

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

CAYO CEZAR DE OLIVEIRA

**INVESTIGAÇÃO E COMPARAÇÃO DE ARQUITETURAS DE AGENTES
INTELIGENTES PARA PERSONALIZAÇÃO DO ENSINO DE PROGRAMAÇÃO**

MEDIANEIRA

2025

CAYO CEZAR DE OLIVEIRA

**INVESTIGAÇÃO E COMPARAÇÃO DE ARQUITETURAS DE AGENTES
INTELIGENTES PARA PERSONALIZAÇÃO DO ENSINO DE PROGRAMAÇÃO**

**Investigation and Comparison of Intelligent Agent Architectures for
Personalization of Programming Teaching**

Proposta de Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Bacharel em Ciência da Computação do Curso de Bacharelado em Ciência da Computação da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Everton Coimbra de Araújo

MEDIANEIRA

2025



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

LISTA DE ABREVIATURAS E SIGLAS

SUMÁRIO

1	INTRODUÇÃO	3
1.1	Objetivos	4
1.1.1	Objetivo geral	4
1.1.2	Objetivos específicos	4
1.2	Justificativa	5
1.3	Estrutura do trabalho	6
2	REFERENCIAL TEÓRICO	7
2.1	Aprendizagem da Programação	7
2.1.1	Educação Básica	7
2.1.2	Educação Superior	8
2.2	Agentes Inteligentes	9
2.3	Aprendizagem com IA	10
2.4	Aplicação de agentes inteligentes no ensino da programação	12
	REFERÊNCIAS	14

1 INTRODUÇÃO

O avanço das tecnologias baseadas em inteligência artificial, em especial dos Agentes Inteligentes, tem provocado transformações significativas em diversos domínios do conhecimento, sendo a educação um dos mais impactados (HOLMES; BIALIK; FADEL, 2019). A capacidade desses sistemas de operar de maneira autônoma, interativa e adaptativa tem promovido inovações na forma como processos de ensino e aprendizagem são concebidos e implementados.

Historicamente, os ambientes virtuais de aprendizagem (AVAs) foram concebidos como ferramentas de apoio à gestão e distribuição de conteúdos educacionais. No entanto, com o progresso tecnológico, especialmente no campo da inteligência artificial, tornou-se possível transcender esse paradigma, incorporando mecanismos de personalização, adaptabilidade e suporte pedagógico mais sofisticado (??).

Nesse contexto, os Agentes Inteligentes surgem como elementos fundamentais para a construção de ambientes educacionais mais responsivos às necessidades individuais dos estudantes. Ao modelar o perfil, o comportamento e o desempenho dos aprendizes, tais sistemas são capazes de oferecer recomendações, adaptar conteúdos, e propor trajetórias de aprendizagem personalizadas, promovendo maior engajamento e autonomia no processo educacional (??).

Este trabalho se insere nesse cenário de inovação, investigando e comparando diferentes arquiteturas de Agentes Inteligentes voltadas à personalização do ensino de programação. A escolha dessa temática justifica-se pela relevância da programação como competência central na formação em Computação e pela reconhecida dificuldade enfrentada por muitos estudantes nesse processo de aprendizagem. A partir dessa investigação, pretende-se contribuir para o avanço de soluções educacionais inteligentes que tornem o ensino de programação mais eficiente, acessível e centrado no estudante.

A pesquisa adotará uma abordagem principalmente qualitativa, com foco na análise e interpretação de textos científicos e técnicos. O material será obtido em base de dados reconhecidas e em repositórios de teses e dissertações. Os documentos selecionados serão analisados criticamente para obter informações sobre definições, componentes, funcionamentos, vantagens, desvantagens e para 5 qual finalidade podem ser usados os diferentes tipos de arquitetura. A comparação será feita com base em critérios definidos previamente, derivados da literatura e relacionados ao tema da personalização do ensino. Esta análise buscará identificar as convergências, divergências e aspectos que ainda não foram explorados no conhecimento atual, resultando em uma avaliação sobre quais arquiteturas são mais indicadas para cada tipo de cenário educacional.

1.1 Objetivos

1.1.1 Objetivo geral

Investigar e comparar arquiteturas distintas de agentes inteligentes quanto à sua eficácia em adaptar estratégias de ensino de programação.

1.1.2 Objetivos específicos

Para atingir o objetivo geral da pesquisa, propõem-se os seguintes objetivos específicos:

- Consolidar os conceitos fundamentais relacionados à aprendizagem personalizada, à Inteligência Artificial aplicada à educação e aos agentes inteligentes, com foco em suas funcionalidades e aplicações no contexto educacional;
- Descrever as arquiteturas de agentes inteligentes baseadas em *regras*, abordando seus componentes centrais (base de regras, motor de inferência, memória de trabalho), vantagens como explicabilidade e limitações como rigidez estrutural;
- Apresentar agentes baseados em *aprendizado de máquina*, incluindo diferentes paradigmas (supervisionado, não supervisionado e por reforço), suas técnicas mais comuns (como redes neurais e árvores de decisão), bem como benefícios e desafios;
- Investigar arquiteturas *híbridas*, que integram abordagens baseadas em regras e aprendizado de máquina, discutindo seus princípios de funcionamento, potenciais sinérgicos e complexidades adicionais;
- Estabelecer critérios comparativos relevantes, tais como:
 - Adaptabilidade;
 - Escalabilidade;
 - Transparência;
 - Complexidade de desenvolvimento;
 - Capacidade de representação do conhecimento.
- Avaliar as arquiteturas em tarefas de personalização como:
 - Modelagem do aluno;
 - Recomendação de conteúdo;
 - Feedback adaptativo.

- Discutir as implicações da escolha arquitetural no desenvolvimento de agentes educacionais inteligentes, identificando desafios recorrentes e propondo caminhos para futuras investigações na área.

1.2 Justificativa

O ensino de programação nos cursos superiores da área de Tecnologia da informação constitui, historicamente, um dos principais desafios pedagógicos enfrentados por discentes e docentes. As disciplinas introdutórias de programação exigem o desenvolvimento de habilidades cognitivas específicas – tais como o raciocínio lógico, a abstração computacional, a decomposição de problemas e a construção de algoritmos – que nem sempre são pré-existentes no repertório dos estudantes ingressantes. Como discutido por(??), tais lacunas comprometem o desempenho acadêmico, contribuindo para elevados índices de reprovação e evasão.

Nesse cenário, a adoção de abordagens baseadas em inteligência artificial, em especial aquelas centradas em Agentes Inteligentes, revela-se promissora como estratégia de personalização do ensino e apoio à aprendizagem. Agentes Inteligentes são sistemas autônomos capazes de perceber o ambiente, processar informações e tomar decisões de forma adaptativa. Quando aplicados ao domínio educacional, esses agentes podem modelar o perfil do estudante, adaptar o conteúdo às suas necessidades específicas e promover um acompanhamento contínuo do seu progresso (FERRÃO; BIDARRA JOSÉ E ROCIO, 2020).

Visto que ainda haja uma lacuna na literatura em relação a uma análise comparativa mais aprofundada das características intrínsecas das diferentes arquiteturas de agentes com os requisitos específicos da personalização da aprendizagem como observado em (??), muitas pesquisas tendem a se concentrar na aplicação de uma arquitetura específica ou na avaliação de um sistema particular, sem oferecer uma visão comparativa que possa orientar novos avanços ou auxiliar na escolha de abordagens para diferentes contextos.

Por isso, é recorrente que desenvolvedores e pesquisadores se deparem com a questão de qual arquitetura melhor se alinha aos seus objetivos pedagógicos e quais são suas limitações técnicas. Isso ocorre porque, enquanto algumas arquiteturas oferecem maior explicabilidade, outras podem proporcionar maior adaptabilidade, entre outras características.

A presente pesquisa justifica-se, portanto, pela necessidade de investigar criticamente as diferentes arquiteturas de agentes inteligentes aplicáveis ao contexto do ensino de programação. Por meio de uma análise teórica comparativa, busca-se identificar as características estruturais que tornam determinadas arquiteturas mais adequadas do que outras para esse propósito. A compreensão dessas diferenças poderá subsidiar o desenvolvimento futuro de sistemas educacionais inteligentes mais eficazes e responsivos às particularidades do processo de aprendizagem em programação.

1.3 Estrutura do trabalho

Este trabalho está organizado da seguinte forma: o capítulo 1, intitulado “introdução”, apresenta a contextualização do tema, a delimitação do problema, os objetivos gerais e específicos, bem como a justificativa e a estrutura do trabalho. O capítulo 2, “Referencial teórico”, aborda os principais conceitos relacionados à aprendizagem com inteligência artificial, à aprendizagem de programação, aos Agentes Inteligentes e à interseção entre estes temas, com foco na utilização de Agentes Inteligentes no ensino de programação. O capítulo 3, “Materiais e Métodos”, descreve as abordagens metodológicas adotadas e os repositórios utilizados no desenvolvimento da pesquisa.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta os principais fundamentos teóricos que embasam a presente pesquisa, organizada em quatro seções: (1) aprendizagem da programação, (2) agentes inteligentes, (3) aprendizagem com inteligência artificial e (4) a interseção entre esses temas, com foco na aplicação de agentes inteligentes no ensino de programação.

2.1 Aprendizagem da Programação

2.1.1 Educação Básica

A aprendizagem de programação tem se consolidado como uma competência fundamental no contexto da sociedade digital, sendo reconhecida por seu potencial de fomentar a autonomia discente, o pensamento crítico, a criatividade e a fluência tecnológica. No Brasil, o ensino de programação na educação básica passou por uma trajetória evolutiva marcada por três fases distintas: uma fase introdutória, centrada no uso da linguagem LOGO e nas ações do programa ProInfo (1996–2000); uma etapa intermediária, caracterizada pelo foco na lógica matemática e no uso da internet (2001–2007); e, posteriormente, uma fase de ampliação do acesso, impulsionada por plataformas com interfaces mais intuitivas, como o Scratch (2008–2016) (FERREIRA RONI COSTA E DUARTE, 2019).

De acordo com Ferreira e Duarte (FERREIRA RONI COSTA E DUARTE, 2019), a inserção da programação no currículo escolar deixou de ser uma exclusividade do ensino superior e passou a ser incorporada às práticas educacionais desde os anos iniciais, por meio de abordagens interdisciplinares e lúdicas, como o desenvolvimento de jogos e robótica educacional. Essa incorporação está alinhada a movimentos internacionais — como a iniciativa *Hour of Code* (2014) — que buscaram promover o letramento digital em escala global.

A implementação do ensino de programação na educação básica demanda abordagens pedagógicas que articulem os conteúdos computacionais com as particularidades cognitivas dos estudantes em fase inicial de escolarização. Nesse cenário, ferramentas como o Scratch têm se destacado por sua proposta de programação em blocos, que substitui a escrita textual de código por elementos visuais manipuláveis, promovendo a compreensão de conceitos computacionais de forma mais acessível e intuitiva (SOUZA; FALCÃO; MELLO, 2021).

Conforme Kelleher e Pausch (2005 apud Souza, Falcão e Mello, 2021), a programação em blocos visa reduzir a barreira de entrada imposta pela sintaxe das linguagens tradicionais, permitindo que os aprendizes concentrem seus esforços na lógica e na semântica dos algoritmos. Tal abordagem estimula a aprendizagem criativa, facilita a abstração progressiva e promove o desenvolvimento de habilidades cognitivas associadas à resolução de problemas. A estrutura visual dos blocos, que se encaixam como peças de quebra-cabeça, auxilia na constru-

ção mental da lógica sequencial e na identificação de padrões de execução (SOUZA; FALCÃO; MELLO, 2021).

Destaca-se ainda o Scratch Jr., uma versão voltada para crianças entre 5 e 7 anos, com interface simplificada para dispositivos móveis. Essa ferramenta tem sido frequentemente associada a práticas pedagógicas como o *storytelling* digital, em que os alunos criam narrativas interativas para dar significado às atividades de programação (SOUZA; FALCÃO; MELLO, 2021).

A literatura aponta que uma das competências centrais desenvolvidas por meio da programação na educação básica é o pensamento computacional — definido por Wing (2006) como a capacidade de decompor problemas, identificar padrões, abstrair soluções e formular algoritmos de forma estruturada (SOUZA; FALCÃO; MELLO, 2021). Selby e Woollard (2013) ampliam essa definição ao caracterizá-lo como uma forma de pensamento que articula cognição, lógica e abstração, sendo essencial para a formação de indivíduos autônomos e aptos a enfrentar desafios complexos em contextos digitais.

Embora autores como Souza, Falcão e Mello (SOUZA; FALCÃO; MELLO, 2021) abordem a Educação Básica de maneira integrada, é necessário considerar a segmentação legal e pedagógica dessa etapa, conforme previsto na Lei de Diretrizes e Bases da Educação Nacional (Lei nº 9.394/96), que a divide em: Educação Infantil, Ensino Fundamental e Ensino Médio. Cada segmento possui especificidades cognitivas e metodológicas que influenciam a forma como a programação é introduzida e desenvolvida. O Scratch Jr., por exemplo, é indicado para a Educação Infantil e os anos iniciais do Ensino Fundamental, ao passo que o Scratch tradicional é mais apropriado para estudantes do Ensino Fundamental II e Médio, por oferecer recursos computacionais mais complexos, como variáveis, estruturas condicionais e laços de repetição. Assim, a prática pedagógica exige adaptações metodológicas compatíveis com a faixa etária e o nível de desenvolvimento dos estudantes, mesmo que essa diferenciação nem sempre seja evidenciada nas referências bibliográficas.

2.1.2 Educação Superior

No ensino superior, os desafios associados à aprendizagem de lógica de programação e algoritmos assumem papel central na formação de estudantes dos cursos de Computação, Engenharia e áreas afins. Por serem conteúdos introdutórios, mas com alta complexidade conceitual, essas disciplinas são muitas vezes identificadas como um dos principais obstáculos para os ingressantes, exigindo grande capacidade de abstração, raciocínio lógico e resolução de problemas. Dificuldades nesse processo podem resultar em altos índices de reprovação e evasão, o que destaca a necessidade de metodologias de ensino mais eficazes (ZACARIAS RODRIGO OLIVEIRA E MELLO, 2019). Nesse contexto, têm sido incorporadas metodologias ativas e tecnologias didáticas com o objetivo de tornar o aprendizado mais acessível e engajador. Estratégias como o uso de ferramentas computacionais para visualização de algoritmos,

aprendizagem baseada em problemas (PBL) e o ensino híbrido têm se mostrado promissoras para estimular a participação ativa dos estudantes e facilitar a assimilação dos conteúdos (ZACARIAS RODRIGO OLIVEIRA E MELLO, 2019). Além disso, a literatura aponta que o desenvolvimento de competências na área de programação no ensino superior requer mais do que o domínio da sintaxe: espera-se que os alunos compreendam algoritmos complexos, estruturas de dados, paradigmas de programação e princípios de engenharia de software. Para isso, a atuação do docente como mediador, aliado ao uso de recursos tecnológicos e práticas pedagógicas inovadoras, torna-se essencial para superar a rigidez dos métodos tradicionais e melhorar os índices de aprendizagem (ZACARIAS RODRIGO OLIVEIRA E MELLO, 2019).

Diante desse cenário, cresce o interesse por tecnologias educacionais que contribuam para personalizar o ensino e apoiar os discentes nas dificuldades específicas de aprendizagem. Entre essas tecnologias, destacam-se os tutores inteligentes, os ambientes adaptativos de aprendizagem e os agentes pedagógicos baseados em inteligência artificial, cuja eficácia tem sido evidenciada em estudos recentes sobre personalização e suporte adaptativo à aprendizagem (HOLMES; BIALIK; FADEL, 2019).

2.2 Agentes Inteligentes

O conceito de agente inteligente é um pilar central da Inteligência Artificial e constitui uma das abordagens mais consolidadas para o desenvolvimento de sistemas autônomos. De acordo com Russell e Norvig (RUSSELL; NORVIG, 2010), um agente pode ser definido como “qualquer coisa que possa ser vista como percebendo seu ambiente por meio de sensores e atuando sobre esse ambiente por meio de atuadores” (p. 34). Essa definição abrange desde agentes físicos, como robôs, até entidades puramente computacionais que operam em ambientes virtuais.

A racionalidade é o critério central na caracterização de um agente inteligente. Um agente é considerado racional quando, para cada sequência possível de percepções, escolhe a ação que maximiza sua medida de desempenho esperada, com base nas informações disponíveis e nos objetivos estabelecidos (RUSSELL; NORVIG, 2010, p. 38). Assim, a racionalidade de um agente não depende de uma onisciência idealizada, mas da sua capacidade de tomar decisões eficazes a partir do conhecimento parcial e contextual que possui.

Os autores distinguem entre a função do agente, entendida como um mapeamento idealizado de percepções para ações, e o programa do agente, que é sua implementação prática em hardware ou software. Enquanto a função do agente representa um modelo teórico de comportamento ótimo, o programa do agente é a materialização computacional que governa suas ações em tempo de execução.

Com base na complexidade funcional e nas capacidades cognitivas, Russell e Norvig (RUSSELL; NORVIG, 2010) classificam os agentes em cinco categorias principais:

1. Agentes de reflexo simples: atuam com base em regras condicionais diretas do tipo se-então, sem considerar o histórico de interações;
2. Agentes baseados em modelo: mantêm uma representação interna do estado do ambiente, permitindo decisões mais informadas;
3. Agentes orientados a objetivos: selecionam ações com base na capacidade de atingir metas predefinidas;
4. Agentes baseados em utilidade: avaliam os possíveis estados futuros com base em uma função de utilidade, buscando não apenas atingir objetivos, mas otimizá-los;
5. Agentes de aprendizado: adaptam seu comportamento com base na experiência, ajustando seus modelos internos e estratégias de ação ao longo do tempo.

A racionalidade de um agente está condicionada a quatro componentes fundamentais: (i) a medida de desempenho que avalia a qualidade das ações; (ii) o conhecimento prévio que o agente possui sobre o ambiente; (iii) o conjunto de ações disponíveis; e (iv) a sequência de percepções que recebe ao longo do tempo. Essa formulação permite a construção de agentes eficazes mesmo em contextos complexos, incertos e parcialmente observáveis — características frequentemente encontradas em ambientes educacionais. Nesse sentido, agentes inteligentes se mostram particularmente adequados para o desenvolvimento de sistemas inteligentes de tutoria e outras aplicações voltadas à personalização do ensino.

Essa aplicabilidade educacional tem sido explorada em pesquisas voltadas ao ensino-aprendizagem online, nas quais agentes inteligentes são empregados para interpretar dados educacionais, realizar inferências automáticas e interagir com os usuários, com o objetivo de personalizar o processo de ensino e oferecer suporte aos mediadores pedagógicos (FERRÃO; BIDARRA JOSÉ E ROCIO, 2020).

2.3 Aprendizagem com IA

A aplicação da Inteligência Artificial (IA) na educação tem se consolidado como uma abordagem eficaz para personalizar e otimizar o processo de ensino-aprendizagem. Recentemente, estudos destacam o papel dos Sistemas Tutores Inteligentes (STIs) na oferta de instrução adaptativa, proporcionando feedback personalizado e suporte em tempo real aos estudantes, o que contribui para melhorias no engajamento e no desempenho acadêmico. Esses sistemas, ao modelarem o conhecimento do domínio e o perfil do aluno, permitem a implementação de estratégias pedagógicas adaptativas, alinhadas às necessidades individuais dos estudantes (CHEN; OUTROS, 2025).

Com o avanço das tecnologias digitais e a consolidação do ensino remoto — acentuada pela pandemia da COVID-19 —, a integração da IA na educação passou a ocupar um

papel ainda mais central nas discussões sobre inovação pedagógica. Embora a literatura especializada discuta há décadas o potencial transformador dessas tecnologias, seu uso efetivo nas instituições de ensino permanece limitado, sobretudo no que se refere à automação inteligente de processos e à personalização de experiências educacionais (FERRÃO; BIDARRA JOSÉ E ROCIO, 2020). A maioria dos ambientes virtuais de aprendizagem carece de funcionalidades inteligentes plenamente operacionais, o que se deve, em grande parte, à ausência de políticas institucionais de capacitação e à subutilização das ferramentas disponíveis.

Estudos de campo revelam uma lacuna significativa entre as capacidades oferecidas pela IA e sua aplicação prática nos contextos educacionais. Há uma carência notável de sistemas capazes de realizar monitoramento contínuo do desempenho discente, propor trilhas adaptativas de aprendizagem e promover mediação pedagógica automatizada. Além disso, a escassez de indicadores estratégicos nos sistemas educacionais compromete a qualidade da tomada de decisão por parte de gestores e docentes (FERRÃO; BIDARRA JOSÉ E ROCIO, 2020).

Historicamente, os primeiros experimentos envolvendo IA na educação surgiram com o intuito de simular interações humano-computador, como evidenciado pelos sistemas ELIZA e SCHOLAR. Com o passar do tempo, esses protótipos deram lugar a sistemas mais sofisticados, capazes de fornecer feedback personalizado e desempenhar funções análogas às de um tutor humano. Apesar da evolução técnica e da disponibilidade de tecnologias baseadas em aprendizado de máquina, o uso cotidiano dessas soluções em contextos educacionais ainda é incipiente (FERRÃO; BIDARRA JOSÉ E ROCIO, 2020).

A literatura contemporânea aponta que as aplicações de IA na educação podem beneficiar distintos atores do processo educativo: estudantes, docentes e gestores institucionais. Para os discentes, os sistemas adaptativos visam alinhar os conteúdos aos estilos e ritmos de aprendizagem individuais. No âmbito docente, a IA pode automatizar tarefas repetitivas, facilitar avaliações e oferecer suporte ao acompanhamento do progresso dos alunos. Já para os gestores, a tecnologia possibilita a geração de análises preditivas e relatórios estratégicos, auxiliando na formulação de políticas voltadas à retenção e ao desempenho acadêmico. Apesar dessas potencialidades, persistem desafios relacionados à aceitação tecnológica, à capacitação profissional e à reconfiguração das práticas pedagógicas tradicionais (FERRÃO; BIDARRA JOSÉ E ROCIO, 2020).

A análise das percepções de usuários, como demonstrado no estudo do modelo *Wizard User*, reforça a demanda por ambientes mais responsivos, interativos e ajustados às necessidades específicas dos estudantes. Nesse contexto, a personalização surge como um requisito recorrente, viabilizado por funcionalidades como recomendações automáticas, notificações inteligentes e suporte contínuo ao processo de aprendizagem. Esses elementos contribuem para um cenário educacional em que a IA atua como catalisadora da transformação digital (FERRÃO; BIDARRA JOSÉ E ROCIO, 2020).

Adicionalmente, a integração da IA com metodologias ativas tem se destacado como uma estratégia eficaz para fomentar o engajamento estudantil e o desenvolvimento da autonomia intelectual. A IA, nesse cenário, desempenha o papel de mediadora do processo de construção do conhecimento, fornecendo recursos para a resolução colaborativa de problemas e para a vivência de experiências de aprendizagem mais dinâmicas, críticas e participativas (GOMES; VERçOSA BRUNO FRANCISCO MONTEIRO E PINTO, 2024).

Por fim, destaca-se a relevância dos sistemas inteligentes no suporte à aprendizagem personalizada em ambientes virtuais. Através de mecanismos de adaptação automática, esses sistemas são capazes de ajustar o ritmo, o conteúdo e as estratégias didáticas de acordo com o perfil cognitivo e comportamental de cada estudante. Essa capacidade de adaptação favorece uma aprendizagem mais significativa, contextualizada e centrada no aluno, ampliando o protagonismo discente e a eficácia das ações pedagógicas (GOMES; VERçOSA BRUNO FRANCISCO MONTEIRO E PINTO, 2024).

2.4 Aplicação de agentes inteligentes no ensino da programação

A crescente incorporação da inteligência artificial (IA) na educação vem transformando profundamente a forma como o ensino de programação é estruturado, avaliado e experienciado pelos alunos. Nesse contexto, os agentes inteligentes surgem como recursos pedagógicos potentes, oferecendo desde suporte automatizado até experiências interativas e personalizadas de aprendizado. A utilização de modelos de linguagem natural como o ChatGPT tem ampliado as possibilidades de tutoria automatizada e feedback em tempo real. Estudos recentes apontam que esses agentes são capazes de resolver corretamente uma variedade de problemas em cursos de ciência da computação, incluindo questões objetivas, respostas discursivas e tarefas de codificação. Contudo, essa capacidade técnica também levanta desafios éticos e pedagógicos, principalmente quando estudantes utilizam essas ferramentas para obter respostas prontas, em vez de construírem ativamente seu conhecimento. Para lidar com essas limitações, pesquisadores têm sugerido mudanças no formato das atividades, como a inclusão de informações distrativas e a reformulação da avaliação, com o objetivo de tornar o uso desses sistemas mais consciente e formativo (WANG *et al.*, 2023).

Paralelamente, abordagens mais interativas têm sido exploradas com base no modelo *Learning by Teaching* (LBT), em que o estudante assume o papel de instrutor, ensinando um agente inteligente. Um exemplo representativo dessa abordagem é o sistema TeachYou, que apresenta o agente AlgoBo, um modelo de linguagem configurado para se comportar como um aluno com dificuldades reais e lacunas de conhecimento. Esse sistema utiliza uma estratégia em duas etapas — refletir e responder — permitindo ao agente simular dúvidas autênticas e revisar suas próprias respostas com base nas explicações fornecidas pelo estudante. Além disso, incorpora uma dinâmica chamada mudança de papéis, em que o agente alterna entre aprendiz e questionador, incentivando o aluno a elaborar explicações mais claras e profundas. A análise

das interações demonstrou que esse tipo de agente é eficaz na promoção do conhecimento, da metacognição e do engajamento, especialmente quando comparado a sistemas mais passivos. Essa abordagem evidencia o potencial dos modelos de linguagem como agentes ensináveis, projetados para criar experiências pedagógicas mais desafiadoras e enriquecedoras (JIN *et al.*, 2024).

Outra contribuição relevante da literatura refere-se à aplicação prática de agentes inteligentes em sistemas de avaliação automática de código. A integração do modelo GPT-3.5-Turbo ao sistema Artemis, por exemplo, possibilitou que os alunos recebessem feedback imediato e contextualizado sobre seus códigos, sem que a solução fosse diretamente fornecida. O propósito da arquitetura adotada nesse caso foi não apenas auxiliar na correção, mas também estimular a aprendizagem por meio de sugestões de melhoria. O estudo identificou diferentes perfis de uso e destacou a importância de equilibrar automação e orientação pedagógica. Apesar dos benefícios, foram observadas limitações, como a tendência do agente a fornecer respostas genéricas e a possibilidade de os estudantes se tornarem excessivamente dependentes da ferramenta, o que pode comprometer o desenvolvimento da autonomia. Essa experiência prática reforça a necessidade de arquiteturas mais refinadas, capazes de se adaptar ao nível de conhecimento do aluno e à complexidade das tarefas (FRANKFORD *et al.*, 2024).

Diante desse panorama, observa-se que diferentes tipos de agentes inteligentes oferecem vantagens específicas, dependendo dos objetivos educacionais, do grau de autonomia desejado para o estudante e do nível de personalização buscado. Enquanto algumas abordagens apostam em agentes especialistas, como no caso do Artemis, outras investem em agentes ensináveis, como o AlgoBo, que se apoiam na interação ativa para promover o aprendizado. Há ainda estratégias voltadas à adaptação de conteúdo e à identificação de uso indevido, reforçando o papel desses agentes como mediadores do processo educativo.

Nesse contexto, torna-se fundamental realizar uma análise teórica comparativa das diferentes arquiteturas de agentes inteligentes aplicadas ao ensino de programação. Esta pesquisa tem como objetivo identificar, a partir da literatura, quais elementos arquitetônicos se mostram mais promissores para a personalização do aprendizado. Ao reunir e sistematizar evidências conceituais, busca-se contribuir para o desenvolvimento de agentes mais eficazes e alinhados às demandas contemporâneas da educação em computação.

REFERÊNCIAS

- CHEN, X.; OUTROS. The impact of artificial intelligence (ai) on students' academic performance: A systematic review. **Education Sciences**, v. 15, n. 3, p. 343, 2025. Disponível em: <https://www.mdpi.com/2227-7102/15/3/343>.
- FERREIRA RONI COSTA E DUARTE, S. Ensino de programação: trajetória histórico-social e os avanços na cultura digital do brasil. **Revista Brasileira de Ensino de Ciência e Tecnologia**, Curitiba, v. 12, n. 1, p. 386–408, 2019. Disponível em: <https://periodicos.utfpr.edu.br/rbect/article/view/7532>.
- FERRÃO, E.; BIDARRA JOSÉ E ROCIO, V. Requisitos de um agente inteligente de apoio ao ensino-aprendizagem on-line: modelo wizard user. **International Journal of Development Research**, v. 11, n. 06, p. 47565–47570, 2020. Disponível em: https://www.researchgate.net/publication/352657537_Requisitos_de_um_Agente_Inteligente_de_Apoio_ao_Ensino-Aprendizagem_On-Line_Modelo_Wizard_User.
- FRANKFORD, E. *et al.* Ai-tutoring in software engineering education. **arXiv preprint**, arXiv:2404.02548, 2024. Disponível em: <https://arxiv.org/abs/2404.02548>.
- GOMES, A. J. F.; VERÇOSA BRUNO FRANCISCO MONTEIRO E PINTO, C. R. S. e. M. C. C. d. e. S. C. d. S. e. R. O. B. d. Potencializando a aprendizagem ativa com tecnologia de ia. **Revista Ibero-Americana de Humanidades, Ciências e Educação**, v. 10, n. 8, p. 3625–3631, 2024. Disponível em: <https://doi.org/10.51891/rease.v10i8.15451>.
- HOLMES, W.; BIALIK, M.; FADEL, C. **Artificial Intelligence in Education: Promises and Implications for Teaching and Learning**. Boston: Center for Curriculum Redesign, 2019.
- JIN, H. *et al.* Teach ai how to code: Using large language models as teachable agents for programming education. *In: Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24)*. Honolulu, HI, USA: [s.n.], 2024. Disponível em: <https://arxiv.org/abs/2309.14534>.
- RUSSELL, S. J.; NORVIG, P. **Inteligência Artificial**. 3. ed. São Paulo: Pearson Education do Brasil, 2010.
- SOUZA, F. A. d.; FALCÃO, T. P.; MELLO, R. F. O ensino de programação na educação básica: uma revisão da literatura. *In: Anais do Simpósio Brasileiro de Informática na Educação (SBIE)*. Porto Alegre: Sociedade Brasileira de Computação, 2021. p. 1265–1275. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/18148>.
- WANG, T. *et al.* Exploring the role of ai assistants in computer science education: Methods, implications, and instructor perspectives. *In: IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. Washington, DC, EUA: [s.n.], 2023. p. 92–102. Disponível em: <https://arxiv.org/abs/2306.03289>.
- ZACARIAS RODRIGO OLIVEIRA E MELLO, D. R. B. Metodologias de ensino de lógica de programação e algoritmos em cursos de graduação. **Revista Interdisciplinar Pensamento Científico**, v. 5, n. 2, 2019. Disponível em: <https://reinpec.cc/index.php/reinpec/article/view/280>.