., Du, J., Zhang, Y., Zhang, F., Zhang, D., and Liu, Y. (2024). Dmqr-rag: Diverse multi-query rewriting for retrieval-augmented generation. *arXiv preprint arXiv:2411.13154*.

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., and Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proceedings of EMNLP-IJCNLP*.