Rumo a uma proposta de assistente estudantil na web descentralizada

Deivid Alves de Carvalho[1], **Fernando Willian Cruz**[1]

[1]Faculdade de Ciências e Tecnologias em Engenharias – Universidade de Brasília (UnB) – Brasília – DF – Brasil

deivid_506_02@hotmail.com, fwcruz@unb.br

Abstract. *Na literatura há vários relatos de uso de agentes de IA no contexto educacional para proporcionar uma modernização de serviços. Neste artigo propõe-se o desenvolvimento de um assistente estudantil baseado em agentes inteligentes para operação em ambiente web descentralizada. O agente proposto faz uso de modelos de linguagem natural executados localmente e tem suas funcionalidades ampliadas por meio da integração dinâmica do agente com ferramentas externas disponíveis no contexto universitário por meio de interface padronizada. Por outro lado, a execução local garante o controle total sobre dados sensíveis dos estudantes, assegurando conformidade com a LGPD e proporcionando autonomia tecnológica às instituições educacionais. Testes experimentais utilizando a plataforma Hugging Face demonstraram capacidade satisfatória do agente em executar raciocínio estruturado e realizar chamadas de ferramentais externos ao modelo para prover respostas às consultas dos estudantes. Os resultados preliminares indicam viabilidade técnica da proposta, oferecendo uma alternativa às soluções centralizadas, baseadas em APIs cloud, para assistência acadêmica personalizada.*

1. Introdução

Na literatura ligada à IA aplicada à educação, tem havido cada vez mais relatos em todo o mundo sobre a exploração de ambientes imersivos para o aprendizado [Lin et al. 2022] [Wang et al. 2022] [Han et al. 2023], principalmente para o ensino superior [Ruwodo et al. 2022], muitos deles na perspectiva de fornecer serviços tradicionais e inovadores no campus por meio de um ambiente metaverso descentralizado.

Uma das formas de se viabilizar a oferta de serviços neste contexto é o uso de agentes artificiais de software, que podem atuar de forma descentralizada, com possibilidades de oferta de serviços de assistência acadêmica personalizada para cada estudante sem depender do uso de servidores centrais. Tal estratégia contrasta com as abordagens centralizadas predominantes, nas quais o estudante é obrigado a se cadastrar em servidores e a utilizar sistemas tradicionais pouco adaptáveis às propostas de modernização que envolvem uso de agentes em ambientes imersivos.

No contexto atual de desenvolvimento de agentes inteligentes baseados em LLMs, observa-se uma predominância de soluções que utilizam execução remota através de APIs de provedores cloud como OpenAI, Anthropic e Google. Embora essas abordagens ofereçam acesso a modelos de grande capacidade e escalabilidade praticamente ilimitada, elas introduzem desvantagens significativas, particularmente no que se refere à privacidade de dados e autonomia tecnológica. Como alternativa, as soluções baseadas

DOI: 10.5753/wdescentraie.2025.15941

em execução local emergem como uma opção estratégica para instituições educacionais, oferecendo vantagens substanciais conforme demonstrado na análise comparativa apresentada na Tabela 1. A tabela faz um comparativo de alguns aspectos de execuções locais, e o uso de APIs Cloud em servidores remotos.

Tabela 1. Análise Comparativa: Execução Local vs APIs Cloud

Aspecto	Execução Local	APIs Cloud
Privacidade de Dados	Controle total sobre dados sensíveis dos estudantes. Conformidade com LGPD garantida.	Compartilhamento necessário com provedores. Riscos de exposição indevida.
Custos Operacionais	Investimento inicial alto. Economia a longo prazo para uso intensivo.	Custos variáveis por uso. Podem tornar-se proibitivos em escala.
Capacidade do Modelo	Limitada por recursos locais. Modelos menores mas especializáveis.	Acesso aos modelos mais avançados. Menor controle sobre customização.
Autonomia Tecnológica	Independência de fornecedores externos. Controle sobre evolução.	Dependência de políticas e preços de terceiros.
Latência e Disponibilidade	Baixa latência. Operação offline possível.	Latência de rede. Dependência de conectividade.
Escalabilidade	Limitada pela infraestrutura local. Requer planejamento de capacidade.	Escalabilidade praticamente ilimitada.

Por outro lado, no contexto acadêmico a adoção de agentes LLM não exime a utilização de sistemas legados, como por exemplo, sistemas acadêmicos, sistema de controle de biblioteca, dentre outros. Portanto, assume-se que a completude de uma solução descentralizada deva considerar o acesso a tais sistemas por meio de interfaces padronizadas. Essa é uma preocupação que deve ser endereçada em projetos que envolvam assistentes acadêmicos.

Com base no escopo apresentado, o artigo sugere o desenvolvimento de um agente de software baseado em execução local de modelos de linguagem natural, capaz de compreender requisições em linguagem natural e resolver problemas acadêmicos através da integração dinâmica de ferramentas externas. Esta abordagem fundamenta-se na capacidade emergente dos LLMs de realizar raciocínio contextual e executar chamadas de funções (*function calling*) de forma autônoma, baseando-se em seus próprios processos cognitivos [Schick et al. 2023]. Este é um trabalho preliminar e uma discussão inicial é feita na próxima seção.

2. Premissas iniciais para construção do assistente acadêmico

Uma das premissas do assistente acadêmico é a sua capacidade uso de modelos de linguagem natural. Essa capacidade consiste em compreender nuances comunicativas humanas, combinada com técnicas de chain-of-thought reasoning [Wei et al. 2022], e permite que o agente não apenas interprete a intenção do usuário, mas também tome decisões lógicas

sobre quais ferramentas utilizar e como combiná-las para resolver problemas complexos. O diferencial proposto aqui é que este agente amplie suas potencialidades pelo acesso a ferramentais acadêmicos cujas interfaces são padronizadas.

A utilização de *function calling* (termo que designa o acesso a funções externas ao escopo do modelo implícito no agente de software) exige capacidade de reflexão e análise contextual do modelo para tomadas de decisões lógicas adequadas [Yao et al. 2022]. Quando o modelo recebe uma requisição, deve analisar não apenas o conteúdo explícito, mas também inferir necessidades implícitas, avaliar ferramentas disponíveis através de suas descrições padronizadas, e determinar a sequência ótima de ações. Esta capacidade reflexiva é fundamental para que o agente compreenda as nuances das ferramentas disponíveis, seus parâmetros específicos e funcionalidades, permitindo decisões informadas sobre a adequação de cada ferramenta para resolver o problema apresentado.

2.1. Padrões de Interoperabilidade

No ambiente educacional, um dos principais desafios é a integração de diferentes sistemas acadêmicos — gestão de cursos, bibliotecas digitais, ambientes virtuais de aprendizagem e serviços administrativos — de forma coerente e transparente para o estudante. Para isso, a adoção de padrões de interoperabilidade é essencial. Um exemplo consolidado é o *IMS Learning Tools Interoperability (IMS-LTI)* [IMS Global Learning Consortium 2019], amplamente utilizado para conectar plataformas educacionais a serviços externos de maneira uniforme, oferecendo mecanismos de autenticação, troca de dados e integração de funcionalidades sem necessidade de acoplamento rígido entre sistemas.

2.1.1. Desafio Específico da Integração LLM-Sistemas Acadêmicos

No contexto de agentes acadêmicos baseados em LLMs, surge uma necessidade específica que os padrões existentes não abordam adequadamente: a comunicação entre modelos de linguagem natural e sistemas acadêmicos legados. Este desafio apresenta características únicas que diferem das integrações sistema-a-sistema tradicionais [Fire and Guestrin 2021].

Primeiramente, os LLMs necessitam compreender funcionalidades de sistemas externos através de descrições em linguagem natural, não apenas através de especificações técnicas. Isto implica que uma ferramenta de consulta acadêmica deva disponibilizar não apenas uma API estruturada, mas também uma descrição compreensível que permita ao modelo raciocinar sobre quando e como utilizá-la [Qin et al. 2023, Schick et al. 2023].

Em segundo lugar, a heterogeneidade semântica dos sistemas acadêmicos representa um obstáculo significativo. A mesma funcionalidade pode ser expressa de diferentes formas em sistemas distintos (e.g., ”matrícula”, ”inscrição”, ”enrollment”), requerendo uma camada de abstração que permita interpretação semântica adequada pelo modelo [Chen et al. 2022].

Por fim, os requisitos de segurança e privacidade no contexto educacional demandam controles específicos, conformidade com a LGPD [Brasil 2018] e auditabilidade completa das interações, aspectos que devem ser considerados nativamente na comunicação com agentes inteligentes.

2.1.2. Necessidade de Padronização para Comunicação LLM-Sistemas

A literatura demonstra que modelos de linguagem podem integrar ferramentas externas de forma eficaz quando estas seguem padrões estruturados [Yao et al. 2022, Parisi et al. 2022]. No entanto, os padrões existentes foram desenvolvidos para integração entre sistemas convencionais, não contemplando as especificidades da comunicação com modelos de linguagem natural.

Um padrão adequado para esta comunicação deveria operar em múltiplas camadas: uma camada de descoberta que permita ao LLM identificar serviços disponíveis, uma camada de descrição semântica que facilite o raciocínio do modelo sobre a adequação de cada serviço, e uma camada de execução que garanta invocação segura e auditável [Mandel et al. 2016, Dolin et al. 2006].

A representação dual — tanto em linguagem natural quanto em formato estruturado — é fundamental para que o modelo consiga raciocinar sobre a ferramenta de forma contextual, mas também invocá-la corretamente em chamadas formais. Esta abordagem contrasta com padrões como JSON-RPC [JSON-RPC Working Group 2010] ou OpenAPI [OpenAPI Initiative 2023], que focam apenas na especificação técnica, sem considerar a interpretação semântica por agentes artificiais.

2.1.3. Elementos Essenciais para Interoperabilidade

Com base na análise das necessidades específicas da comunicação LLM-sistemas acadêmicos, alguns elementos se mostram essenciais para uma interoperabilidade eficaz:

Descoberta Automática de Serviços: Os sistemas acadêmicos devem expor suas funcionalidades de forma que modelos de linguagem possam descobri-las automaticamente, incluindo metadados sobre capacidades, restrições e contextos de uso adequados.

Descrição Semântica Rica: Cada funcionalidade deve ser acompanhada de descrições que permitam ao modelo compreender não apenas *o que* a função faz, mas *quando* e *como* utilizá-la apropriadamente no contexto acadêmico.

Padronização de Formatos: A adoção de formatos estruturados consistentes para parâmetros, respostas e códigos de erro facilita o aprendizado do modelo e reduz a complexidade de integração [Byron et al. 2015].

Controles de Segurança Nativos: Mecanismos de autenticação, autorização e auditoria devem ser integrados ao padrão, considerando os requisitos específicos de privacidade no contexto educacional.

Versionamento e Evolução: O padrão deve contemplar mecanismos que permitam evolução independente dos sistemas acadêmicos e dos agentes, mantendo compatibilidade e funcionalidade ao longo do tempo.

2.1.4. Relação com Padrões Existentes

A padronização da comunicação LLM-sistemas acadêmicos não substitui, mas complementa padrões consolidados como IMS-LTI. Enquanto o IMS-LTI foca na interoperabili-

dade entre plataformas educacionais, a necessidade emergente é de um padrão que aborde especificamente a comunicação entre inteligência artificial e sistemas acadêmicos.

Iniciativas recentes como o *Model Context Protocol (MCP)* [OpenAI et al. 2024] demonstram a viabilidade técnica de padronizar a comunicação entre LLMs e ferramentas externas. No contexto educacional, essa abordagem poderia ser especializada para contemplar requisitos específicos do domínio acadêmico, como semântica educacional, controles de privacidade estudantil e auditabilidade institucional.

A interoperabilidade torna-se, assim, peça-chave para que agentes descentralizados alcancem maturidade prática no ambiente educacional. A padronização não apenas facilita o desenvolvimento e manutenção de soluções, mas também permite que os próprios modelos compreendam de forma explícita as funcionalidades disponíveis, ampliando confiabilidade e transparência na assistência acadêmica [Lu et al. 2023].

Esta necessidade de padronização representa uma oportunidade para o desenvolvimento de especificações que, futuramente, possam evoluir para contemplar não apenas a comunicação LLM-sistemas, mas também a interoperabilidade mais ampla entre diferentes sistemas acadêmicos, criando um ecossistema educacional verdadeiramente integrado.

2.2. Validação Experimental

A validação experimental desta proposta não se restringe apenas a demonstrar a capacidade técnica de um LLM em realizar raciocínio estruturado, mas também a explorar quais cenários acadêmicos justificam o uso de modelos de grande porte em comparação com modelos menores (MLMs ou SLMs). Essa reflexão é necessária, dado que em muitos contextos institucionais o custo computacional de um LLM local pode ser elevado, enquanto soluções mais enxutas já seriam suficientes para atender às necessidades.

Os experimentos preliminares foram conduzidos com modelos hospedados na plataforma Hugging Face [Wolf et al. 2020], configurados para executar *function calling* em Python com descrições de ferramentas fornecidas em formato JSON. As consultas foram simuladas a partir de cenários típicos no ambiente universitário, tais como:

- **Consultas administrativas**: verificação de horários de disciplinas, prazos de matrícula, ou regras de avaliação. Esses cenários geralmente requerem apenas recuperação de informações estruturadas, podendo ser eficientemente tratados por modelos de menor escala, como MLMs especializados em compreensão de texto acadêmico.
- **Serviços de apoio ao estudante**: reserva de laboratórios, consulta de disponibilidade de equipamentos, ou informações sobre funcionamento de bibliotecas. Aqui, a integração com sistemas legados por meio de padrões como IMS-LTI garante interoperabilidade. Modelos médios (SMLs) podem ser suficientes, dado que as consultas envolvem passos lógicos curtos e dados bem estruturados.
- **Apoio acadêmico avançado**: elaboração de planos de estudo personalizados, recomendações de bibliografia complementar e explicação de conceitos complexos. Nesses casos, a necessidade de raciocínio contextualizado e encadeado sugere maior benefício com o uso de LLMs, capazes de combinar raciocínio simbólico e textual [Wei et al. 2022, Yao et al. 2022].
- **Cenários híbridos**: perguntas que misturam linguagem natural vaga com necessidades de consulta formal, por exemplo: “Quais disciplinas que ainda posso cursar

no próximo semestre, considerando que reprovei em Cálculo I e terei espaço para uma disciplina de 30h?". Nesses casos, um LLM pode decompor o problema em subtarefas e coordenar ferramentas externas para fornecer uma resposta integrada.

Os resultados preliminares indicaram que é possível atingir desempenho satisfatório em grande parte das consultas administrativas e operacionais. A adoção de LLMs, mostra-se mais adequada em situações que demandam raciocínio complexo, interpretação de instruções ambíguas e combinação de múltiplas ferramentas para resolver consultas abertas.

Essa constatação abre espaço para discussão sobre arquiteturas híbridas: enquanto SMLs e MLMs poderiam atender a casos de uso mais objetivos e frequentes, os LLMs poderiam ser reservados a cenários de maior complexidade cognitiva, reduzindo custos operacionais e consumo de infraestrutura sem comprometer a experiência do estudante.

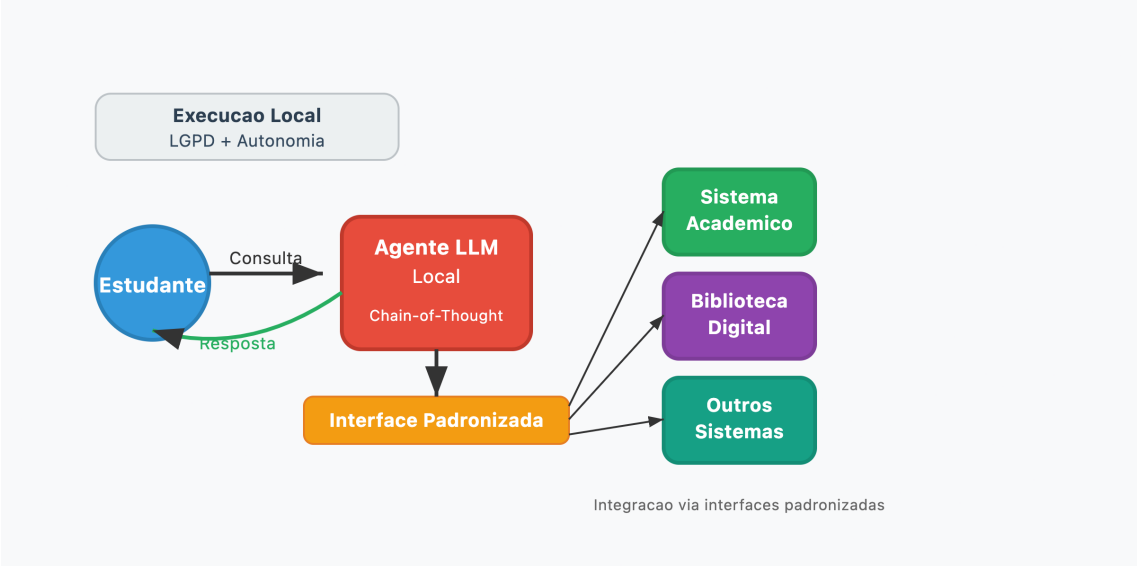

Figura 1. Visão geral da arquitetura proposta. O estudante interage com o agente LLM local através de linguagem natural, que utiliza interfaces padronizadas para acessar sistemas acadêmicos, garantindo execução local e conformidade com a LGPD.

3. Conclusões

Este trabalho investigou agentes inteligentes baseados em LLMs para assistência acadêmica descentralizada. A combinação de chain-of-thought reasoning com function calling demonstra potencial para assistentes acadêmicos adaptativos. A análise de padrões de interoperabilidade se mostra muito útil em conjunto a modelos LLMs, pois é uma forma do modelo compreender as nuances e objetivos das ferramentas integradas. Além disso, auxilia o desenvolvedor a implementar soluções de software sem ter contato via código com o agente, apenas utilizando interfaces padronizadas.

A priorização da execução local alinha-se com objetivos de privacidade e descentralização. Resultados preliminares são encorajadores, fundamentando desenvolvimento de protótipo mais sofisticado e contribuindo para o avanço de agentes inteligentes no ambiente educacional.

Referências

Brasil (2018). Lei geral de proteção de dados pessoais (lgpd) - lei nº 13.709.

Byron, L., Schrock, N., and Schafer, D. (2015). Graphql: A data query and manipulation language for apis. *Facebook Engineering Blog*.

Chen, W., Ma, X., Wang, X., and Cohen, W. W. (2022). Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*.

Dolin, R. H., Alschuler, L., Boyer, S., Beebe, C., Behlen, F. M., Biron, P. V., and Shabo, A. (2006). Hl7 clinical document architecture, release 2. In *Journal of the American Medical Informatics Association*, volume 13, pages 30–39. BMJ Publishing Group.

Fire, M. M. and Guestrin, C. (2021). Interoperability standards in healthcare: A systematic review. *IEEE Transactions on Biomedical Engineering*, 68(8):2312–2322.

Han, J., Liu, G., and Gao, Y. (2023). Learners in the metaverse: A systematic review on the use of roblox in learning. *Education Sciences*, 13(3):296.

IMS Global Learning Consortium (2019). Learning tools interoperability (lti) version 1.3 and lti advantage. Technical report, IMS Global Learning Consortium.

JSON-RPC Working Group (2010). Json-rpc 2.0 specification. Technical report.

Lin, H., Wan, S., Gan, W., Chen, J., and Chao, H. C. (2022). Metaverse in education: Vision, opportunities, and challenges. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 2857–2866. IEEE.

Lu, P., Peng, B., Cheng, H., Galley, M., Chang, K.-W., Wu, Y. N., Zhu, S.-C., and Gao, J. (2023). Chameleon: Plug-and-play compositional reasoning with large language models. *arXiv preprint arXiv:2304.09842*.

Mandel, J. C., Kreda, D. A., Mandl, K. D., Kohane, I. S., and Ramoni, R. B. (2016). Smart on fhir: a standards-based, interoperable apps platform for electronic health records. *Journal of the American Medical Informatics Association*, 23(5):899–908.

OpenAI, Anthropic, DeepMind, G., Research, M., et al. (2024). Model context protocol (mcp): An open protocol for tool integration with llms. *arXiv preprint arXiv:2407.12590*.

OpenAPI Initiative (2023). Openapi specification version 3.1.0. Technical report, Linux Foundation.

Parisi, A., Zhao, Y., and Fiedel, N. (2022). Talm: Tool augmented language models. In *arXiv preprint arXiv:2205.12255*.

Qin, Y., Liang, S., Ye, Y., Zhu, K., Yan, L., Lu, Y., Lin, Y., Cong, X., Tang, X., Qian, B., Zhao, S., Hong, L., Tian, R., Xie, R., Zhou, J., Gerstein, M., Li, D., Liu, Z., and Sun, M. (2023). Tool learning with foundation models. *arXiv preprint arXiv:2304.08354*.

Ruwodo, V., Pinomaa, A., Vesisenaho, M., Ntinda, M., and Sutinen, E. (2022). Enhancing software engineering education in africa through a metaversity. In *2022 IEEE Frontiers in Education Conference (FIE)*, pages 1–8. IEEE.

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Zettlemoyer, L., Cancedda, N., and Scialom, T. (2023). Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*.

Wang, M., Yu, H., Bell, Z., and Chu, X. (2022). Constructing an edu-metaverse ecosystem: A new and innovative framework. *IEEE Transactions on Learning Technologies*, 15(6):685–696.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., and Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. (2020). Transformers: State-of-the-art natural language processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. (2022). React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.