Grafo[1] de Conhecimentos dos Grupos PET: IA na coleta automatizada de informações e suas implicações

NUNES, A.D.B.[1]; MARTINS, A.[1]; NUNES, R. F.[1]; KAWAZOE, S. H. L.[1]; MELAYE, L.[2]; TACLA, C.A.[3]

[1]Grupo PET- Engenharia de Computação, UTFPR, Campus Curitiba; [2]Iniciação Científica, UTFPR, Campus Curitiba; [3]Tutor(a) do Grupo PET- Engenharia de Computação, UTFPR, Campus Curitiba
E-mail: arthurbemnunes@alunos.utfpr.edu.br, petecoutfpr@gmail.com

RESUMO: O Programa de Educação Tutorial sofre com a descentralização de informações, o que diminui a visibilidade de suas atividades. O objetivo deste trabalho é desenvolver um sistema de informação que utiliza inteligência artificial (modelo de linguagem de grande escala) e banco de dados em grafos para coletar, organizar e facilitar o acesso a essas informações. A metodologia envolveu a definição de uma ontologia e o uso de agentes inteligentes para realizar *web scraping* nos sites dos grupos PET, extraindo dados de suas atividades. As informações coletadas são armazenadas em um banco de dados de grafos. Um *front-end* permite a visualização interativa dos dados. Como resultado, obteve-se um protótipo funcional que executa o fluxo completo de extração, armazenamento e consulta, validando a arquitetura proposta. A abordagem realizada demonstrou-se adequada para representar e consultar as atividades de diferentes grupos. Conclui-se que o sistema valida a ideia de facilitar o acesso às informações dos grupos PET. O artigo discute as implicações de utilização de IA na coleta de dados públicos dos grupos PET quanto à correção, tratamento dos dados com IA e o fornecimento implícito deles aos modelos de linguagem.

Palavras-chave: Grafos de Conhecimento, Agentes Inteligentes, Modelos de Linguagem, Atividades PET

Knowledge Graph of PET Groups: AI in automated information gathering and its implications

ABSTRACT: The Tutorial Education Program suffers from information decentralization, which reduces the visibility of its activities. The objective of this work is to develop an information system that uses artificial intelligence (a large language model) and a graph database to collect, organize, and facilitate access to this distributed information. The methodology involved defining an ontology and using intelligent agents to perform web

[1]Área do conhecimento: Ciências Exatas e da Terra (1.03.00.00-7); Transformação Digital; ODS: Educação de Qualidade; Indústria, Inovação e Infraestrutura

scraping on PET group websites, extracting data about their activities. The collected information is stored in a graph database. A front-end interface enables interactive data visualization. As a result, a functional prototype was obtained that performs the complete flow of extraction, transformation, storage, and querying, thus validating the proposed architecture. The approach proved adequate for representing and querying the activities of different groups. It is concluded that the system validates the idea of facilitating access to PET group information. The paper discusses the implications of using AI for collecting public data from PET groups, regarding data accuracy, AI-based data processing, and the implicit provision of such data to language models.

Keywords: Knowledge Graphs; Intelligent Agents; Large Language Model; PET Activities

Introdução

O PET é um dos projetos de educação superior do MEC (CENAPET, 2025). No entanto, há uma grande dificuldade em encontrar informações das atividades de cada grupo e do programa, diminuindo a visibilidade de uma iniciativa tão importante para a educação brasileira. Deste modo, surgiu a ideia de utilizar ferramentas computacionais de inteligência artificial e banco de dados em grafos para a criação de um sistema que coleta informações dos sites dos grupos PET espalhados pelo Brasil a fim de facilitar o acesso e melhorar a divulgação das atividades dos grupos e do programa. Porém, levantou-se uma questão dos direitos de uso das informações publicadas pelos grupos PET bem como sua cessão implícita aos modelos de linguagem de grande escala (LLMs, na sigla em inglês), como GPT, Llama e outros similares.

Um LLM funciona, em essência, como um preditor da próxima palavra. Dado um *prompt*, uma sequência de palavras, o modelo calcula, com base em tudo que aprendeu durante o treinamento utilizando-se de um corpus de textos, qual é a palavra mais provável de ocorrer em seguida. A LLM é uma técnica de aprendizado de máquina da Inteligência Artificial e foi utilizada com objetivo de automatizar a coleta das informações de atividades dos grupos PET identificando partes relevantes de descrições de atividades encontradas em páginas web.

Um banco de dados em grafos armazena estruturas matemáticas que representam relações por meio de triplas <*sujeito, predicado, objeto*> (ALI et al., 2022), por exemplo, <*'PETECO', 'realiza oficina', 'pensamento computacional'*>. É uma ferramenta adequada ao

projeto porque permite representar informações sem que as relações (predicados) sejam todas definidas a priori. Os predicados podem ser acrescentados à medida que as informações são coletadas. É possível navegar nas informações, filtrá-las para responder perguntas (ex. quais atividades o grupo PETECO e o grupo PETCoCe realizam em conjunto) e, além disso, é possível identificar padrões de atividades realizadas por um conjunto de grupos como as atividades mais comumente realizadas pelos grupos PET de uma área específica.

Os predicados e os tipos de sujeito e objeto são definidos por meio de uma ontologia. Uma ontologia, como artefato computacional, é uma representação do vocabulário comum utilizado por uma ou mais aplicações de forma a permitir a comunicação entre elas. A ontologia influencia a estrutura da representação das informações e a forma do armazenamento dos dados. Escolheu-se representar as ontologias com a linguagem OWL[2] na forma de triplas <*sujeito, predicado, objeto*>. Logo, há congruência entre o banco de dados em grafo e a ontologia.

As tecnologias mencionadas foram escolhidas porque permitem trabalhar com informações incompletas ou faltantes. Também, é possível aprender padrões de atividades segundo critérios de interesse do usuário. Em seguida, aborda-se o método para diminuir o problema de se conseguir informações sobre os grupos PET no Brasil.

Método

Dividiu-se a equipe em 2 grupos, o primeiro pelo banco de dados, e o outro pelo *web scraping*, que permite a coleta automática de informações dos websites dos grupos. Em relação ao banco de dados, há duas opções comumente utilizadas: GraphDB e Neo4j. O GraphDB é de código aberto e tem mais funcionalidades para trabalhar com ontologias. Porém, optou-se pelo Neo4j devido à sua maior popularidade e facilidade de implementação (BORG; PERSSON, 2017). Em relação aos agentes inteligentes, escolheu-se a ferramenta CrewAI (WINLAND et al., [s.d.]), principalmente devido à facilidade de integração com os LLMs. O LLM escolhido foi o Llama porque pode ser executado localmente, garantindo um maior controle nos testes de

[2] OWL: Web Ontology Language - https://www.w3.org/OWL

funcionamento do site (LI et al., 2024).

Desenvolveu-se uma aplicação web na linguagem Python no *back-end*[3] utilizando a fastAPI[4]. Para o *front-end*[5] utilizou-se a linguagem Javascript associada ao framework Vue-Js, que facilita a criação de interfaces dinâmicas e responsivas.

Adotou-se a conteinerização do projeto para facilitar a instalação e execução do aplicativo nos computadores de desenvolvedores e usuários. Optou-se pelo uso do Docker, com o objetivo de encapsular os serviços da aplicação, especialmente o *back-end*, em contêineres independentes. Essa abordagem facilita a replicação do ambiente de desenvolvimento no ambiente de produção, reduz problemas de compatibilidade e permite que a aplicação seja executada localmente de forma padronizada.

Em paralelo às definições e realizações técnicas, modelou-se as informações (tipos de sujeito/objeto e predicados) por meio de uma ontologia. De forma não exaustiva, foram modelados os tipos de grupo conexão dos saberes e de curso por meio de uma relação de especialização (predicado *type*). A afiliação dos grupos a um campus universitário foi definida pelo predicado *estahNoCampus*. Assim, a tripla <PETECO, *type,* GrupoDeCurso> é interpretada como o grupo PETECO (sujeito) é do tipo (predicado) grupo de curso (objeto).

Em seguida, foram definidas as funcionalidades da aplicação, responsáveis por integrar e dar dinamismo ao sistema. A aquisição e extração de dados realiza a coleta de informações dos *websites* dos grupos PET. Um *script* em Python roda os agentes inteligentes do crewAI, que recebem um *prompt* para procurar os *links* para os *websites* na página da CENAPET e os armazenam em uma lista. Essa lista é repassada a um segundo *script*, responsável pelo *scraping*. É na coleta de informações das páginas web que o modelo LLM é utilizado como um interpretador de conteúdos capaz de localizar passagens de interesse. O agente acessa uma página web por vez e busca por informações de sujeitos, predicados e objetos que estão de acordo com a ontologia com o auxílio da LLM, como: grupo, afiliação, título da atividade, data de publicação, resumo, participantes, área de estudo e área acadêmica do estudo. A saída dessa

[3] Parte da aplicação que executada no servidor que realiza as operações solicitadas pelo usuário
[4] API: *Application Programming Interface* - Interface de Programação de Aplicações
[5] responsável pela interação direta com o usuário

tarefa é um arquivo dos dados coletados em formato JSON (propriedade-valor, ex. título: Oficina de Pensamento Computacional, grupo: PETECO), que é passado ao *backend*. O *backend* transforma este arquivo em triplas de acordo com a ontologia e os armazena no banco de dados Neo4J. Com os dados armazenados, o *backend* pode fazer a ponte do Neo4J e o *front-end* para fins de consulta das informações.

No *front-end*, quando o usuário clica em um dos "nós" (o sujeito ou o objeto da tripla), o aplicativo deixa visível apenas as informações que possuem relação com o nó selecionado. Por exemplo, ao selecionar o nó do grupo PETECO, aparecem todos os nós ligados a ele: <PETECO, realizaAtividade, pensamento computacional>, <PETECO, estahNoCampus, UTFPR-Curitiba> etc. A partir daí é possível clicar em outros nós e navegar no grafo.

Resultados e Discussão

O principal resultado deste projeto foi a criação de um protótipo funcional do aplicativo de coleta e consulta de informações de grupos PET. O protótipo criado demonstrou ser capaz de executar todo o processo de extração e armazenamento de dados, validando a arquitetura proposta.

No *front-end,* um usuário pode clicar em um nó que representa um grupo PET e ver todas as atividades a ela relacionadas pelo seu tipo (ensino, pesquisa, extensão ou mista). A abordagem baseada em LLM para extração de dados apresentou diversos desafios, como a distinção entre sites, onde seus designs e informações são apresentados de maneiras distintas, ademais, a falta de atualização de informações em muitos sites também dificultou a extração de informações atuais, assim, indicando a necessidade de melhorar os *prompts* utilizados, utilizar um LLM com maior capacidade de interpretação de textos aumentando o risco de alucinação - quando o modelo gera textos sem respaldo na realidade, ou colocar etapas de validação humana para os dados coletados, com objetivo de melhorar a precisão da IA na interpretação das informações e excluir aquelas que não condizem com a realidade.

O conjunto de tecnologias adotadas não apenas armazena os dados, mas também representa e explora as relações semânticas entre eles, possibilitando consultas que seriam

difíceis de realizar em bancos de dados relacionais tradicionais. Por exemplo, ao pesquisar atividades de grupos da área de Ciências Exatas no tema "robótica", um mecanismo de busca convencional retornaria apenas uma lista de atividades classificadas nesse tema. Já em uma abordagem com ontologia e orientada a grafos, o usuário pode filtrar e explorar conexões como atividades realizadas em universidades de uma mesma região ou a um tema relacionado à robótica (ex. domótica) que são especializações de um conceito mais geral (automação). Essa possibilidade permite aos usuários identificarem relações que não seriam facilmente visíveis em sistemas relacionais. Além disso, é possível estabelecer um vocabulário comum e ampliá-lo com o uso de LLM a medida que descobrem novos predicados.

Conclusões

Diversas funcionalidades precisam de refinamento, como melhorar a busca de *links* de outros aplicativos, além de *links* para páginas web. É necessário melhorar o processo de implantação do aplicativo que deve ser executado em nuvem para uma verdadeira distribuição. Isso envolve a definição das permissões associadas a papeis de usuários, assim como programar códigos de segurança web.

Outro desafio importante é o uso de informações públicas por sistemas de inteligência artificial, no nosso caso particular, na etapa de *web scrapping*. Mesmo que as informações das atividades dos grupos estejam disponíveis na Web, isso não significa que podem ser usadas livremente, nem que são sempre confiáveis. Se o aplicativo coletar informações desatualizadas ou interpretá-las de forma equivocada devido ao risco de alucinação dos LLMs, isso pode prejudicar os grupos PET, divulgando conteúdos que não são verdadeiros. Além disso, ao submeter um texto a um LLM, pode-se estar, implicitamente, contribuindo com dados que poderão ser utilizados por outras pessoas e/ou organizações que fazem uso da LLM para fins desconhecidos. Para amenizar esse tipo de problema, foi criado um servidor local, que não depende de servidores externos e tem acesso controlado. Ainda assim, ao expandir o projeto e incluir mais dados, é preciso ter cuidado para não usar informações que exigem autorização, respeitando os direitos de quem as criou ou publicou. Finalmente, a abordagem proposta pode

ser utilizada em diversos tipos de projeto que tenham informações distribuídas, sejam projetos de pesquisa, extensão ou educacionais.

Agradecimentos

Os autores agradecem ao Programa de Educação Tutorial (PET) do Ministério da Educação (MEC) pelo fomento e apoio financeiro que possibilitam a realização deste trabalho. Agradecemos também à Executiva Nacional dos Grupos PET (CENAPET) por seu papel fundamental na articulação e fortalecimento do programa em âmbito nacional e à Universidade Tecnológica Federal do Paraná (UTFPR) pelo suporte institucional e infraestrutura indispensáveis para a execução de nossas atividades.

Referências

EXECUTIVA NACIONAL DOS GRUPOS PET (CENAPET). **CENAPET**. Brasil, 2025. Disponível em: https://cenapet.org.br/. Acesso em: 4 jun. 2025.

ALI, W.; SALEEM, M.; YAO, B.; HOGAN, A.; NGOMO, AC. N. A survey of RDF stores & SPARQL engines for querying knowledge graphs. **The VLDB Journal**, v. 31, n. 3, p. 1–26, mai 2022. DOI: https://doi.org/10.1007/s00778-021-00711-3.

BORG, D.; PERSSON, D. **A performance comparison between graph databases**. 2017. Trabalho de Conclusão de Curso (Bacharelado) – Kristianstad University, Kristianstad, 2017. Disponível em: <https://researchportal.hkr.se/sv/studentTheses/a-performance-comparison-between-graph-databases/>. Acesso em: 8 nov 2025.

WINLAND, V.; SYED, M; GUTOWSKA, A. *O que é o crewAI?* Disponível em: <https://www.ibm.com/br-pt/think/topics/crew-ai>. Acesso em: 8 nov 2025.

LI, X.; WANG, S.; ZENG, S.; WU, Y.; YANG, Y. A survey on LLM-based multi-agent systems: workflow, infrastructure, and challenges. **Vicinagearth**, v. 1, n. 1, p. 9, out 2024. DOI: https://doi.org/10.1007/s44336-024-00009-2.