UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL-REI

Julia Luiza Ferreira Santos

Criação de Grafos de Conhecimento Baseados na Extração de Entidades e Relações em Textos de Imunologia

São João del-Rei

2025

UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL-REI

Julia Luiza Ferreira Santos

Criação de Grafos de Conhecimento Baseados na Extração de Entidades e Relações em Textos de Imunologia

Monografia apresentada como requisito da disciplina de Projeto Orientado em Computação I do Curso de Bacharelado em Ciência da Computação da UFSJ.

Orientador: Alexandre Bittencourt Pigozzo

Universidade Federal de São João del-Rei – UFSJ

Bacharelado em Ciência da Computação

São João del-Rei

2025

Julia Luiza Ferreira Santos

Criação de Grafos de Conhecimento Baseados na Extração de Entidades e Relações em Textos de Imunologia

Monografia apresentada como requisito da disciplina de Projeto Orientado em Computação I do Curso de Bacharelado em Ciência da Computação da UFSJ.

Trabalho aprovado. São João del-Rei, 25 de novembro de 2025:

Alexandre Bittencourt Pigozzo
Orientador

Professor
Convidado 1

Professor
Convidado 2

São João del-Rei
2025

Não por força nem por poder, mas pelo meu Espírito,
diz o Senhor dos Exércitos
Zacarias 4:6

Agradecimentos

Agradeço...

"Não vos amoldeis às estruturas deste mundo,
mas transformai-vos pela renovação da mente,
a fim de distinguir qual é a vontade de Deus:
o que é bom, o que Lhe é agradável, o que é perfeito.
(Bíblia Sagrada, Romanos 12, 2)

Resumo

Resumo

Palavras-chaves: latex. abntex. editoração de texto.

Abstract

This is the english abstract.

Key-words: latex. abntex. text editoration.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

Fig.	Area of the i^{th} component
456	Isto é um número
123	Isto é outro número
lauro cesar	este é o meu nome

Lista de símbolos

Γ	Letra grega Gama
Λ	Lambda
ζ	Letra grega minúscula zeta
$\in$	Pertence

Sumário

1 Introdução

Os grafos de conhecimento apresentam uma ampla gama de aplicações, como sistemas de recomendação, detecção de fake news e sistemas de diálogo (1). Um grafo é representado na forma de triplas (Entidade1 – Relação – Entidade2), onde os nós representam entidades ou conceitos e as arestas representam os relacionamentos entre eles. Ele exibe o conhecimento estruturado por meio de uma interface gráfica, permitindo que os usuários visualizem e descubram, de forma mais intuitiva, as relações entre as informações (2). Aplicações bem-sucedidas de grafos de conhecimento já podem ser observadas na área médica. Grafos médicos integram dados provenientes de múltiplas fontes e auxiliam tarefas como busca de informações, diagnóstico, tratamento e prognóstico (3). Por analogia, um grafo voltado à imunologia pode não apenas estruturar o conhecimento básico da área, mas também apoiar análises sobre interações imunológicas e interpretação de processos complexos.

A construção de grafos de conhecimento, porém, enfrenta algumas limitações. Em áreas altamente especializadas, pode ser necessária a participação de um grande número de especialistas para definir corretamente as entidades e relações relevantes. Ainda que os avanços em PLN tenham reduzido essa dependência, grafos construídos a partir de ferramentas de PLN geralmente exigem grandes quantidades de dados anotados para treinar modelos com boa precisão.

Nesse sentido, os LLMs contribuem significativamente para superar esse desafio, uma vez que permitem extrair Entidades Nomeadas e relações sem demandar grande esforço humano ou dados adicionais de treinamento . Isso ocorre porque técnicas como o fine-tuning podem aprimorar o desempenho dos modelos em tarefas textuais (4). Estudos recentes têm demonstrado que a adição de apenas uma única demonstração de tarefa ao prompt aumenta de forma expressiva a qualidade da extração de triplas, e que o método RAG (*Retrieval Augmented Generation*) contribui para elevar a precisão dessas extrações (5).

Com base nessas observações, este trabalho propõe um sistema para a criação de grafos de conhecimento a partir de artigos científicos da área da imunologia. O objetivo é estruturar o conhecimento presente nesses materiais de modo a evidenciar relações e interações biológicas entre entidades imunológicas. Para isso, utilizou-se a API gemini-2.5-flash para extrair quadruplas de conhecimento — compostas por entidade1, relação, entidade2 e uma condição que especifica quando a relação ocorre — no nível de parágrafos. A extração é guiada por um prompt contendo alguns exemplos, os quais são recuperados pelo método RAG a partir de uma pequena base de dados construída com anotações

manuais de exemplos da tarefa fornecida ao modelo.

2 Pesquisa

2.1 Trabalhos Relacionados

Alguns trabalhos serviram de inspiração ou referência para o desenvolvimento da metodologia deste Trabalho de Conclusão de Curso (TCC). O primeiro a ser destacado é o TCC do Rodrigues (2025) (6), que aborda o uso de modelos generativos (LLaMA 3.2:3B e o DeepSeek-R1:8B) para extração de entidades nomeadas de textos da área de Imunologia e de textos sobre modelagem computacional do sistema imune.

Rodrigues (2025) propôs um pipeline, cuja primeira etapa consiste na segmentação de um arquivo PDF de entrada para obtenção de trechos usados como tarefas para o modelo. A tarefa é expressa em um prompt construído de modo a restringir o uso de termos a uma lista pré-definida de entidades nomeadas. Para melhorar o desempenho do modelo, o sistema inclui no prompt uma contextualização do assunto tratado no texto de entrada do LLM. Para isso, é utilizado o método RAG, que recupera do banco de dados o contexto cuja incorporação vetorial mais se assemelha à do texto da tarefa. O banco de dados escolhido foi o MongoDB, devido à sua capacidade de busca vetorial otimizada (Vector Search). Esse trabalho, no entanto, não empreende a tarefa de extração de relações entre as entidades obtidas. Além disso, não explora a capacidade dos LLMs generativos de criar novos tokens — recurso que poderia ser vantajoso em casos em que as entidades nomeadas e as relações são apresentadas de forma implícita no texto.

O trabalho de Polat et al. (2024)(5), em contrapartida, testa diferentes métodos de engenharia de prompt nos modelos GPT-4, Mistral 7B e Llama 3. Em vez de estabelecer previamente um escopo fixo de possíveis relações, os autores permitem que o modelo atue extraindo relações para além de correspondências exatas, o que resultou em inferências mais refinadas a partir dos textos. Nesse trabalho, o RAG foi utilizado para selecionar exemplos de tarefas semelhantes à que o modelo deveria executar.

Outro diferencial do artigo foi a sua proposta de um método de avaliação baseado em ontologias. O objetivo é reduzir a dependência da avaliação humana, aproveitando a semântica dos dados e as restrições de propriedades presentes no Wikidata(7) para automatizar o processo de validação das triplas extraídas. No entanto, a ontologia do Wikidata é dinâmica, e edições realizadas por usuários podem afetar a avaliação. O conteúdo existente na base também é propenso a erros, não podendo ser tomado como verdade absoluta. Além disso, a técnica usada no trabalho para vincular uma entidade obtida a um termo existente no Wikidata também apresentou limitações, uma vez que algumas correspondências geradas não foram corretas(5). Apesar dessas limitações, a proposta de avaliação

se mostrou promissora e com potencial de aperfeiçoamento.

Já o trabalho de Xu et al. (2024) (3) utiliza o modelo ChatGPT 3.5-Turbo-16k API nas tarefas de Reconhecimento de Entidades Nomeadas (NER) e Extração de Relações (RE). Diferentemente dos estudos anteriormente citados, o método proposto também realiza a desambiguação de entidades, removendo ambiguidades e sinônimos por meio da métrica de similaridade de Jaccard. A partir das entidades e relações extraídas, os autores constroem um grafo de conhecimento sobre insuficiência cardíaca, o qual pode auxiliar na tomada de decisões clínicas e facilitar a visualização de informações relevantes para diagnóstico, tratamento e prognóstico.

De modo geral, os estudos analisados trazem contribuições importantes para o uso de modelos de linguagem de grande porte em tarefas de extração de informação, especialmente por meio de suas metodologias de engenharia de prompts e do uso do RAG. Entre eles, o trabalho de Rodrigues (2025) se destaca por aplicar modelos generativos à extração de entidades nomeadas em textos biomédicos, embora não aborde a extração de relações. Essa limitação marca o primeiro ponto de partida para o desenvolvimento deste TCC. Além disso, nota-se que nenhum dos estudos explora a identificação das condições em que as relações ocorrem, o que configura o segundo aspecto inovador desta proposta. Ambos os pontos são detalhados na seção de Metodologia.

2.2 Metodologia

Com o objetivo de estruturar o conhecimento presente em textos da área de imunologia, esta metodologia utiliza técnicas de Reconhecimento de Entidades Nomeadas (NER) e Extração de Relações (RE), apoiadas em um modelo baseado em engenharia de prompts inspirado em Polat et al. (2024). Para a obtenção de exemplos relevantes ao modelo, empregou-se o método RAG (Retrieval-Augmented Generation), utilizando o Vector Search do MongoDB como mecanismo de busca semântica. Além disso, com o intuito de facilitar a compreensão das interações biológicas, o sistema desenvolvido gera um grafo de conhecimento que representa visualmente as relações identificadas entre as entidades.

A metodologia foi organizada em cinco etapas principais: (1) preparação da base de dados; (2) leitura e segmentação de artigos; (3) aplicação do RAG; (4) construção do prompt; e (5) desenvolvimento da interface. Cada uma dessas etapas será detalhada nas subseções a seguir.

2.3 Base de dados

O banco de dados utilizado neste TCC foi o MongoDB. A escolha se deve à disponibilidade do recurso de pesquisa vetorial (Vector Search). Diferentemente da busca

tradicional, que localiza correspondências exatas de texto, a pesquisa vetorial identifica vetores próximos à consulta em um espaço multidimensional (8). Para isso, as incorporações vetoriais do texto de entrada são comparadas com as demais armazenadas no banco de dados. Diferentes tipos de métricas podem ser utilizadas para se calcular a proximidade vetorial entre as incorporações de texto. Neste trabalho, utilizou-se a similaridade do cosseno.

A base de dados é utilizada para armazenar exemplos de tarefas semelhantes às que o modelo deve executar. Assim, cada registro contém um trecho de um artigo da área de imunologia, sua incorporação vetorial (embedding) e uma lista de quádruplas. Cada quádrupla contém entidades nomeadas identificadas no texto, às relações entre elas e as condições em que essas relações ocorrem. O banco de dados foi anotado manualmente, a partir da seleção de trechos, análises e extração de informações de seções de textos da área de imunologia (9, 10).

As incorporações foram geradas utilizando a ferramenta Sentence Transformers (SBERT), um módulo Python voltado para acessar, usar e treinar modelos de embedding. Essa ferramenta permite calcular representações vetoriais densas para sentenças e parágrafos (11). O modelo adotado foi o all-MiniLM-L6-v2, que mapeia textos para um espaço vetorial de 384 dimensões, sendo adequado para tarefas como agrupamento e busca semântica (12).

A seguir, apresenta-se um exemplo de quádruplas extraídas a partir de um trecho de texto da área de imunologia(13):

"The CD8-TL recognizes intracytoplasmic antigens presented by MHC molecules class I, which are expressed by practically all nucleated cells. ... At a subsequent contact, the CD8-TL can eliminate by cytotoxicity any cell that presents this specific antigen. The CD8-TL induce apoptosis in the target cell through the action of perforins and granzymes and can also lead to apoptosis through the expression of the Fas L receptor. ... The TL ... can recognize antigens even in the absence of presentation by the MHC molecule."


```
[
  {
    "source": "CD8-TL",
    "relation": "recognizes",
    "destiny": "intracytoplasmic antigens",
    "condition": "presented by MHC class I molecules"
  },
  {
    "source": "MHC class I molecules",
    "relation": "expressed_by",
```

```
    "destiny": "nucleated cells",
    "condition": ""
  },
  {
    "source": " TL",
    "relation": "recognizes",
    "destiny": "antigens",
    "condition": "even in the absence of presentation by the MHC molecule"
  },
  {
    "source": "CD8-TL",
    "relation": "eliminates",
    "destiny": "target cell",
    "condition": "presence of specific antigen and previous contact"
  },
  {
    "source": "CD8-TL",
    "relation": "induces_apoptosis_in",
    "destiny": "target cell",
    "condition": "through perforins and granzymes action OR
    Fas L receptor interaction"
  }
]
```


2.4 Segmentação de artigos

Muitos estudos se concentram em extrair relações a partir de sentenças individuais. Entretanto, muitas aplicações do mundo real exigem que os sistemas de RE identifiquem entidades e relações que ultrapassam os limites das frases. A linha de pesquisa em RE voltada à extração de relações em nível de documento apresenta a vantagem de capturar de forma mais completa as informações relacionais distribuídas ao longo de um texto (14).

Considerando a relevância das análises que envolvem múltiplas sentenças — e a capacidade dos LLMs de lidar com esse tipo de processamento — neste trabalho o processo de RE e NER foi conduzido em nível de parágrafo. Para isso, utilizou-se a biblioteca *pdfplumber* para leitura do conteúdo dos artigos e posterior segmentação em parágrafos. A escolha de trabalhar com parágrafos foi motivada pela necessidade de preservar o contexto e as relações implícitas entre as entidades.

2.5 RAG

O método RAG consiste em fornecer ao modelo exemplos relevantes selecionados a partir de similaridade semântica. Sua estrutura é composta por quatro elementos principais: uma base de conhecimento, um codificador de embeddings, um mecanismo de recuperação e um gerador (5). A base de conhecimento e o processo de codificação dos textos em embeddings — utilizados para identificar o exemplo mais semelhante por meio da busca vetorial — foram apresentados na Seção 4.1.

De forma resumida, o trecho extraído do artigo na etapa de segmentação — que será utilizado como tarefa de RE e NER — é inicialmente convertido em embeddings. Em seguida, o mecanismo de busca vetorial identifica, dentro da base de conhecimento, o exemplo mais similar a esse trecho. O exemplo recuperado é utilizado na construção do prompt que descreve a tarefa a ser executada pelo modelo. A etapa de engenharia de prompts é detalhada na subseção seguinte. A partir desse prompt, o modelo interpreta as instruções e realiza a análise solicitada, gerando como saída a extração das entidades nomeadas presentes no trecho, as relações entre elas e as condições necessárias para que essas relações ocorram. O modelo escolhido para esse fim foi o *gemini-2.5-flash*.

2.6 Prompt

A técnica de Engenharia de Prompt utilizada foi o *few-shot prompting*, que consiste em apresentar ao modelo alguns exemplos da tarefa a ser executada — no caso deste trabalho, três exemplos. O prompt foi estruturado para cumprir as seguintes funções:

- definir o papel do modelo;
- indicar os elementos que devem ser ignorados no texto;
- contextualizar a área de atuação (imunologia);
- descrever claramente a tarefa (extração de quádruplas);
- fornecer exemplos da tarefa (obtidos via método RAG);
- apresentar o trecho de texto sobre o qual o modelo deve atuar.

O papel atribuído ao modelo é o de **analisador especializado em extrair relações biológicas**. Seu objetivo é identificar apenas interações biologicamente relevantes, representadas como relações entre células, moléculas ou processos imunológicos. Para evitar ruídos, o modelo foi instruído a ignorar relações que não pertencem ao domínio biológico, como relações sociais, históricas ou culturais.

A saída esperada foi definida no formato `JSON`, com cada quádrupla composta pelos seguintes campos:

- **source**: entidade de origem da relação;
- **relation**: verbo que expressa a interação;
- **destiny**: entidade de destino da relação;
- **condition**: condição necessária para que a relação ocorra, inferida pelo modelo a partir do contexto.

2.7 Interface

Para a geração do grafo de conhecimento a partir dos `JSONs` produzidos pelo modelo, foi utilizada a biblioteca *Pyvis*, responsável por criar grafos interativos em formato `HTML`. Já para o desenvolvimento da interface gráfica com o usuário, optou-se pela biblioteca *Streamlit*, que permite a construção rápida e intuitiva de aplicações voltadas à visualização e interação com os resultados gerados pelo sistema.

Referências

1 QUIROZ, R. N.; CAMACHO, J. V.; PEñATA, E. Z.; LEMUS, Y. B.; LóPEZ-FERNáNDEZ, C.; ESCORCIA, L. G.; FERNáNDEZ-PONCE, C.; COBOS, M. R.; MORENO, J. F.; FIORILLO-MORENO, O.; QUIROZ, E. N. Multiscale information processing in the immune system. *Frontiers in Immunology*, v. 16, 2025. Disponível em: <https://www.frontiersin.org/journals/immunology/articles/10.3389/fimmu.2025.1563992>. Citado na página 13.

2 HAO, X.; JI, Z.; LI, X.; YIN, L.; LIU, L.; SUN, M.; LIU, Q.; YANG, R. Construction and application of a knowledge graph. *Remote Sensing*, v. 13, n. 13, p. 2511, 2021. Disponível em: <https://doi.org/10.3390/rs13132511>. Citado na página 13.

3 XU, T.; GU, Y.; XUE, M.; GU, R.; LI, B.; GU, X. Knowledge graph construction for heart failure using large language models with prompt engineering. *Frontiers in Computational Neuroscience*, v. 18, p. 1389475, 2024. Citado 2 vezes nas páginas 13 e 16.

4 YE, Y. Construction and application of materials knowledge graph in multidisciplinary materials science via large language model. *Advances in Neural Information Processing Systems*, v. 37, 2024. Citado na página 13.

5 POLAT, F.; TIDDI, I.; GROTH, P. Testing prompt engineering methods for knowledge extraction from text. *Semantic Web*, 2024. Disponível em: <https://www.researchgate.net/publication/384205838_Testing_prompt_engineering_methods_for_knowledge_extraction_from_text>. Citado 3 vezes nas páginas 13, 15 e 19.

6 RODRIGUES, G. M. *Desenvolvimento de um prompt para o reconhecimento de entidades nomeadas usando modelos de inteligência artificial generativa*. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) — Universidade Federal de São João del-Rei (UFSJ), São João del-Rei, 2025. Citado na página 15.

7 VRANDECIC, D.; KRÖTZSCH, M. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, v. 57, n. 10, p. 78–85, 2014. Citado na página 15.

8 MONGODB. *MongoDB Vector Search Overview*. 2025. <https://www.mongodb.com/docs/atlas/atlas-vector-search/vector-search-overview/>. Accessed: 2025-02-17. Citado na página 17.

9 SOMPAYRAC, L. *How the Immune System Works*. 6. ed. [S.l.]: Wiley Blackwell, 2019. 145 p. Soft cover; includes index. Citado na página 17.

10 JúNIOR, D. M.; ARAúJO, J. A.; CATELAN, T. T.; SOUZA, A. W.; CRUVINEL, W. M.; ANDRADE, L. E.; SILVA, N. P. Immune system - part ii: basis of the immunological response mediated by t and b lymphocytes. *Revista Brasileira de Reumatologia*, v. 50, n. 5, p. 552–580, Sep-Oct 2010. Citado na página 17.

11 SENTENCETRANSFORMERS Documentation. 2025. Created with Sphinx using a theme provided by Read the Docs. Disponível em: <https://sbert.net/>. Citado na página 17.

12 SENTENCE-TRANSFORMERS. *all-MiniLM-L6-v2*. 2021. <https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2>. Accessed: 2025-02-17. Citado na página 17.

13 DE, S.; SANYAL, D. K.; MUKHERJEE, I. Fine-tuned encoder models with data augmentation beat chatgpt in agricultural named entity recognition and relation extraction. *Expert Systems with Applications*, v. 277, p. 127126, 2025. ISSN 0957-4174. Disponível em: <https://doi.org/10.1016/j.eswa.2025.127126>. Citado na página 17.

14 ZHAO, X.; DENG, Y.; YANG, M.; WANG, L.; ZHANG, R.; CHENG, H.; LAM, W.; SHEN, Y.; XU, R. A comprehensive survey on relation extraction: Recent advances and new frontiers. *ACM Computing Surveys*, v. 56, n. 11, p. 293, 2024. Disponível em: <https://doi.org/10.1145/3674501>. Citado na página 18.

Apêndices

APÊNDICE A – Titulo

Anexos

ANEXO A – Título.