API INTELIGENTE COM IA GENERATIVA E RAG PARA CONSULTAS EM DOCUMENTOS EMPRESARIAIS

INTELLIGENT API WITH GENERATIVE AI AND RAG FOR QUERYING BUSINESS DOCUMENTS

Cauê Paulino Amendola, Lucas Ribeiro, Luiz Antonio Ferraro Mathias, Maurício Neves Asenjo

Universidade Santa Cecília, Curso de Sistemas de Informação
e-mail: paulinocaue@gmail.com

RESUMO – Este trabalho apresenta o desenvolvimento de uma API capaz de processar documentos internos empresariais em formatos PDF e planilha eletrônica, tornando-os pesquisáveis e acessíveis por meio de consultas inteligentes com Inteligência Artificial Generativa. A arquitetura proposta utiliza a metodologia RAG (Geração Aumentada por Recuperação), integrando técnicas de extração de texto, reconhecimento óptico de caracteres (OCR), armazenamento em banco de dados, geração de embeddings e mecanismos de orquestração para consultas inteligentes. Como resultado, os documentos corporativos tornam-se bases de conhecimento para respostas contextuais, superando limitações da busca semântica tradicional e garantindo maior precisão e rapidez para o usuário. A proposta foi validada por meio de um case desenvolvido em ambiente acadêmico, demonstrando potencial para adaptação em empresas de diferentes setores.

Palavras-chave: Inteligência Artificial; RAG; IA Generativa; Processamento de Documentos; Banco de Dados NoSQL.

ABSTRACT – This work presents the development of an API capable of processing internal corporate documents in PDF and Excel formats, making them searchable and accessible through intelligent queries with Generative Artificial Intelligence. The proposed architecture applies the RAG (Retrieval-Augmented Generation) methodology, integrating techniques of text extraction, optical character recognition (OCR), database storage, embeddings generation, and orchestration mechanisms for intelligent queries. As a result, corporate documents become knowledge bases for contextual responses, overcoming the limitations of traditional semantic search and ensuring greater accuracy and speed for the user. The proposal was validated through a case developed in an academic environment, demonstrating potential for adaptation in companies from different sectors.

Keywords: Artificial Intelligence; RAG; Generative AI; Document Processing; NoSQL Databases.

1 INTRODUÇÃO

O avanço das tecnologias de Inteligência Artificial Generativa tem possibilitado a criação de ferramentas capazes de transformar grandes volumes de dados em informações úteis para a tomada de decisão (OpenAI, 2023). Dentro do ambiente corporativo, empresas lidam diariamente com diferentes tipos de documentos — como relatórios, planilhas, contratos e arquivos em PDF — que muitas vezes permanecem pouco aproveitados, seja pela falta de padronização, seja pela dificuldade em acessar informações de maneira rápida e precisa. Essa limitação gera atrasos, retrabalho e perda de eficiência em processos que dependem diretamente de dados confiáveis.

Uma das abordagens mais promissoras para enfrentar esse desafio é a Geração Aumentada por Recuperação (RAG). Essa metodologia combina dois processos centrais: a recuperação de informações armazenadas em bases de dados e a geração de respostas por meio de modelos de linguagem (LangChain, 2023). O funcionamento ocorre em duas etapas principais. Primeiro, tanto as perguntas dos usuários quanto o conteúdo dos documentos são transformados em representações numéricas chamadas *embeddings*, que permitem comparar semelhanças entre diferentes textos. Em seguida, o sistema recupera os trechos mais relevantes para a consulta e os utiliza como base para produzir uma resposta clara e contextualizada.

Para que esse processo seja eficaz, é necessário realizar etapas de extração e organização de dados. No caso de documentos em PDF, o conteúdo textual pode ser coletado diretamente quando o arquivo contém texto editável, ou, em documentos digitalizados, por meio de técnicas de Reconhecimento Óptico de Caracteres (OCR). Já em planilhas do tipo Excel, a leitura das colunas e linhas possibilita identificar padrões e relacionar informações estruturadas. Após a extração, os dados são tratados e armazenados em uma base que permite consultas rápidas e organizadas, garantindo que as informações estejam disponíveis de maneira prática e acessível.

Diante desse cenário, a hipótese considerada neste trabalho é que o uso da metodologia RAG, aliada à Inteligência Artificial Generativa, possibilita maior precisão, rapidez e contextualização nas consultas a documentos internos quando comparada à busca tradicional por palavras-chave. Essa hipótese fundamenta o desenvolvimento de uma solução tecnológica que transforma documentos corporativos em bases de conhecimento, oferecendo resultados mais completos e alinhados às necessidades reais do usuário.

Assim, o presente estudo tem como objetivos: (I) desenvolver uma API capaz de processar documentos em PDF e Excel; (II) implementar um fluxo de consultas baseado na metodologia RAG, capaz de recuperar informações de forma eficiente; e (III) validar a proposta por meio de

um *case* em ambiente acadêmico, demonstrando seu potencial de adaptação para diferentes setores empresariais.

2 MATERIAIS E MÉTODOS

A pesquisa caracteriza-se como um **estudo de caso tecnológico**, com desenvolvimento de *software* em ambiente controlado.

2.1 Arquitetura do Sistema

A solução foi desenvolvida em linguagem de programação Java, utilizando o *framework* Spring Boot, escolhido pela sua robustez e ampla adoção em aplicações corporativas, o que garante escalabilidade e facilidade de integração com diferentes tecnologias.

Para o armazenamento dos dados, optou-se pelo uso do MongoDB, um banco de dados NoSQL adequado para lidar com estruturas flexíveis, escolha esta, motivada pela necessidade de armazenar tanto o conteúdo textual extraído dos documentos quanto os vetores numéricos (*embeddings*), que não seguem um modelo relacional fixo.

No processamento de linguagem natural, foram empregados dois modelos distintos: text-embedding-3-large para a criação dos *embeddings* e GPT-4o para a geração de respostas.

Para a orquestração do fluxo RAG (Geração Aumentada por Recuperação), utilizou-se o LangChain4j, que integra os diferentes módulos da aplicação, coordenando as etapas de recuperação de trechos relevantes, passagem de contexto e geração da resposta final.

O processamento dos documentos foi realizado em duas frentes. No caso de PDFs editáveis, utilizou-se a biblioteca Apache PDFBox para a extração direta de texto. Para arquivos em formato de imagem ou digitalizados, aplicou-se a técnica OCR com a ferramenta Tesseract, possibilitando a conversão de imagens em texto pesquisável (Smith, 2007).

Por fim, foi utilizada a técnica de *chunking*, que consiste em dividir textos extensos em partes menores para facilitar o processamento. Essa divisão é necessária porque os modelos responsáveis pela geração de *embeddings* possuem limites de tokens (unidades de texto) que podem analisar por vez. Neste estudo, os documentos foram segmentados em blocos de até 2200 tokens, de modo a garantir um bom equilíbrio entre desempenho computacional e a preservação do contexto semântico do conteúdo original.

2.2 Fluxo de Processamento

O funcionamento da API foi estruturado em três etapas principais: preparação dos documentos, consulta do usuário e geração da resposta.

Na etapa de preparação, os documentos nos formatos anteriormente citados, foram carregados no sistema. Nos PDFs editáveis, o conteúdo foi extraído diretamente; já nos compostos apenas por imagens, aplicou-se OCR, que converte imagens em texto pesquisável. No caso das planilhas eletrônicas, os dados foram lidos de forma estruturada, identificando colunas e informações relevantes. Após a extração, o texto passou pelo processo de *chunking*.

Na etapa de consulta, o usuário enviava perguntas em linguagem natural, as quais, foram transformadas em *embeddings*, o que permitiu comparar a consulta com os *embeddings* previamente armazenados na base de dados.

Na etapa de resposta, os trechos mais relevantes foram recuperados do banco de dados MongoDB e enviados ao modelo de linguagem GPT-4o, que, a partir desse contexto, gerou uma resposta contextualizada. Assim, o sistema não apenas localizou a informação nos documentos, mas também a apresentou de forma organizada e alinhada à necessidade do usuário.

2.3 Critérios de Validação

A validação da proposta foi conduzida por meio de um *case* acadêmico, simulando um cenário real de consulta a documentos internos. Para esse teste, foram selecionados documentos de natureza acadêmica, incluindo planilhas de horários de professores e arquivos PDF com conteúdo de disciplinas. A escolha desses documentos deve-se ao fato de apresentarem dados estruturados (planilhas) e não estruturados (PDFs), o que permitiu avaliar a eficiência do sistema em diferentes tipos de arquivo.

O processo de validação ocorreu em duas etapas: os documentos foram carregados na API e processados pelo fluxo descrito anteriormente, gerando os *embeddings* e armazenando-os no banco de dados; em seguida, foram formuladas perguntas específicas em linguagem natural, representando casos de uso reais, como por exemplo: “Qual o horário da aula do professor X?” ou “Quais os principais tópicos de Fundamentos de Linguagem de Programação?”.

As respostas fornecidas pela API foram comparadas manualmente com o conteúdo original dos documentos para verificar sua precisão. O desempenho foi considerado satisfatório quando a resposta recuperada coincidia com a informação presente nos arquivos e estava redigida de forma contextualizada. Como critério adicional, foi observada a capacidade do sistema de localizar corretamente os trechos relevantes, mesmo quando a pergunta não continha as mesmas palavras utilizadas no documento, demonstrando a eficiência do modelo em compreender o sentido da consulta.

Essa metodologia de validação permitiu avaliar tanto a exatidão das respostas quanto a capacidade de generalização da API, confirmando sua viabilidade para cenários práticos.

3 RESULTADOS E DISCUSSÃO

Os resultados obtidos evidenciam que a solução proposta é capaz de transformar documentos internos em bases de conhecimento utilizáveis. Durante os testes realizados com planilhas Excel, o sistema apresentou desempenho satisfatório, conseguindo responder corretamente à maioria das consultas diretas. Esse resultado mostra que a API é capaz de interpretar perguntas em linguagem natural e localizar as informações de forma consistente.

No processamento de arquivos PDF, a combinação das bibliotecas de extração de texto e do OCR mostrou-se eficiente, permitindo que mesmo documentos compostos apenas por imagens pudessem ser convertidos em conteúdo pesquisável. Essa característica é especialmente relevante em ambientes corporativos, onde muitos arquivos são digitalizados sem padronização de formato.

Outro ponto observado foi a vantagem do fluxo baseado em RAG em relação às buscas tradicionais por palavra-chave. O sistema não apenas recuperou trechos específicos, mas também produziu respostas contextualizadas, mais próximas da forma como o usuário formula suas perguntas. Essa capacidade reforça o potencial de aplicação da arquitetura em cenários reais, oferecendo maior precisão e praticidade.

Para ilustrar o funcionamento da solução, a Figura 1 apresenta a interface de *upload* de documentos utilizada nos testes, enquanto a Figura 2 mostra um exemplo de consulta e resposta gerada pela API em ambiente acadêmico.

Figura 1 – Tela de upload de documentos na API.

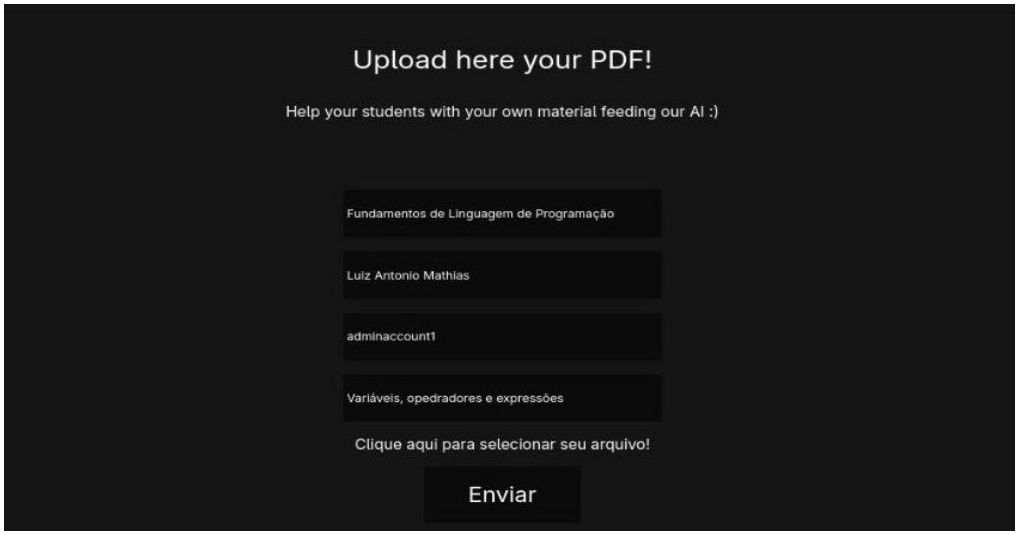
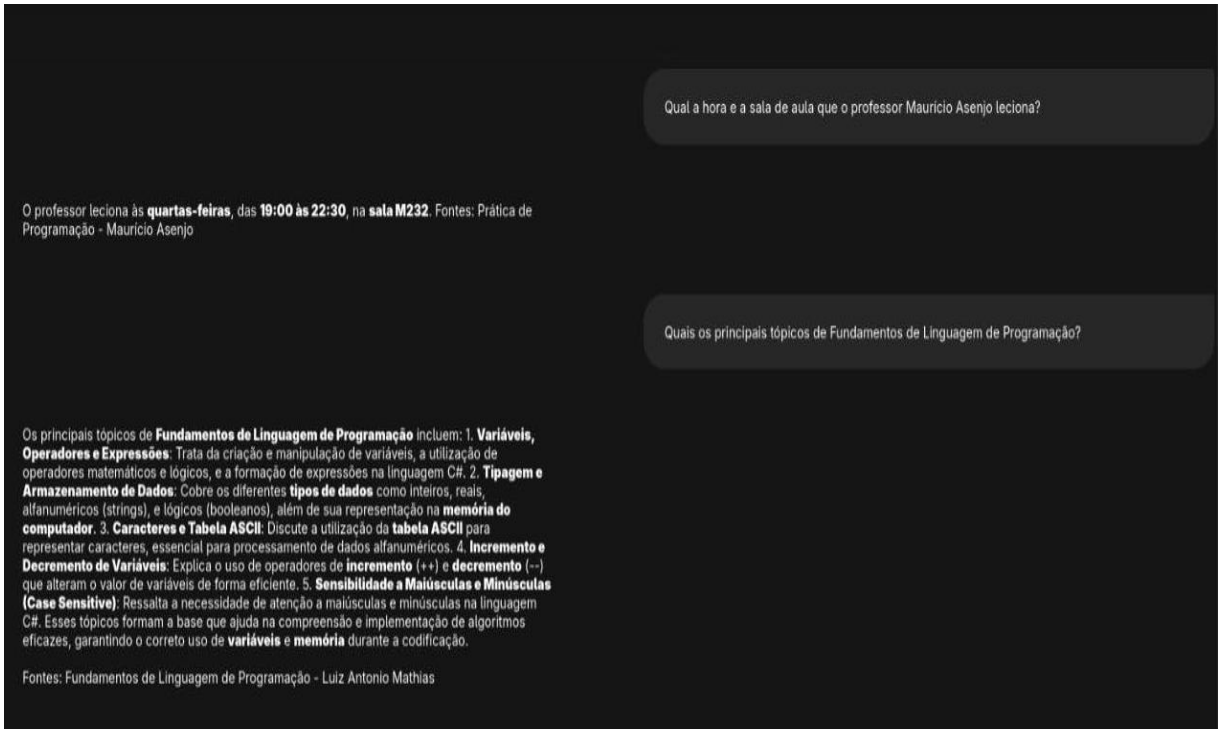


Figura 2 – Exemplo de consulta em linguagem natural e resposta contextualizada.



Os resultados indicaram que a arquitetura proposta é escalável e pode ser aplicada em diferentes setores, como educação, jurídico e financeiro, nos quais a consulta a documentos internos é uma necessidade recorrente (McKinsey & Company, 2023). A principal contribuição deste trabalho, entretanto, está na validação prática em ambiente real, utilizando documentos de uso cotidiano. Essa aplicação prática demonstra não apenas a viabilidade técnica da solução,

mas também sua relevância como alternativa eficiente para o tratamento e utilização de informações corporativas.

4 CONCLUSÃO

O presente estudo apresentou o desenvolvimento de uma API voltada para consultas inteligentes em documentos empresariais, fundamentada na integração entre Inteligência Artificial Generativa e a metodologia RAG. A proposta foi validada por meio de um *case* em ambiente acadêmico, demonstrando sua viabilidade e apontando que o sistema pode oferecer respostas mais precisas, rápidas e contextualizadas em comparação com buscas tradicionais por palavras-chave. Observou-se ainda que a arquitetura é escalável e pode ser adaptada a diferentes setores corporativos. Como trabalhos futuros, pretende-se ampliar o suporte a múltiplos formatos de documentos, estabelecer métricas de avaliação de desempenho e aplicar a solução em empresas reais, reforçando sua contribuição prática e acadêmica. Agradeço aos professores que acompanharam o desenvolvimento desta pesquisa, cujo apoio foi essencial para a construção deste trabalho.

5 REFERÊNCIAS

LangChain. (2023). *LangChain Documentation.* Disponível em: https://docs.langchain.com

McKinsey & Company. (2023). *The Economic Potential of Generative AI: The Next Productivity Frontier.* McKinsey Global Institute. Disponível em: https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-economic-potential-of-generative-ai-the-next-productivity-frontier

OpenAI. (2023). *Introducing ChatGPT.* OpenAI Blog. Disponível em: https://openai.com/blog/chatgpt

Smith, R. (2007). *An Overview of the Tesseract OCR Engine.* Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR), 629–633. Disponível em: https://research.google.com/pubs/archive/33418.pdf