

PalanqueIA: A RAG Chatbot for Government Plans in the Brazilian Electoral Landscape

Abstract

Context: According to the Brazilian electoral code, candidates for Executive positions are required to present their government plans. This plan outlines proposed actions and policies, though implementation is discretionary. Proposed legislation (Bill 2082/21) requires the electoral courts to provide online access to these plans, aiming to counter electoral demagoguery. **Problem:** In Brazilian politics, electoral campaigns often lack substantive proposals for regional development, prioritizing self-promotion and personal attacks over meaningful debate. Candidates frequently make unrealistic promises beyond legal or budgetary limits, fostering disinformation and public mistrust. **Proposed Solution:** The proposed solution entails the development of a conversational system. PalanqueIA is an advanced virtual assistant designed to support candidates, analysts, and policymakers in the formulation of public policies. Utilizing artificial intelligence and Retrieval-Augmented Generation, it facilitates users' access to, exploration of, and generation of innovative policy proposals based on comprehensive data and contextual insights. **Information Systems Theory:** The work is grounded in the theory of information retrieval in Information Science and Computer Science. **Method:** The methodology employed was a Proof of Concept. The vector database was constructed from official documents, and user interaction with the chatbot was monitored to evaluate the accuracy of the responses. **Summary of Results:** The findings demonstrated that the chatbot is capable of executing complex queries efficiently, providing relevant and contextualized answers. The combination of a large language model and the vector base proved effective in retrieving information. **Contributions and Impact on the Information Systems Area:** This research contributes to the Information Systems field by demonstrating how Information Retrieval techniques can be enhanced by advanced language models in specific search contexts.

Keywords

Government plans, chatbot, conversational AI, retrieval-augmented generation, elections

1 Introdução

Incluído pela Lei nº 12.034 de 2009 [4], a qual estabeleceu novas regras para o processo eleitoral, o pedido de registro de candidatura dos cargos a Prefeito, Governador do Estado e Presidente da República, deve vir acompanhada, obrigatoriamente, do plano de governo do candidato.

O plano de governo é um documento no qual candidatos a cargos do Executivo apresentam à sociedade suas propostas, planejamentos e ideias de ações e intenções que pretendem implementar caso sejam eleitos durante o mandato. As propostas não são juridicamente vinculativas, permitindo ao agente político eleito discricionariedade quanto à sua implementação.

A divulgação de conteúdos é de particular importância. O Projeto de Lei 2082/21, atualmente em tramitação na Câmara dos

Deputados[6], prevê que o Tribunal Superior Eleitoral (TSE) e os Tribunais Regionais Eleitorais indiquem aos eleitores, em suas páginas oficiais na internet, formas de acesso ao conteúdo do conteúdo supracitado. Segundo o autor do Projeto, a proposta política serve como antídoto à demagogia eleitoral.

1.1 Problemática

No cenário político brasileiro, as campanhas eleitorais frequentemente são marcadas pela superficialidade e pela falta de compromisso com propostas sólidas visando ao desenvolvimento regional. Em vez de focar em ideias e iniciativas concretas, os debates políticos geralmente priorizam a autopromoção e, não raramente, a ataques pessoais e retórica vazia, desviando a atenção das questões fundamentais que impactam o bem-estar social. [1]

Outro aspecto considerável são as esdrúxulas promessas feitas em campanha, frequentemente desvinculada de um plano de execução factível. Ao propor medidas que vão além das competências legais ou sem consideração pelas restrições orçamentárias e legislativas, candidatos não apenas desviam o foco das reais necessidades do público, mas também promovem um ambiente de desinformação, minando o debate político qualificado. Tais promessas, ao não serem cumpridas, contribuem para o descrédito das instituições políticas e perpetuam um ciclo de frustração e desconfiança na sociedade, comprometendo a responsabilização e a eficácia do processo eleitoral. [7]

Esse quadro afeta diretamente a qualidade da informação disponível para os eleitores, fazendo com que a sociedade desconheça e não compreenda as ideias políticas dos candidatos, e desmotivando um debate público mais substancial. Como resultado, o cenário eleitoral acaba favorecendo práticas como compra de votos e difusão de desinformação, ao invés de gerar discussões informadas sobre o futuro social.

1.2 Solução Proosta

Este trabalho apresenta o PalanqueIA, um assistente virtual inteligente projetado para auxiliar os candidatos, analistas, gestores de políticas públicas e outros profissionais, no desenvolvimento de políticas públicas. Construído sob o formato de um ChatBot com base em inteligência artificial (IA) e técnicas de Recuperação Aumentada de Dados (RAG), o PalanqueIA é especialista em recuperação e criação de propostas políticas.

A ferramenta integra uma ampla base de dados com milhares de propostas políticas encaminhadas pelos candidatos na ocasião de sua candidatura, associada a um grande modelo de linguagem (LLM), possibilitando aos usuários consultar propostas existentes, explorar alternativas ou obter sugestões para novas políticas. A arquitetura RAG emprega técnicas avançadas de recuperação e geração de informações, permitindo que o chatbot responda a consultas com precisão e contextualização, oferecendo *insights* baseados em políticas já formuladas e em sugestões geradas pela inteligência artificial.

1.3 Objetivos

O objetivo principal deste estudo é realizar uma prova de conceito de um sistema inteligente de recuperação e proposição de planos de governo utilizando técnicas RAG e LLMs. O PalanqueIA pretende oferecer um ambiente acessível e interativo, onde candidatos, eleitores e formuladores de políticas públicas possam interagir de maneira qualificada com propostas políticas existentes, facilitando o processo de criação e aprimoramento de planos de governo por meio de consultas em temas específicos.

No sentido de alcançar a implementação do sistema e validar a prova de conceito, o estudo se baseou em três objetivos secundários:

- Construir o banco de dados vetorial de planos de governo de candidatos políticos
- Criar o mecanismo de recuperação da informação
- Desenvolver o sistema de informação em formato de ChatBot

1.4 Abordagem Teórica

Segundo [3], a teoria da recuperação da informação (RI) se concentra em buscar, classificar e apresentar os documentos relevantes de uma coleção, em resposta à consulta de um usuário. Os processos principais envolvidos são: indexação, operações de consulta, busca e classificação e interface do usuário.

Esse campo é fundamentado em modelos matemáticos e computacionais, que analisam a similaridade entre a consulta do usuário e os documentos disponíveis com base em técnicas que avaliam a correspondência semântica, a frequência de termos e a estrutura dos dados.

Os autores [3] também destacam a importância da avaliação na RI, utilizando métricas como revocação, precisão e medidas de classificação para determinar a eficácia do sistema. Segundo os autores, encontrar o equilíbrio ideal entre revocação e precisão é um desafio na RI.

No presente estudo, a teoria de RI é aplicada baseada no modelo vetorial de recuperação. Esse modelo representa documentos e consultas como vetores em um espaço multidimensional, onde cada termo ou palavra ocupa uma dimensão específica. Neste modelo, a relevância de um documento para uma consulta é calculada com base na similaridade de seus vetores — neste caso, medida pelo cosseno do ângulo entre os vetores. Se o ângulo for pequeno, a similaridade é maior, indicando que o documento é relevante para a consulta.

A RI e o RAG estão fortemente relacionados, uma vez que o RAG depende de técnicas de RI para fornecer informações relevantes que servem de base para a geração de respostas de linguagem natural. Portanto, o RAG pode ser considerado uma integração do modelo vetorial da RI com os LLMs.

COLOCAR ISSO AQUI contribuição e sua originalidade científica, apresentando a justificativa e relevância do trabalho;

2 Trabalhos Relacionados

A técnica RAG emergiu como uma promissora abordagem na área de processamento de linguagem natural (PLN), através do trabalho em conjunto dos LLMs, responsáveis por facilitar o entendimento do usuário e gerar respostas em linguagem natural, com a recuperação de dados disponibilizados em inúmeros fontes e domínios. Inúmeros estudos têm explorado as aplicações RAG em vários assuntos de

pesquisa, revelando seu potencial em melhorar as respostas em sistemas baseados em diálogos e perguntas-respostas. [16]

O artigo seminal que introduziu os conceitos básicos da técnica RAG foi publicado em 2020. Os autores buscavam solucionar o problema de limitação da precisão das LLMs em tarefas intensivas de conhecimento. Além disso, fornecer proveniência para suas decisões e a atualização do conhecimento ao longo do tempo foram dificuldades encontradas pelos autores. [11]

Uma notável revisão sistemática[10] descreve os rápidos avanços em RAG. Os autores reconhecem as limitações significativas dos LLMs e reconhece que a técnica RAG reduz efetivamente o problema de conteúdo factóide, estabelecendo como tecnologia-chave no avanço dos chatbots.

Uma das principais contribuições deste estudo foi a apresentação da árvore tecnológica da pesquisa RAG. Ela descreve três modos de envolvimento da RAG com os LLMs, cada uma com uma abordagem distinta:

- **Inferência:** concentra os primeiros estudos na técnica. A informação é recuperada e enviada à LLM no momento em que ele gera uma resposta ao usuário, acessando dados em tempo real.
- **Ajuste-fino:** Após o aprofundamento da técnica, a mesma foi se integrando ao ajuste fino em LLMs pré-treinadas. O processo de ajuste fino é incorporado com buscas e integração de informações.
- **Pré-treinamento:** Por fim, os grandes modelos de linguagem começaram a ser treinados desde sua fundação utilizando conjunto de dados expandidos através de RAG.

Além desta contribuição, a revisão classifica a pesquisa RAG em três paradigmas: ingênuo, avançado e modular. Segundo os autores, apesar do RAG ingênuo ter custo-benefício e superar o LLM nativo, ele apresenta limitações. O RAG avançado e o modular surge como resposta a essas deficiências.

- **Ingênuo:** representa a metodologia inicial utilizada. Segue uma estrutura tradicional de “recuperação-leitura” em três etapas: indexação, recuperação e geração. Este paradigma possui problemas com precisão e recall, além de criar alucinações quando o modelo gera conteúdo além dos documentos recuperados.
- **Avançado:** utilizando estratégias de pré e pós-recuperação, o paradigma avançado aprimora o RAG ingênuo. São usadas técnicas de otimização da indexação. A consulta inicial é transformada por refinamento e reescrita, na fase pré-recuperação. Já na fase pós, métodos como reordenação dos fragmentos e compressão do contexto ajudam a selecionar e priorizar informações relevantes.
- **Modular:** é um aprimoramento dos paradigmas anteriores, com adição de módulos de busca e refinamento do recuperador via ajuste fino.

O artigo [12] propõe uma nova abordagem para aprimorar a pesquisa em banco de dados usando LLMs. Em vez de utilizar diretamente a consulta do usuário, os autores introduzem uma etapa de reescrita de consulta, visando preencher a lacuna entre a necessidade de informação do usuário e a maneira como essa necessidade é expressa. O conceito central envolve o emprego de um LLM para gerar uma consulta mais eficaz, que é posteriormente utilizada por

um mecanismo de busca (como o Bing) para recuperar informações relevantes. Essa técnica se mostra particularmente benéfica em cenários em que os usuários formulam perguntas complexas ou ambíguas, como em tarefas de resposta a perguntas multi-hop. Os autores demonstram que essa reescrita de consulta, seja realizada por um LLM pré-treinado ou um modelo menor treinado especificamente para essa tarefa, leva consistentemente a um melhor desempenho de pesquisa. O artigo postula que a reescrita de consulta facilita uma pesquisa mais precisa, resultando na recuperação de documentos mais relevantes e, conseqüentemente, respostas mais precisas do LLM.

3 Metodologia

A metodologia deste estudo é estruturada em torno da implementação do PalanqueIA e posterior avaliação. Para validar a prova de conceito do sistema, foram delineados três objetivos principais: a construção de um banco de dados vetorial contendo planos de governo dos candidatos, o desenvolvimento de um mecanismo eficiente de recuperação de informações e a criação de uma interface interativa em formato de ChatBot.

O primeiro objetivo envolveu a coleta e organização de dados de propostas políticas, permitindo a formação de um repositório robusto que serve como base para as consultas dos usuários. Em seguida, o mecanismo de recuperação foi projetado para otimizar a busca por informações relevantes, utilizando algoritmos avançados que garantem a precisão e a relevância dos dados apresentados. Por último, o sistema foi desenvolvido em uma plataforma amigável, possibilitando a interação fluida entre candidatos, eleitores e formuladores de políticas públicas.

A aplicação das teorias de Recuperação da Informação ao modelo vetorial fundamenta a abordagem, permitindo que as respostas geradas sejam contextualizadas e alinhadas às necessidades dos usuários.

O código aplicado no desenvolvimento o estudo encontra-se disponível em repositório anônimo. [2]

O sistema desenvolvido para a submissão de artigos pode ser acessado [14], onde é possível visualizar suas funcionalidades.

A seguir, segue o desenvolvimento detalhado de cada objetivo secundário do estudo:

3.1 Desenvolvimento do banco de dados vetorial

O processo da criação do banco de planos de governo foi projetado em cinco etapas sequenciais, desde a extração dos dados no sítio do TSE até a persistência em um banco de dados específico para fins vetoriais. A Figura 1 ilustra, em detalhes, o processo de construção do banco utilizado neste projeto, destacando os seus componentes.

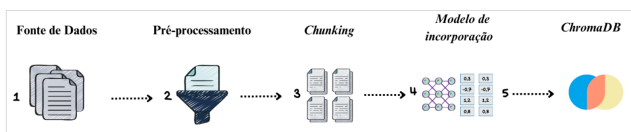


Figure 1: Fluxo das etapas na construção do banco de dados vetorial

A primeira etapa consiste na coleta dos arquivos disponibilizados em formato .PDF, especificamente planos de governo de candidatos a cargos executivos nas eleições municipais de 2024. Esses documentos foram obtidos no sítio oficial do Tribunal Superior Eleitoral (TSE) [18]. Para a leitura dos arquivos, foi empregada a biblioteca PyPDF2 [15], que permitiu processar e armazenar os documentos como objetos estruturados. Na construção dos metadados, foram utilizados dados das candidaturas disponíveis na mesma plataforma do TSE.

Na segunda etapa, de pré-processamento, foi constatada uma considerável diversidade de formatos, tipos e estruturas nos planos de governo disponíveis. Essa heterogeneidade nos arquivos representa um desafio significativo para a conformidade do banco de dados. Para abordar os arquivos em formato de imagem, foi empregada a biblioteca em Python (madmaze, 2024), que possibilitou a conversão das imagens em strings, na medida do possível. Aproximadamente 8% dos arquivos foram descartados devido a resultados insatisfatórios no tratamento de dados. A seguir, apresenta-se um exemplo que ilustra a falta de padronização nos documentos mencionados.

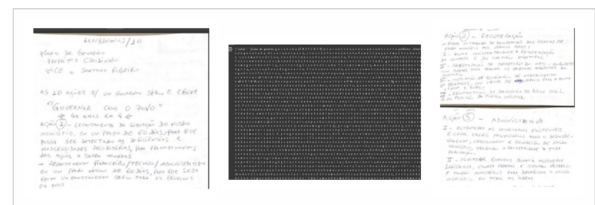


Figure 2: Exemplos de planos de governo disponibilizados

Na continuidade da etapa de pré-processamento, foram aplicadas técnicas de limpeza de dados com o intuito de padronizar e otimizar as informações para a criação de vetores. Entre as técnicas utilizadas, destacam-se a remoção de caracteres indesejados e a conversão do texto para letras minúsculas, entre outras. A granularidade de recuperação foi estabelecida no nível de parágrafo, permitindo que a resposta incluísse tanto o antecessor quanto o sucessor do parágrafo em questão.

A terceira etapa, de fragmentação (chunking), envolve a divisão do texto extraído em partes menores, adequadas para a mensuração da similaridade entre o texto da consulta e o banco de dados. O tamanho definido para cada fragmento foi de 256 tokens, e um overlay de 20 tokens foi aplicado.

O framework empregado para o gerenciamento do banco de dados vetorial foi o ChromaDB. Conforme descrito em seu site [5], Chroma é um banco de dados vetorial de código aberto, nativo para inteligência artificial, que oferece ferramentas para o armazenamento de *embeddings* e metadados, bem como para a geração, consultas e pesquisa desses *embeddings*.

Para a geração dos *embeddings*, utilizou-se o modelo sentence-transformers/all-MiniLM-L6-v2 [8], disponível na plataforma huggingface.co. Nesta fase de incorporação dos dados, o modelo é responsável por gerar as representações vetoriais dos textos as quais serão armazenados no banco.

Por fim, a etapa de armazenamento consiste na persistência dos dados vetoriais gerados no banco. Durante essa fase, ocorre a indexação, onde os fragmentos enriquecidos com informações de metadados, como nome do candidato, partido e outras informações extraídas dos dados dos candidatos coletados na etapa 1, são gravados no banco de dados.

Ao término do processo de desenvolvimento do banco de dados, a base gravou 14.197 propostas políticas, fragmentadas em 1.768.177 *chunks*, conforme ilustrado na tabela a seguir.

Table 1: Quantidade de fragmentos por Estado

Estados	Fragmentos	Estados	Fragmentos
AC	7.754	MG	259.482
AL	26.054	MS	29.789
AM	24.479	MT	43.456
AP	6.949	PA	62.303
BA	173.226	PB	52.504
CE	62.539	PE	57.611
ES	39.795	PI	51.925
GO	62.610	PR	122.548
MA	69.118	RJ	64.371
RN	46.353	RO	16.830
RR	3.864	RS	100.370
SC	72.560	SP	256.654
SE	25.999	TO	29.034
TOTAL	1.768.177		

Conforme se observa nas nuvens de palavras a seguir, existe a predominância de alguns assuntos em detrimento de outros, tais como educação, saúde e segurança. As nuvens de palavras foram construídas utilizando as bibliotecas nltk e wordcloud. Além das palavras stopwords padrão do nltk, foi acrescentada à lista: a, o, é, em, de, ao, e, da, que, plano, governo, municipal, município, políticas, programa, programas, públicos, público e pública.



Figure 3: Nuvem de palavras por frequência

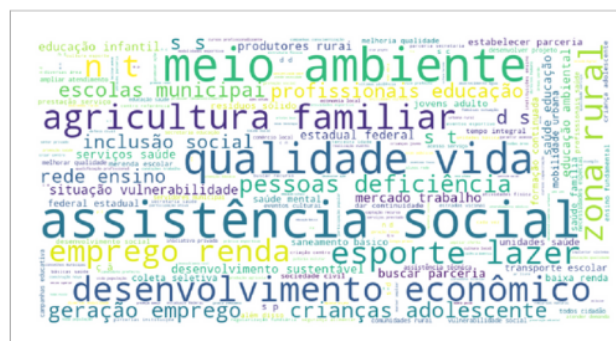


Figure 4: Nuvem de palavras - 200 mais relevantes

Por fim, na figura 5, apresentamos a distribuição de frequência da quantidade de palavras em cada fragmento. O valor médio é de 31,4 palavras e o desvio-padrão de 9,3 palavras.

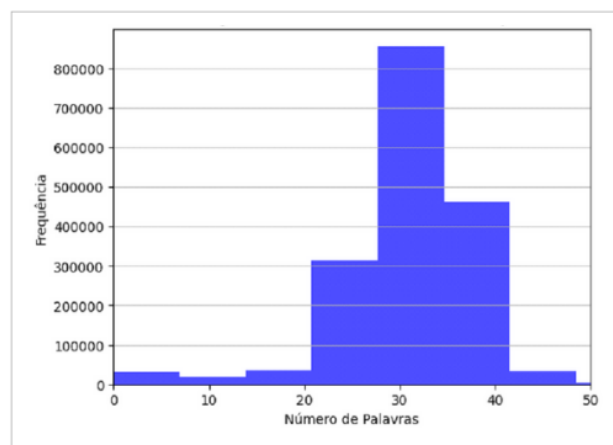


Figure 5: Histograma da quantidade de palavras por fragmento

3.2 Mecanismo de Recuperação da Informação

O mecanismo de recuperação da informação, em outras palavras, a engenharia de dados, desde a entrada do usuário na interface até o retorno ao mesmo, pode ser dividida em 6 etapas principais, conforme ilustração a seguir:

A seguir, segue o detalhamento de cada etapa do mecanismo da engenharia de dados do estudo:

3.2.1 Etapa 1. O fluxo de dados se inicia a partir de uma consulta do usuário interessado em acessar propostas de governo acerca de um tema específico ou de algum candidato em especial. A etapa inicial consiste no envio e recebimento da informação da consulta inicial.

3.2.2 Etapa 2. O propósito dessa etapa é otimizar a consulta inicial do usuário a fim de recuperar informações realmente relevantes

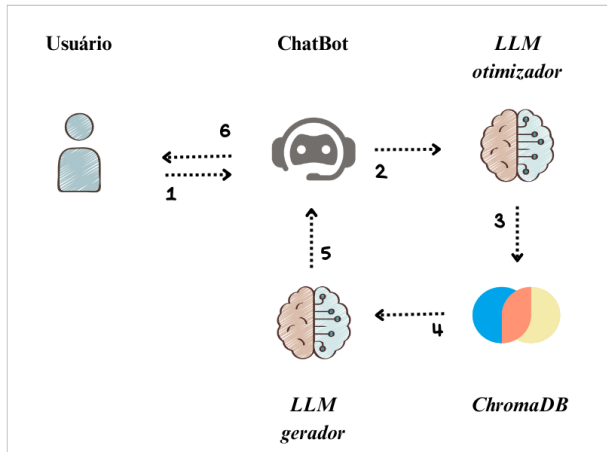


Figure 6: Engenharia de Dados do PalanqueIA

do banco de dados. Um dos principais desafios do RAG é sua dependência com a consulta original do usuário. É comum termos formulações imprecisas, imprudentes e de difícil entendimento, além dos problemas de vocabulários, regionalismos e outras dificuldades linguísticas e semânticas. Para contornar essa problemática, executamos um processo de melhoramento na consulta inicial. Esse processo é realizado em fases conforme fluxo abaixo:

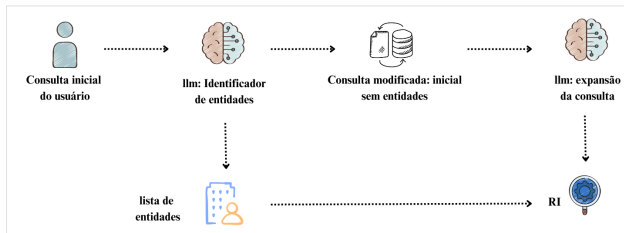


Figure 7: Processo de otimização da consulta inicial

A primeira fase se resume a identificação de entidades relacionadas. O modelo utilizado foi o FacebookAI/xlm-roberta-large-finetuned-conll03-english disponibilizado na plataforma huggingface.co. O objetivo de separação das entidades é criar uma relevância maior às mesmas para o processo de recuperação, e ainda, evitar que o modelo de otimização da consulta crie alucinações e informações baseadas em entidades, uma vez que, a partir de testes locais, foi possível observar que o fato de existir entidades no prompt aumenta consideravelmente a probabilidade de alucinação. (Rewrite extended)

A partir desta identificação e separação, essa função retorna dois objetos: a lista de entidades da consulta e a consulta modificada sem as entidades. Enquanto a lista de entidades é armazenada para ser posteriormente anexada novamente à consulta para ser parte da consulta do sistema de recuperação RAG, a consulta modificada sem as entidades é encaminhada ao modelo Meta Llama 3.1 405B com os seguintes parâmetros:

System content: “Você é um candidato político.” User content: “Retorne em poucas palavras 3 propostas diferentes de acordo com o seguinte pedido” + consulta inicial sem entidades.

Por fim, o sistema agrega a resposta da llm Llama – llm de expansão junto à lista de entidades para compor o texto final a ser realmente encaminhado ao recuperador de informação.

3.2.3 Etapa 3. Nesta etapa, a consulta otimizada é enviada para o Chroma com o objetivo de buscar textos similares, isto é, documentos relevantes para a resposta final. Esta etapa é nevrálgica para o processo de RAG. Esta etapa pode ser dividida em duas fases: geração de embeddings e busca no ChromaDB.

- **Geração de embeddings:** Nesta fase, a consulta otimizada é transformada em um vetor de embeddings – representações numéricas da consulta que capturam o significado semântico. O modelo utilizado é o sentence-transformers/all-MiniLM-L6-v2.
- **Busca no ChromaDB:** Para encontrar documentos semelhantes no banco de dados ChromaDB utilizamos uma busca de similaridade com o uso da distância similaridade do cosseno, conforme equação abaixo:
- Após a recuperação dos cinco fragmentos mais similares, retornamos à base para recuperar os fragmentos vizinhos do principal. É verificado se os vizinhos são da mesma proposta e juntada ao principal.
- Por fim, foi realizado, a partir de testes locais de qualidade de resposta, um filtro na distância mínima para o fragmento ser classificado como similar. O corte definido foi até 0,40 para o fragmento ser relevante. Para ser realizado este teste, foram considerados 100 prompts com temas relevantes e 100 prompts com assuntos aleatórios. Com base nesse experimento, o maior valor com temas relevantes foi de aproximadamente 0,40.

3.2.4 Etapa 4. Esta etapa consiste em sintetizar as informações recuperadas ao longo do processo para o retorno ao usuário final (llm gerador). Para a execução do resumo das informações levantadas, foi utilizado o modelo Llama 3.1 405B com os seguintes parâmetros:

System content: “Você é um assistente que resume em poucas palavras com foco em” + consulta modificada sem entidades User content: documentos relevantes retornado pela RAG

3.2.5 Etapa 5. Essa etapa se resume a coletar a resposta do llm gerador e encaminhar ao chatbot.

3.2.6 Etapa 6. Por fim, nessa etapa, o ChatBot prepara a resposta final e a envia ao usuário, e encerra a atividade sem guardar memória.

4 Arquitetura Modular do Sistema

A seção abaixo detalha os componentes principais do sistema de informação de forma resumida. O sistema é composto por quatro módulos, conforme imagem a seguir.

Como previamente visto no Gráfico 1, a arquitetura do sistema é composta de três módulos: *Front-End*: Interface do usuário *Back-End* LLMs Endpoints

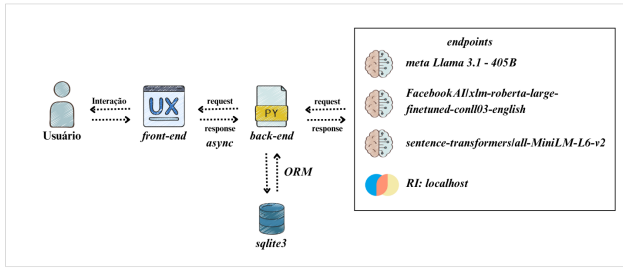


Figure 8: Arquitetura do PalanqueIA

4.1 Front-End

O Front-End representa a camada de apresentação do sistema, projetada para fornecer uma interface de usuário intuitiva e interativa. Ele foi implementado utilizando tecnologias HTML, CSS e JavaScript, criando um aplicativo de página única baseado em uma interface de bate-papo entre o usuário e o sistema. Este design permite que o usuário interaja de forma contínua e dinâmica, oferecendo uma experiência de usuário fluida e responsiva. A camada front-end é responsável por coletar consultas do usuário e exibir respostas processadas, visando simplificar a interação do usuário com as funcionalidades complexas do sistema. O ChatBot inicia a interação com o seguinte texto:

"Olá! Sou sua assistente virtual especializada em busca de propostas políticas. Estou aqui para ajudá-lo a explorar e entender diferentes iniciativas governamentais e políticas públicas, desde planos de incentivo à saúde até estratégias de gestão de recursos públicos. Se desejar buscar por planos de governos ou ver propostas inovadoras de determinado tema, é só pesquisar! Vamos começar?"

A interface possui alguns recursos úteis para melhorar a experiência do usuário, conforme mostrado na ilustração abaixo:



Figure 9: Interface de Usuário do PalanqueIA

- Botões para definição o modo de ação do PalanqueIA: PESQUISA ou CRIAÇÃO
- Novo Chat: abre uma nova janela com um Chat em branco
- Exemplos Práticos: fornece alguns prompts padrão para auxiliar na consulta do usuário

4.2 Back-End

O Back-End do PalanqueIA serve como o núcleo de processamento, mediando a comunicação entre o Front-End, os LLMs *endpoints* e o sistema de recuperação de informações. Implementado usando o framework em Python FastAPI, o módulo foi projetado para lidar com solicitações assíncronas, garantindo respostas rápidas e eficientes às interações do usuário. Suas responsabilidades incluem: Receber a consulta inicial do usuário e otimizar o texto para maximizar a precisão das respostas geradas pelos LLMs. Encaminhando a consulta otimizada para o módulo RAG, que retorna documentos relevantes para o contexto da consulta. Enviando a consulta e os documentos recuperados para um modelo LLM para gerar uma resposta contextualizada e precisa. Retornando a resposta processada para o Front-End, que a apresenta ao usuário. Para persistência de dados, como logs de interação e informações de acesso, o back-end utiliza um banco de dados SQLite3 em conjunto com um Mapeamento Objeto-Relacional (ORM), facilitando o armazenamento e a recuperação de dados estruturados e seguros.

4.3 Endpoints

Os *endpoints* são utilizados para acessar os LLMs que suportam tarefas como geração de texto, resumo e inferência e o banco de dados vetorial. Os modelos são hospedados na plataforma Hugging Face, a saber:

meta-llama/Llama-3.1-405B [13]: A coleção Meta Llama 3.1 de modelos multilíngues de linguagem grande (LLMs) é uma coleção de modelos generativos pré-treinados e ajustados para instruções nos tamanhos 8B, 70B e 405B (entrada de texto/saída de texto). Os modelos somente de texto ajustado para instruções Llama 3.1 (8B, 70B, 405B) são otimizados para casos de uso de diálogo multilíngue e superam muitos dos modelos de software livre e bate-papo fechado disponíveis em benchmarks comuns do setor.

FacebookAI/xlm-roberta-large-finetuned-conll03-english [9]: O modelo XLM-RoBERTa foi proposto em Unsupervised Cross-lingual Representation Learning at Scale. É baseado no modelo RoBERTa do Facebook lançado em 2019. É um grande modelo de linguagem multilíngue, treinado em 2,5 TB de dados CommonCrawl filtrados. Este modelo é XLM-RoBERTa-large ajustado com o conjunto de dados conll2003 em inglês.

Sentence-transformers/all-MiniLM-L6-v2 [17]: O modelo destina-se a ser usado como um codificador de frases e parágrafos curtos. Dado um texto de entrada, ele gera um vetor que captura a informação semântica. O vetor de sentença pode ser usado para tarefas de recuperação de informações, agrupamento ou similaridade de sentenças.

RI-Recuperador Informação

Cada modelo é selecionado e utilizado com base em suas especificidades, garantindo precisão e relevância na resposta do sistema.

5 Resultados e Discussão

Modo: Preciso consulta: retorne projetos na área de oncologia tempo de espera: 22 segundos respostas: Candidato: LÚDIO FRANK MENDES CABRAL, de CUIABÁ-MT

Projeto de Lúdio Frank Mendes Cabral:

- **Prevenção e diagnóstico precoce de câncer** - Acompanhamento pela atenção primária - **Serviço de atendimento com**

assistentes sociais e psicólogos para pessoas com suspeita ou diagnóstico de câncer - Organização do acesso à rede hospitalar pública e contratada para atender demanda de média e alta complexidade.

Candidato: JOAO CARLOS BARBOSA SANTOS, de SERRANÓPOLIS DE MINAS-MG

Projeto de Apoio ao Paciente com Câncer em Serranópolis de Minas-MG:

- Detectar necessidades individuais e familiares para política assistencial efetiva - Criar programa de apoio ao paciente com câncer com: + Acesso rápido ao tratamento + Ajuda psicológica ao enfermo e sua família - Fortalecer vigilância em saúde - Valorizar atuação do Conselho Municipal de Saúde - Contratar médicos especialistas (pediatra, ginecologista) para atendimento diário.

reposta do ChatGPT-4: Nas eleições municipais de 2022 no Brasil, as propostas de candidatos para a área de oncologia e saúde pública variaram significativamente, focando principalmente no acesso a tratamentos e na estruturação de redes de atenção ao câncer. Alguns candidatos priorizaram a melhoria no acesso a medicamentos oncológicos nas unidades de saúde, integrando farmácias locais para facilitar o tratamento em regiões mais afastadas dos grandes centros, onde a logística é um desafio. Esse modelo permite que a medicação seja distribuída diretamente aos pacientes através de farmácias regionais, reduzindo deslocamentos e agilizando o acesso ao tratamento OBSERVATÓRIO HOSPITALAR FIOCRUZ SCIELO

Em termos de políticas públicas, alguns programas municipais também reforçaram o papel do Sistema Único de Saúde (SUS) na prevenção e controle do câncer, incluindo ações como o diagnóstico precoce e o suporte ampliado para tratamento hormonal e cuidados paliativos. A descentralização do tratamento, em que municípios vizinhos auxiliam no acompanhamento de pacientes que precisam buscar tratamento fora de sua cidade, foi uma medida discutida, embora haja desafios na efetiva adesão e uso correto dos medicamentos distribuídos SCIELO

Para mais detalhes sobre os programas e as abordagens que foram defendidas em várias cidades, o Observatório de Política e Gestão Hospitalar oferece um panorama das propostas de saúde nas eleições municipais, inclusive para 2024, que podem fornecer insights comparativos sobre a evolução das iniciativas políticas na área da oncologia OBSERVATÓRIO HOSPITALAR FIOCRUZ

consulta: traga iniciativas sobre o tema combate à criminalidade tempo de espera: 18 segundos respostas: Candidato: GONSALO ANTONIO MENDES DE MAGALHÃES, de ICARAÍ DE MINAS-MG

Proposta de Gonsalo Antonio Mendes de Magalhães:

- Objetivo: Aumentar a segurança e prevenir crimes - Estratégias: + **Fortalecer as forças de segurança com mais efetivo policial** + Implementar programas de prevenção comunitária em parceria com escolas e empresas locais + Realizar campanhas educativas e de conscientização sobre segurança no trânsito nas escolas (Semana do Trânsito)

Candidato: PEDRO ANTONIO PEREIRA MALHEIROS, de SEBASTIÃO LARANJEIRAS-BA

Proposta de Pedro Antonio Pereira Malheiros:

- **Fortalecer políticas de segurança com mais presença policial e tecnologias de vigilância.** - Implementar programas comunitários de prevenção ao crime, focando em educação e inclusão

social de jovens. - Objetivo: melhorar a segurança e a qualidade de vida dos cidadãos.

Modo criativo: consulta: Forneça uma proposta de governo sobre o assunto saúde do idoso tempo: 20 segundos

Segue aqui uma proposta de governo para a saúde do idoso:

Proposta: "Saúde do Idoso: Atenção Integral e Humanizada"

Objetivo: Garantir o acesso a serviços de saúde de qualidade, humanizados e especializados para a população idosa, promovendo a prevenção, o tratamento e a reabilitação de doenças crônicas e condições específicas da idade avançada.

Eixos de Ação:

1. Atenção Primária: Fortalecer a atenção primária à saúde, com equipes de saúde da família treinadas para atender às necessidades específicas da população idosa, incluindo a prevenção e o tratamento de doenças crônicas. 2. Especialidades Médicas: Garantir o acesso a especialidades médicas, como geriatria, nefrologia, cardiologia e neurologia, para atender às necessidades específicas da população idosa. 3. Reabilitação e Fisioterapia: Oferecer serviços de reabilitação e fisioterapia para ajudar a população idosa a manter a mobilidade e a independência. 4. Saúde Mental: Promover a saúde mental da população idosa, com ações de prevenção, diagnóstico e tratamento de condições como depressão, ansiedade e doenças neurodegenerativas. 5. Assistência Domiciliar: Oferecer assistência domiciliar para a população idosa que necessita de cuidados especiais, incluindo a entrega de medicamentos e o acompanhamento médico. 6. Educação e Prevenção: Promover a educação e a prevenção sobre doenças crônicas e condições específicas da idade avançada, com ações de saúde pública e campanhas de conscientização. 7. Parcerias e Redes: Estabelecer parcerias com instituições de saúde, hospitais e organizações não governamentais para garantir a articulação e a integração dos serviços de saúde para a população idosa.

Metas:

- Reduzir a taxa de mortalidade por doenças crônicas em 20% em 4 anos. - Aumentar a cobertura de atenção primária à saúde para 90% da população idosa em 4 anos. - Garantir o acesso a especialidades médicas para 80% da população idosa em 4 anos. - Reduzir a taxa de hospitalização por condições evitáveis em 30% em 4 anos.

Orçamento: R\$ 10 milhões anuais, para serem destinados à implementação das ações e metas propostas.

Essa proposta de governo visa garantir a saúde e o bem-estar da população idosa, promovendo a prevenção, o tratamento e a reabilitação de doenças crônicas e condições específicas da idade avançada. Além disso, busca fortalecer a atenção primária à saúde, garantir o acesso a especialidades médicas e promover a saúde mental da população idosa.

Por meio da leitura das respostas acima entregues pelo sistema, conclui-se que é possível obter informações úteis e satisfatórias, em um tempo razoável, em se tratando de planos de governo a partir de um chatbot, utilizando RAG e LLMs.

Em relação à comparação do resultado obtido entre o PalanqueIA e o ChatGPT-4, é possível observar que, para certos domínios, os chatbots desenvolvidos utilizando RAG são mais específicos em suas respostas, ressaltando o potencial do uso da técnica para assistentes especializados em assuntos de um domínio particular.

Caso seja feita a comparação do retorno obtido com uma pesquisa manual, é impensável o tempo gasto de um interessado em fazer a

leitura de 14.197 planos de governo a fim de recuperar os projetos em determinado assunto. Porém, em um cálculo grotesco, podemos calcular da seguinte forma: se cada fragmento for lido em 10 segundos, gastaríamos 17.681.770 segundos, equivalente à 4.911 horas de leitura.

6 Conclusão

O estudo demonstrou a viabilidade do uso de técnicas RAG integradas aos LLMs para criar um sistema inteligente de recuperação e proposição de planos de governo. O PalanqueIA, um chatbot desenvolvido como prova de conceito, demonstrou a capacidade de:

- Aperfeiçoar a consulta inicial do usuário através de um mecanismo inovador de reescrita da consulta inicial, utilizando o reconhecimento de entidades;
- Consultar uma base de dados vetorial com mais de 1,7 milhão de fragmentos de propostas políticas;
- No modo de Consulta, fornecer respostas relevantes e contextualizadas às consultas dos usuários;
- No modo de criação, gerar propostas de Governo criativas a partir de outras pré-existentes.

A comparação com o ChatGPT-4 revelou que o PalanqueIA, por ser especializado em planos de governo, oferece respostas mais específicas e focadas no domínio político.

A eficiência do PalanqueIA se destaca ao considerar a extensão da base de dados: a leitura manual dos 14.197 planos de governo, a uma taxa estimada de 10 segundos por fragmento, consumiria 4.911 horas.

O estudo evidencia o potencial do RAG e LLMs para aprimorar o processo eleitoral. O PalanqueIA pode:

- Auxiliar candidatos na formulação de propostas políticas inovadoras e embasadas em dados;
- Facilitar a análise comparativa de planos de governo por eleitores e analistas políticos;
- Promover um debate público mais qualificado e focado em propostas concretas

A pesquisa contribui para o avanço do campo da Ciência Política Computacional ao aplicar técnicas de Inteligência Artificial para democratizar o acesso à informação e fomentar a participação cidadã.

Sugestões para trabalhos futuros:

- Expandir a base de dados com planos de governo de outros anos e cargos eletivos;
- Avaliar a usabilidade e a satisfação dos usuários com o PalanqueIA;
- Investigar o impacto do sistema no debate público e na qualidade das propostas políticas;
- Criar uma proposta política completa baseada em modelos de gestão de Projetos.

Este estudo se conecta a múltiplos tópicos de interesse da área de Sistemas de Informação. Primeiramente, ele se encaixa no tópico "Sistemas de informação e o mundo aberto", pois o PalanqueIA, acessível online, visa democratizar o acesso à informação sobre planos de governo. A pesquisa também aborda a "Visão sociotécnica de sistemas de informação", considerando tanto os aspectos técnicos

da implementação do PalanqueIA com RAG e LLMs quanto os impactos sociais da ferramenta na participação cidadã e na qualidade do debate público. O processo de desenvolvimento do PalanqueIA, detalhado nas etapas de construção do banco de dados vetorial, do mecanismo de recuperação da informação e da interface do chatbot, ilustra a aplicação de "Paradigmas, modelagem, design, engenharia e avaliação de sistemas de informação". A pesquisa se destaca pela aplicação de "Tecnologias emergentes aplicadas a sistemas de informação", utilizando RAG e LLMs para a análise e proposição de políticas públicas. As questões éticas relacionadas ao uso de dados de planos de governo são abordadas, garantindo transparência e responsabilidade, o que se relaciona ao tópico "Ética, cultura, legislação e política em sistemas de informação". O PalanqueIA, inserido no ecossistema digital do governo eletrônico, conecta candidatos, eleitores e formuladores de políticas, abordando o tema "Ecossistemas digitais e de negócios". Finalmente, a utilização de LLMs e técnicas de PLN no desenvolvimento do chatbot enquadra a pesquisa no tópico "Sistemas de informação e Inteligência Artificial (Generativa, LLM, PLN, entre outros)".

Por fim, este estudo se alinha com os Grandes Desafios de Pesquisa em Sistemas de Informação no Brasil para a Década de 2016-2026, em particular com os desafios Sistemas de Informação e o Mundo Aberto e Metodologias e Tecnologias para a participação cidadã. O PalanqueIA, ao utilizar tecnologias de ponta como RAG e LLMs para democratizar o acesso à informação e promover a participação cidadã, contribui para a criação de um ecossistema digital mais transparente e colaborativo, em consonância com os princípios do GrandSI-BR.

References

- [1] Ana Paula Albarelli. 2020. *Uma análise da descortesia como estratégia de persuasão em interações polêmicas: o debate político*. Ph.D. Dissertation. Universidade de São Paulo. <https://doi.org/10.11606/T.8.2020.tde-18082020-170840>
- [2] Anon. 2024. Model Study. <https://anonymous.4open.science/r/ModelStudy-C21B/>. Accessed: 2024-11-04.
- [3] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. 2011. *Modern Information Retrieval: The Concepts and Technology Behind Search* (second ed.). Addison Wesley, New York.
- [4] Brasil. 2009. Lei nº 12.034, de 29 de setembro de 2009. Institui modificações no Código Eleitoral e na Lei dos Partidos Políticos e estabelece normas para o voto eletrônico. Diário Oficial da União: seção 1, Brasília, DF, 30 set. 2009.
- [5] ChromaDB. 2024. Chroma Docs. <https://docs.trychroma.com/>. Acesso em: [data de acesso].
- [6] Câmara dos Deputados. 2024. *Projeto prevê ampla divulgação de programas de governo de candidatos*. <https://www.camara.leg.br/noticias/788631-projeto-preve-ampla-divulgacao-de-programas-de-governo-de-candidatos/> Acesso em: 24 set. 2024.
- [7] Simone Diniz and Lucy Oliveira. 2020. Programas de Governo e Promessas de Campanha. *Revista de Sociologia e Política* 28, 76 (2020), e006. <https://doi.org/10.1590/1678-987320287606>
- [8] Hugging Face. 2024. sentence-transformers/all-MiniLM-L6-v2. <https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2>. Acesso em: [data de acesso].
- [9] Facebook. 2024. XLM-RoBERTa Large Fine-Tuned on CoNLL-03 (English). <https://huggingface.co/FacebookAI/xlm-roberta-large-finetuned-conll03-english> Accessed: 2024-11-01.
- [10] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2023. Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv preprint arXiv:2312.10997* (2023). <https://doi.org/10.48550/ARXIV.2312.10997>
- [11] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, et al. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *arXiv preprint arXiv:2005.11401* (2020). <https://doi.org/10.48550/ARXIV.2005.11401>
- [12] Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query Rewriting for Retrieval-Augmented Large Language Models. *arXiv* (2023).

- [//doi.org/10.48550/ARXIV.2305.14283](https://doi.org/10.48550/ARXIV.2305.14283)
- [13] Meta. 2024. Llama 3.1 - 405B. <https://huggingface.co/meta-llama/Llama-3.1-405B> Accessed: 2024-11-01.
- [14] Palanqueia. [n. d.]. Palanqueia: O que é e como funciona? <https://www.palanqueia.site/> Accessed: 2024-11-04.
- [15] PyPDF. 2024. Welcome to PyPDF2. <https://pypdf2.readthedocs.io/en/3.x/>. Acesso em: [data de acesso].
- [16] Mujahid Ali Quidwai and Alessandro Lagana. 2024. A RAG Chatbot for Precision Medicine of Multiple Myeloma. <https://doi.org/10.1101/2024.03.14.24304293>
- Preprint.
- [17] Sentence-Transformers. 2024. all-MiniLM-L6-v2. <https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2> Accessed: 2024-11-01.
- [18] Tribunal Superior Eleitoral. 2024. Divulgação de Candidaturas e Contas Eleitorais. <https://divulgacandcontas.tse.jus.br/>. Acesso em: 24 set. 2024.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009