

Palavras-Chave

LLMs (*Large Language Models*), RAG (*Retrieval-Augmented Generation*), Engenharia de Software.

Objetivos e Motivação

O objetivo principal deste trabalho é utilizar LLMs e técnicas de RAG para realizar o levantamento automático de evidências técnicas em projetos de software, focando na identificação da implementação de módulos a partir de contratos de desenvolvimento de software. A necessidade de automação na verificação e validação de requisitos de software em contratos é crescente. Tradicionalmente, a revisão manual desses documentos é trabalhosa e sujeita a erros. A proposta é explorar a aplicação de modelos de linguagem para acelerar esse processo, tornando-o mais preciso e eficiente.

Metodologia e Desenvolvimento

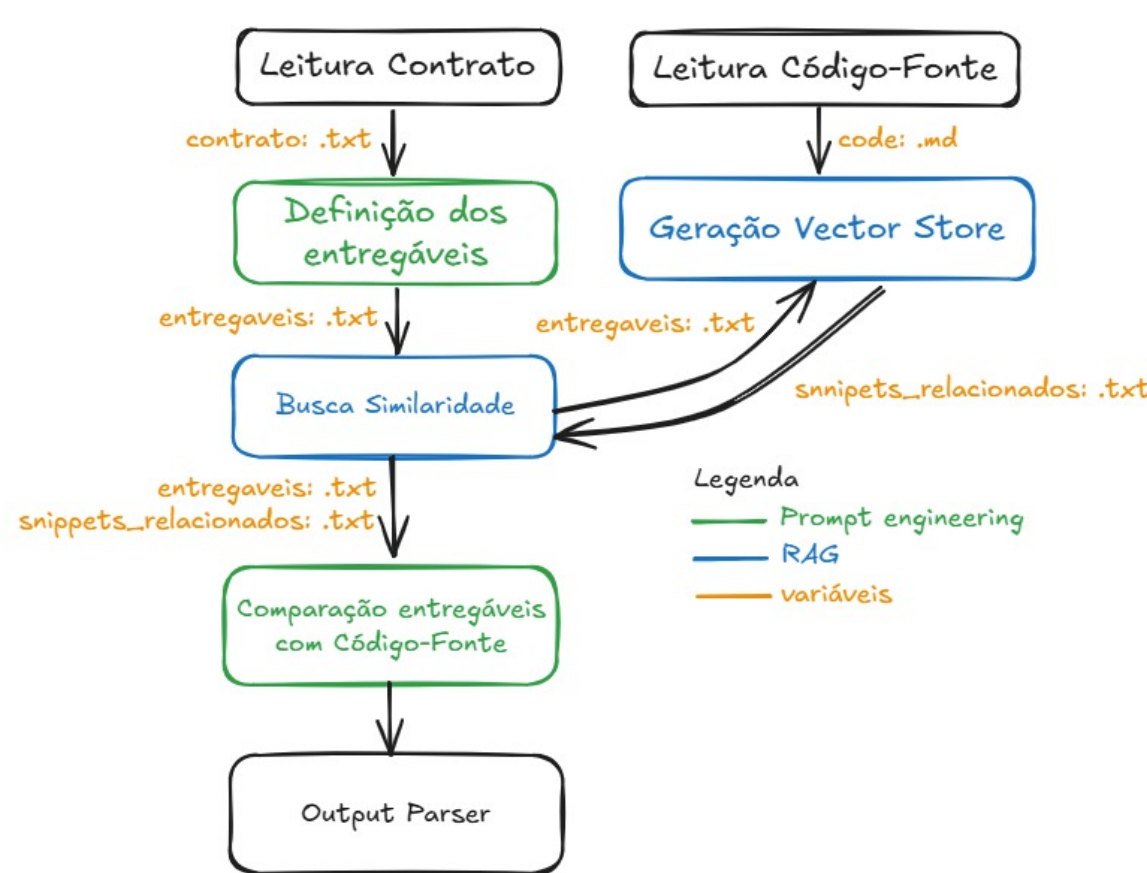


Figura 1. Fluxo metodológico para extração e verificação de entregáveis a partir do contrato e código-fonte

A pesquisa adota uma abordagem qualitativa, utilizando uma *Proof of Concept* (PoC) para investigar o uso de LLMs, para levantamento automático de evidências técnicas no contexto de comercialização de projetos de software.

Etapas do Processo:

- Definição dos Artefatos de Entrada:
 - Artefatos Simulados:** Documentos criados de forma específica para o estudo, simulando parcialmente a documentação de um projeto real.
 - Documentos Reais:** Materiais fornecidos por uma pequena empresa de desenvolvimento de software, representando dados reais de projetos.
- Processo de Análise:
 - Identificação dos Tópicos Comerciais:** Definição dos tópicos que devem ser comprovados por evidências, com base na documentação comercial do projeto.
 - Busca e Indexação das Evidências Técnicas:** Extração das partes técnicas relevantes da documentação técnica, utilizando o LLM para indexação e busca.
 - Relacionamento das Evidências com os Tópicos Comerciais:** Vínculo entre as evidências extraídas na etapa anterior e os tópicos comerciais definidos, criando um banco de dados vetorial para armazenar e reutilizar os dados.

Tecnologias e Ferramentas Utilizadas:

- LLMs: Gemini (Modelo de Linguagem de Última Geração da Google).
- Linguagem de Programação: Python.
- Ambiente de Desenvolvimento: Google Colab.
- Armazenamento de Dados: Banco de Dados Vetorial para indexação e reutilização das evidências técnicas.

Resultados e Discussões

Aspecto	Desafio	Solução/Resultado
Diversidade de Documentos	Variedade em formato e detalhamento	Combinação de LLMs e RAG para extração de informações de contratos e código-fonte
Indexação e Busca	Dificuldade em organizar evidências técnicas	Uso de LLMs para criar um banco de dados vetorial que agiliza a consulta das evidências
Verificação de Requisitos	Alto tempo e esforço na validação manual de requisitos	Redução de tempo e esforço na verificação de requisitos usando LLMs para levantamento automático de evidências

Tabela 1. Resultados e Discussões: Desafios e Soluções

Desafios e Abordagens: O principal desafio foi a diversidade da documentação de software, que varia em formato e nível de detalhamento. A combinação de LLMs e técnicas de RAG mostrou-se eficaz para lidar com essa variedade, permitindo a extração de

informações relevantes de documentos como contratos e código-fonte.

Resultados Preliminares: Os resultados da Proof of Concept (PoC) indicam que a abordagem consegue identificar e relacionar evidências técnicas com tópicos comerciais, reduzindo significativamente o tempo e esforço necessários para validar requisitos de software. A utilização de um banco de dados vetorial facilitou a consulta eficiente das evidências extraídas.

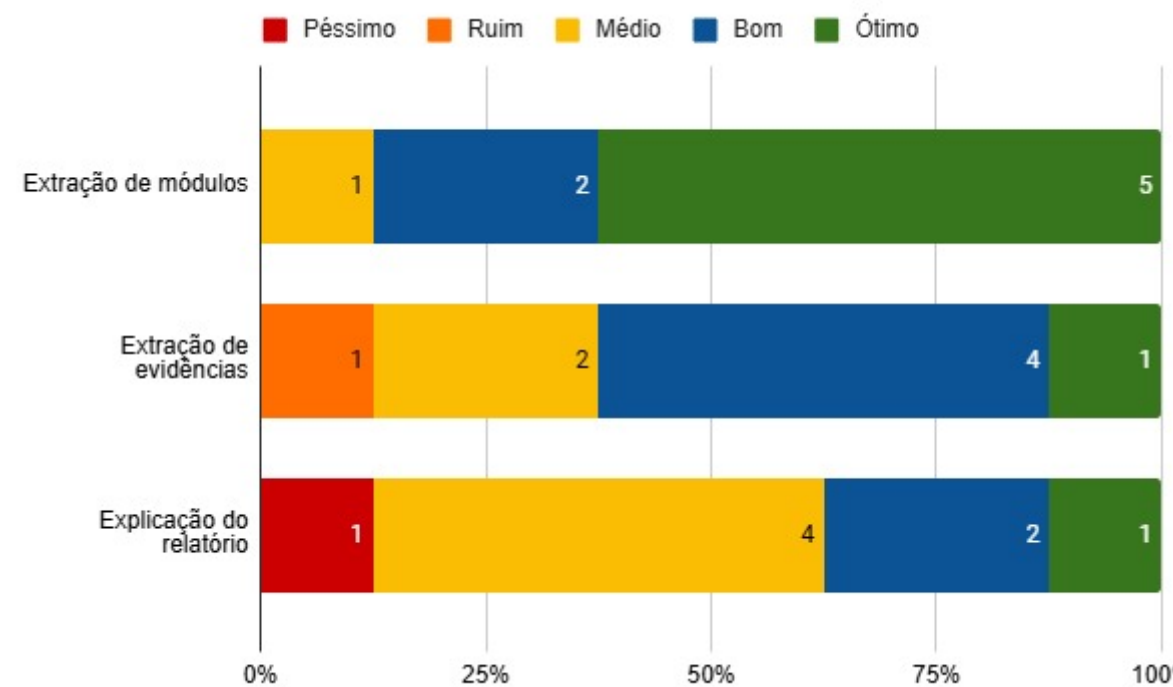


Figura 2. Distribuição das avaliações das funcionalidades implementadas

A avaliação dos resultados revela um desempenho positivo na extração de módulos, com 5 avaliações 'Ótimo' e apenas 1 'Ruim', indicando uma alta taxa de satisfação nessa etapa. Em contraste, a explicação do relatório apresentou maior dispersão, incluindo 1 avaliação 'Péssimo' e 4 'Médio'. Esse resultado sugere que, embora a extração de evidências seja eficaz, a forma como os resultados são comunicados precisa de aprimoramento na comunicação dos resultados. A utilização de uma linguagem mais clara, a inclusão de exemplos práticos e a organização mais intuitiva das informações podem contribuir para um melhor entendimento do relatório.

Conclusões

Este trabalho usou LLMs e técnicas de RAG para automatizar o levantamento de evidências técnicas em projetos de software, focando na validação de requisitos em contratos. A pesquisa mostrou que essa abordagem pode melhorar a análise de contratos e código-fonte, tornando o processo mais eficiente.

Os resultados preliminares indicam que a combinação de LLMs e RAG pode aprimorar a verificação de requisitos, embora limitações como o tamanho amostral e a qualidade da documentação demandem mais estudos.

Direções futuras incluem a criação de uma ferramenta automatizada, a inclusão de novos artefatos como fontes de evidências e a análise de diferentes LLMs para melhorar a precisão. Além disso, sugere-se integrar Inteligência Artificial (IA) para otimizar a gestão de projetos e expandir a aplicação para outros tipos de contratos de software.

Estudos Futuros

- Automação e Ferramenta Completa:** Desenvolver uma *ferramenta* mais automatizada para gerar relatórios de verificação de requisitos, integrando-a com sistemas de gerenciamento de projetos e bancos de dados.
- Exploração de Artefatos Adicionais:** Incluir *e-mails*, *commits*, *manuals de uso* e *gravações de reuniões* como fontes de evidências técnicas, ampliando a base de dados utilizada.
- Análise de Diferentes LLMs:** Comparar o desempenho de diversos *LLMs* para determinar qual oferece melhor *precisão e eficiência* na extração de evidências técnicas.
- Pesquisa de Campo:** Realizar *pesquisas com stakeholders* do setor para avaliar a aceitação e a utilidade da ferramenta, ajustando-a conforme o feedback recebido.
- Integração com IA para Apoio à Decisão:** Investigar a integração da ferramenta com *plataformas de IA* para otimizar a gestão de *orçamento, prazo e riscos* em projetos de software.
- Expansão para Outros Tipos de Contratos:** Expandir a pesquisa para outros *tipos de contratos* de software, como os de manutenção e customização, adaptando a ferramenta para diferentes contextos.

Referências Principais

- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.
- Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Agradecimentos

A **Phac Soluções Tecnológicas** pela cessão de dados e avaliação do projeto, e ao **IME** (Instituto de Matemática e Estatística) pelo suporte na disponibilização de recursos.