



Aplicações da IA Generativa no Ensino de Programação para Cursos de Engenharia e Computação: Uma Abordagem Inovadora no Desenvolvimento de Habilidades de Codificação

DOI: 10.37702/2175-957X.COBENGE.2024.4874

Autores: SERGIO RICARDO XAVIER DA SILVA

Resumo: A implementação de Inteligência Artificial (IA) Generativa no ensino de programação de computadores surge como uma solução promissora aos desafios encontrados em cursos de Engenharia, Computação e áreas correlatas, caracterizados por altas taxas de evasão e reprovação. Este artigo explora a integração de tecnologias de IA Adaptativa, incluindo chatbots, games inteligentes, juízes online, assistentes inteligentes e sistemas de tutoria inteligente. A análise centra-se no impacto dessas ferramentas na personalização do aprendizado, na motivação dos alunos e na eficácia na aquisição de habilidades de programação, propondo um paradigma de ensino mais adaptativo e interativo.

Palavras-chave: Inteligência Artificial Generativa, Ensino de Programação, Tecnologias Adaptativas, Personalização do Aprendizado, Engajamento dos Estudantes.

APLICAÇÕES DA IA GENERATIVA NO ENSINO DE PROGRAMAÇÃO PARA CURSOS DE ENGENHARIA E COMPUTAÇÃO: UMA ABORDAGEM INOVADORA NO DESENVOLVIMENTO DE HABILIDADES DE CODIFICAÇÃO

1 INTRODUÇÃO

Disciplinas introdutórias de programação são fundamentais na formação acadêmica em Engenharias e Computação, mas frequentemente enfrentam problemas como desinteresse e dificuldade de aprendizagem por parte dos alunos. A integração da IA no ensino de programação oferece uma abordagem atualizada, centrada no estudante, que pode potencialmente reduzir as barreiras ao aprendizado.

A utilização da Inteligência Artificial Generativa, especificamente o ChatGPT, no ensino de programação de computadores, revela-se como uma abordagem que tanto complementa quanto desafia os métodos tradicionais de ensino.

Entretanto, a utilização da IA Generativa no ensino de programação não é isenta de desafios. Iqbal *et al.* (2022) apontam que, apesar das vantagens como a assessoria no planejamento das aulas e na avaliação, há preocupações significativas relacionadas ao potencial de plágio, além de uma possível dependência que pode prejudicar o desenvolvimento do pensamento crítico e das habilidades de resolução de problemas. De acordo com Rodrigues e Rodrigues (2023), a utilização da IA Generativa na educação superior traz à tona uma questão complexa que vai além do plágio, a necessidade de fomentar um desenvolvimento crítico e criativo entre os alunos.

Essa dualidade entre os benefícios potenciais e os riscos associados destaca a necessidade de uma abordagem equilibrada no uso de ferramentas de IA Generativa em ambientes educacionais. Conforme sugerido por Elkhodr *et al.* (2023), a integração do ChatGPT como uma ferramenta de assistência pode melhorar a compreensão e o desempenho dos alunos em tarefas de programação, desde que acompanhada de diretrizes éticas e metodológicas claras que orientem seu uso apropriado.

Conforme destacado por Menta e Brito (2024), a implementação da IA nas instituições de ensino transcende a simples automação de tarefas, promovendo uma aprendizagem personalizada e ajustada às necessidades individuais de cada estudante. No contexto do ensino de programação, a IA generativa introduz uma nova dimensão de interatividade e personalização. Essa tecnologia não apenas facilita o acesso imediato a informações e a solução de problemas complexos, mas também instiga a reflexão crítica sobre o uso ético dessas ferramentas.

Nesse contexto, este artigo visa avaliar o impacto da utilização da Inteligência Artificial Generativa, especificamente através de ferramentas como o ChatGPT, no ensino de programação. O objetivo é explorar como essa tecnologia pode servir tanto como complemento quanto como desafio aos métodos tradicionais de ensino, proporcionando uma aprendizagem personalizada e interativa. Além disso, busca-se entender as complexidades e desafios associados à sua integração, como questões éticas e o potencial impacto no desenvolvimento de habilidades críticas e de resolução de problemas nos alunos.

No cenário atual da Educação, as Tecnologias Adaptativas emergem como ferramentas importantes para enfrentar os desafios associados à personalização da aprendizagem. Essas tecnologias, empregando algoritmos avançados e técnicas de

inteligência artificial, ajustam o conteúdo educacional para atender às necessidades individuais de cada estudante. Elas não apenas aumentam a eficiência pedagógica, mas também melhoram significativamente o engajamento dos estudantes, oferecendo novas possibilidades para superar as taxas tradicionalmente altas de evasão e reprovação em cursos de Engenharia, Computação e áreas correlatas.

O presente trabalho está organizado da seguinte maneira. A seção 2 estabelece as principais distinções entre a IA Tradicional e a IA Generativa. A seção 3 discute como a implementação de IA Generativa no ensino de programação de computadores pode transformar positivamente o processo de ensino-aprendizagem, explorando aplicações como *chatbots*, jogos inteligentes, juizes online, assistentes inteligentes e sistemas de tutoria. A seção 4 descreve uma pesquisa realizada com o uso do ChatGPT em aulas de Técnicas de Programação, cujos resultados são apresentados na seção 5. Por fim, a seção 6 conclui o artigo apresentando as considerações finais e perspectivas futuras.

2 IA TRADICIONAL VERSUS IA GENERATIVA

A evolução da Inteligência Artificial trouxe consigo uma distinção significativa entre as abordagens tradicionais e as generativas, impactando diretamente o ensino de programação de computadores. A IA Tradicional é predominantemente utilizada para análises e previsões baseadas em dados históricos, operando dentro de um *framework*¹ de aprendizado supervisionado. Essa forma de IA foca em automatizar tarefas específicas e lógicas, dependendo fortemente de grandes conjuntos de dados rotulados e regras predefinidas. Em contrapartida, a IA Generativa, ou *GenAI*, é caracterizada pela sua capacidade de criar e gerar novos dados, conteúdos e até arte de forma autônoma, utilizando aprendizado não supervisionado ou semisupervisionado através de redes neurais complexas, como as Redes Adversárias Generativas (*GANs - Generative Adversarial Networks*).

As *GANs* representam uma classe de algoritmos de aprendizado de máquina dentro do paradigma de aprendizado não supervisionado, proposto inicialmente por Goodfellow *et al.* (2014). Esta metodologia envolve duas redes neurais distintas, a geradora e a discriminadora, que competem uma contra a outra em um jogo teórico. A rede geradora tem como objetivo produzir amostras artificiais indistinguíveis das reais, enquanto a rede discriminadora esforça-se para diferenciar as amostras reais das geradas pela rede adversária. O processo iterativo de treinamento das *GANs* melhora continuamente a qualidade das amostras geradas, tornando-as progressivamente mais difíceis de serem diferenciadas das originais. Este mecanismo proporciona avanços significativos em diversos campos de aplicação, como visão computacional, síntese de voz, e criação de arte, destacando-se pela capacidade de gerar dados novos e realistas a partir de um conjunto limitado de exemplos de treinamento.

Segundo Park *et al.* (2021), a crescente complexidade das *GANs* e suas aplicações promissoras no campo da visão computacional, assim como na geração de dados não visuais, como voz e linguagem natural, ampliam consideravelmente o horizonte do ensino de programação.

A Tabela 1 resume as principais diferenças entre a IA Tradicional e a IA Generativa, destacando suas características, abordagens de aprendizado, geração de conteúdo, entre outros.

¹ Um "*framework*" é uma estrutura conceitual e tecnológica que orienta e suporta o desenvolvimento de aplicações, fornecendo componentes pré-fabricados e padrões de projeto. Ele facilita a conformidade com normas específicas, mas também pode limitar a criatividade do desenvolvedor ao impor certas estruturas e metodologias.

Tabela 1 – IA Tradicional versus IA Generativa

Característica	IA Tradicional	IA Generativa
Objetivo Principal	Focada na automatização e melhoria de tarefas específicas utilizando lógica formal e análises estruturadas.	Visa a criação de novos dados e conteúdo inovador, expandindo as capacidades criativas da IA.
Abordagem de Aprendizado	Utiliza algoritmos baseados em dados históricos e regras predefinidas para melhorar a precisão das tarefas.	Emprega aprendizado de máquina avançado, incluindo redes neurais profundas para criar e aprender continuamente.
Tomada de Decisões	Segue uma lógica predeterminada e baseia-se em parâmetros bem definidos para automação e previsões.	Adota uma abordagem não-determinística que permite a geração de soluções inovadoras e até mesmo a improvisação.
Geração de Conteúdo	Limitada na geração autônoma de novos conteúdos; tipicamente realiza tarefas definidas e repetitivas.	Capaz de gerar conteúdo único, como novas imagens, textos, música e outros tipos de dados criativos.
Aplicações Comuns	Amplamente usada em sistemas de recomendação, diagnósticos médicos, processamento de transações e reconhecimento de padrões.	Aplicada em arte generativa, desenvolvimento de jogos, educação interativa e inovações em entretenimento.
Treinamento do Modelo	Dependente de grandes volumes de dados rotulados, muitas vezes exigindo supervisão e ajustes manuais.	Pode ser treinada com menos supervisão, utilizando também dados não rotulados, graças ao aprendizado por reforço e transferência.
Limitações	Pode ser limitada pela qualidade e viés dos dados históricos; pode ter dificuldade em se adaptar a mudanças contextuais.	Enfrenta desafios éticos significativos, como a criação de <i>deepfakes</i> ; a qualidade do conteúdo gerado pode variar.
Desenvolvimento de Algoritmos	Baseado predominantemente em algoritmos de aprendizado de máquina tradicionais, como árvores de decisão e regressão.	Incorpora redes neurais convolucionais ² , redes adversárias generativas (GANs) e outras arquiteturas de aprendizado profundo.
Escalabilidade e Adaptação	Requer ajustes e reestruturação frequentes para escalar ou adaptar-se a novos domínios.	Demonstram uma maior capacidade de adaptação e escalabilidade devido à natureza flexível do aprendizado de máquina.

Fonte: Próprio do autor.

No contexto educacional, a IA Generativa tem o potencial de influenciar positivamente o modo como o ensino de programação é conduzido. Por exemplo, pode-se utilizar essa tecnologia para gerar exercícios de programação dinâmicos e personalizados que se adaptam ao nível de conhecimento dos estudantes. Ferramentas baseadas em IA Generativa, como ChatGPT e *Google's Bard*, podem ser utilizadas para fornecer tutoria instantânea e *feedbacks* construtivos, melhorando significativamente a interação aluno-máquina.

Tanto a IA Tradicional quanto a Generativa têm seus lugares assegurados no futuro do ensino de programação de computadores. Enquanto a primeira oferece uma base sólida e confiável, a segunda abre novas possibilidades para a criatividade e personalização no aprendizado.

² As redes neurais convolucionais (CNNs, do inglês *Convolutional Neural Networks*) são uma classe de redes neurais profundas, comumente aplicadas no processamento e análise de imagens. Elas são especialmente poderosas para tarefas que envolvem reconhecimento de padrões e classificação de imagens, tendo se mostrado eficazes em sistemas de visão computacional. A arquitetura das CNNs é inspirada pela organização do córtex visual dos animais, que apresenta uma estrutura hierárquica de processamento de informações visuais.

3 CHATBOTS, GAMES INTELIGENTES, JUÍZES ONLINE, ASSISTENTES INTELIGENTES E SISTEMAS DE TUTORIA INTELIGENTE

Esta seção explora várias aplicações reais de IA Generativa, incluindo *chatbots*, games inteligentes, juízes online, assistentes inteligentes e sistemas de tutoria inteligente, demonstrando como cada uma contribui positivamente para o ensino de programação.

Os *chatbots*, ao serem utilizados como ferramentas educacionais, desempenham um papel importante ao fornecer suporte interativo ao processo de ensino-aprendizagem. Eles permitem que os alunos interajam dinamicamente com conteúdos específicos de programação e recebam *feedback* instantâneo sobre suas tentativas de codificação. Ademais, os *chatbots* representam um importante recurso na redução da carga de trabalho dos professores, desta forma, enquanto os *chatbots* gerenciam interações mais padronizadas, os educadores podem dedicar mais tempo e energia para desenvolver estratégias pedagógicas inovadoras e para o acompanhamento personalizado dos alunos.

Segundo Yilmaz e Yilmaz (2023), games inteligentes, que incorporam elementos de IA, transformam o ensino de programação em uma experiência mais engajadora e interativa. Tais games adaptam os desafios ao nível de habilidade do jogador, promovendo a aplicação prática de conceitos teóricos em cenários dinâmicos e envolventes. Esta abordagem não só melhora o engajamento dos alunos, mas também reforça a aprendizagem ativa, permitindo uma imersão completa no processo de aprendizagem através da prática contínua e do *feedback* instantâneo.

Sistemas de avaliação de código, também conhecidos como juízes online, tais como *HackerRank* e *LeetCode*, utilizam algoritmos de inteligência artificial para analisar e avaliar a qualidade dos programas escritos pelos usuários. Essas plataformas fornecem *feedback* instantâneo e objetivo, permitindo aos estudantes compreenderem claramente suas deficiências e progressos.

Estas ferramentas oferecem uma extensa coleção de problemas de programação que abordam uma variedade de tópicos como estruturas de dados, algoritmos e técnicas de resolução de problemas. Elas também são conhecidas por seus desafios de codificação e simulados de entrevistas, que ajudam os usuários a se prepararem para as demandas específicas das entrevistas em grandes empresas de tecnologia, frequentemente referidas como *Big Techs*.

Assistentes inteligentes como o *GitHub Copilot* e o *Google Colab* facilitam interações mais ricas e adaptativas. O *GitHub Copilot*, por exemplo, oferece sugestões de código baseadas em comentários e código anterior, enquanto o *Google Colab* permite que os alunos trabalhem com *notebooks* de Python em um ambiente interativo e colaborativo.

Sistemas de Tutoria Inteligente (STIs) como o *Carnegie Learning's MATHia*, por exemplo, utilizam IA para oferecer suporte educacional altamente personalizado em Matemática. Projetado para fornecer instrução personalizada e adaptativa, o MATHia atende às necessidades individuais dos estudantes, enquanto eles exploram conteúdos matemáticos. Esse sistema é especialmente valioso no contexto do ensino de Engenharia, onde os princípios matemáticos são fundamentais.

A utilização da IA Generativa no ensino de programação tem o potencial de transformar a maneira como essa disciplina é ensinada e aprendida. Ferramentas como *chatbots* e sistemas de tutoria inteligente oferecem suporte educacional personalizado, que pode ajudar a superar barreiras no aprendizado de programação, proporcionando respostas rápidas e facilitando a depuração de erros (Surameery e Shakor, 2023).

Contudo, segundo Yilmaz e Yilmaz (2023), a integração da IA no ensino de programação também apresenta desafios significativos, a dependência destas

ferramentas pode também reduzir o desenvolvimento de habilidades de resolução de problemas inerentes à programação, uma vez que estudantes podem se tornar menos propensos a engajar-se profundamente com os problemas se as respostas são prontamente fornecidas por uma IA. Rodrigues e Rodrigues (2023) argumentam que sem regulamentações éticas adequadas e um uso crítico das ferramentas de IA, os estudantes podem enfrentar limitações significativas em sua capacidade de aprender e inovar de maneira autônoma.

De acordo com Rahman e Watanobe (2023), é essencial que políticas eficazes sejam desenvolvidas para apoiar o ensino de programação, levando em conta essas complexidades introduzidas pelo uso de ferramentas de IA. Isso inclui garantir que os alunos sejam capacitados para utilizar essas ferramentas de maneira crítica, promovendo um equilíbrio entre o uso de tecnologia e o desenvolvimento de habilidades analíticas e criativas necessárias para a programação de computadores.

4 O CHATGPT NAS AULAS DE TÉCNICAS DE PROGRAMAÇÃO

Nesta seção, descreve-se detalhadamente a pesquisa realizada em sala de aula com o uso da IA Generativa, especificamente o ChatGPT, na disciplina de Técnicas de Programação. A investigação adotou um desenho experimental controlado, no qual os participantes foram distribuídos aleatoriamente em dois grupos: um grupo de controle e um grupo experimental.

O grupo de controle seguiu o currículo tradicional da disciplina, utilizando métodos convencionais de ensino que incluem leituras, exemplos práticos e exercícios de programação, mas sem o auxílio de ferramentas de IA Generativa. Este grupo serviu como uma base de comparação para avaliar os benefícios da introdução de tecnologias baseadas em IA no processo de ensino-aprendizagem.

Por outro lado, o grupo experimental teve uma abordagem diferenciada, integrando o ChatGPT como uma ferramenta de suporte ao aprendizado. Os estudantes deste grupo utilizaram o ChatGPT para obter explicações personalizadas e *feedback* imediato sobre problemas de programação, além de terem acesso a tutoriais e sessões de perguntas e respostas dinâmicas conduzidas pela IA. A integração dessa tecnologia visava não apenas facilitar a compreensão de conceitos complexos e a aplicação prática deles, mas também estimular a capacidade dos alunos de resolver problemas de forma autônoma e criativa.

Em uma das atividades propostas para o grupo experimental, os estudantes enfrentaram o desafio de desenvolver um código em Python para simular o comportamento de um oscilador amortecido, modelado pela equação diferencial $\frac{d^2x}{dt^2} + 2\zeta\omega_0 \frac{dx}{dt} + \omega_0^2 x = 0$. Este problema foi introduzido aos alunos, que deveriam programar uma função que simulasse a dinâmica do oscilador ao longo do tempo com condições iniciais e parâmetros específicos como ζ , ω_0 , $x(0)$ e $\frac{dx}{dt}(0)$.

Inicialmente, os alunos solicitaram ao ChatGPT um exemplo de código que implementasse a solução numérica da equação do oscilador usando o método de Euler. O ChatGPT forneceu um código básico, porém funcional, conforme pode-se observar na Figura 1.

Figura 1 – Solução numérica da equação do oscilador amortecido utilizando o método de Euler.

```
import numpy as np
import matplotlib.pyplot as plt

def oscilador_amortecido(zeta, omega0, x0, v0, t_final, dt):
    t = np.arange(0, t_final, dt)
    x = np.zeros(len(t))
    v = np.zeros(len(t))
    x[0] = x0
    v[0] = v0

    for i in range(1, len(t)):
        v[i] = v[i-1] + (-2*zeta*omega0*v[i-1] - omega0**2*x[i-1]) * dt
        x[i] = x[i-1] + v[i-1] * dt

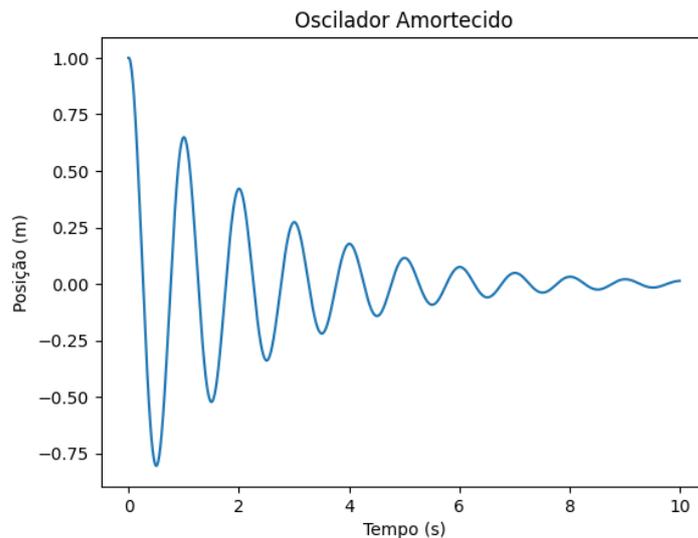
    plt.plot(t, x)
    plt.title('Oscilador Amortecido')
    plt.xlabel('Tempo (s)')
    plt.ylabel('Posição (m)')
    plt.show()

oscilador_amortecido(0.1, 2*np.pi, 1, 0, 10, 0.01)
```

Fonte: Próprio do autor.

No entanto, ao testarem o código, os alunos notaram instabilidades nos resultados, observadas na Figura 2, que sugestionavam uma divergência numérica.

Figura 2 – Gráfico do oscilador amortecido utilizando o método de Euler.



Fonte: Próprio do autor.

Os estudantes então recorreram ao ChatGPT para entender os erros, e a ferramenta sugeriu a adoção do método de Euler-Cromer³ para aumentar a estabilidade do sistema, uma modificação que eles rapidamente implementaram, anteriormente, a posição $x[i]$ era atualizada usando a velocidade no início do intervalo $v[i-1]$, agora, a partir do novo método, a posição $x[i]$ passou a ser atualizada usando a nova velocidade calculada $v[i]$, como pode-se observar na Figura 3.

Figura 3 – Solução numérica da equação do oscilador amortecido utilizando o método de Euler-Cromer.

```
def oscilador_amortecido_ec(zeta, omega0, x0, v0, t_final, dt):
    t = np.arange(0, t_final, dt)
    x = np.zeros(len(t))
    v = np.zeros(len(t))
    x[0] = x0
    v[0] = v0

    for i in range(1, len(t)):
        v[i] = v[i-1] + (-2*zeta*omega0*v[i-1] - omega0**2*x[i-1]) * dt
        x[i] = x[i-1] + v[i] * dt

    plt.plot(t, x)
    plt.title('Oscilador Amortecido (Método Euler-Cromer)')
    plt.xlabel('Tempo (s)')
    plt.ylabel('Posição (m)')
    plt.show()

oscilador_amortecido_ec(0.1, 2*np.pi, 1, 0, 10, 0.01)
```

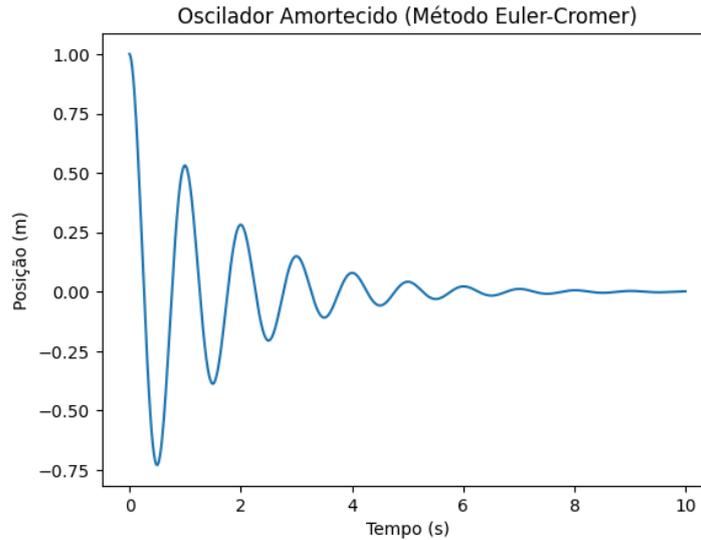
Fonte: Próprio do autor.

Com o código agora estabilizado, os estudantes executaram o programa utilizando diferentes valores para os parâmetros ζ e ω_0 , observando como essas variações afetavam o comportamento do oscilador amortecido. Durante essa fase de refinamento e aprendizado, eles continuaram a utilizar o ChatGPT para obter *feedback* rápido e discutir os resultados obtidos, permitindo-lhes uma melhor compreensão da dinâmica subjacente ao modelo.

O código revisado utilizou a nova velocidade para atualizar a posição, um ajuste que melhorou significativamente a precisão dos resultados, como pode-se observar na Figura 4.

³ O método de Euler-Cromer é uma variação do método de Euler explícito, que é utilizado para resolver equações diferenciais numéricas. Ambos os métodos pertencem à categoria dos métodos de integração numérica de primeira ordem, mas o método de Euler-Cromer oferece certas vantagens em termos de estabilidade, especialmente para sistemas oscilatórios como os encontrados em problemas de Física.

Figura 4 – Gráfico do oscilador amortecido utilizando o método de Euler-Cromer.



Fonte: Próprio do autor.

Esta atividade não apenas reforçou o conhecimento dos alunos sobre métodos numéricos e modelagem de sistemas físicos, mas também demonstrou a eficiência do ChatGPT como uma ferramenta de ensino interativa e adaptativa que enriqueceu a experiência de aprendizado.

Ao longo do semestre, foram coletados dados sobre o desempenho dos alunos em várias avaliações, incluindo testes escritos, projetos de programação e pesquisas de satisfação. Além disso, foram analisadas métricas relacionadas ao engajamento dos estudantes e ao desenvolvimento de habilidades específicas de programação. A pesquisa buscou identificar não apenas o desempenho do uso do ChatGPT em melhorar os resultados de aprendizagem, mas também observar possíveis mudanças nas atitudes e na autopercepção dos alunos em relação à programação.

Com o objetivo de avaliar a percepção dos estudantes sobre a utilização da IA Generativa em sua experiência de aprendizado, as pesquisas de satisfação foram elaboradas com uma série de perguntas focadas em diversos aspectos. Questões como "Como você avalia a eficácia do ChatGPT em ajudá-lo a compreender conceitos de programação?", "O uso do ChatGPT influenciou sua motivação para aprender programação?" e "Você sentiu que o *feedback* fornecido pelo ChatGPT foi claro e útil?" foram incluídas para medir a utilidade percebida da ferramenta. Além disso, perguntou-se aos alunos sobre a facilidade de uso do ChatGPT e se eles recomendariam sua utilização a outros estudantes. Essas questões foram formuladas para coletar tanto a satisfação geral com a ferramenta quanto os possíveis benefícios educacionais e desafios enfrentados.

5 RESULTADOS DA PESQUISA

A metodologia adotada para esta pesquisa permitiu uma análise comparativa entre os dois grupos, oferecendo resultados relevantes sobre os impactos positivos e os desafios associados ao uso da IA Generativa no ensino de programação de computadores. Durante o semestre, foram coletados dados de desempenho dos estudantes em diversas avaliações, incluindo testes escritos e projetos de programação,

além de pesquisas de satisfação que focaram em aspectos como compreensão de conceitos e motivação.

Em termos de resultados acadêmicos, observou-se uma melhoria significativa na compreensão de conceitos de programação e na habilidade de aplicar esses conceitos em problemas práticos. Estatisticamente, os estudantes do grupo experimental, que utilizaram o ChatGPT, mostraram um aumento de aproximadamente 30% na precisão de seus códigos em comparação com o grupo de controle. Entretanto, um achado particularmente interessante foi que, apesar dos benefícios observados em várias dimensões avaliadas, não foram identificadas diferenças estatisticamente significativas na motivação dos estudantes ao enfrentarem desafios de programação mais complexos entre os grupos controle e experimental.

As respostas dos estudantes às pesquisas de satisfação refletiram uma visão amplamente positiva sobre a utilização do ChatGPT como ferramenta de apoio ao aprendizado de programação. Na questão sobre a eficácia do ChatGPT em ajudar a compreender conceitos de programação, 85% dos alunos do grupo experimental relataram uma melhoria significativa em sua capacidade de compreender e aplicar novos conceitos de programação devido às explicações detalhadas fornecidas pela IA. Quando questionados sobre a clareza do *feedback* recebido, 78% dos participantes concordaram que as orientações fornecidas pelo ChatGPT foram claras e extremamente úteis, especialmente na identificação e correção de erros.

No entanto, em relação à motivação para aprender programação, as respostas foram mais variadas. Aproximadamente 60% dos estudantes mencionaram não sentir uma mudança significativa em sua motivação ao resolver problemas mais complexos, sugerindo que, enquanto o ChatGPT auxiliou no aspecto técnico, ele não impactou substancialmente a motivação intrínseca dos alunos frente a maiores desafios. Cerca de 90% dos alunos afirmaram que encontraram facilidade no uso do ChatGPT e que recomendariam sua utilização a outros estudantes.

Esses resultados sugerem que, enquanto o ChatGPT é uma ferramenta eficiente para melhorar diversos aspectos do aprendizado de programação — tais como a compreensão de novos conceitos e a correção de erros de código — ele não substitui integralmente a necessidade de estratégias motivacionais, que são essenciais para engajar os alunos em tarefas desafiadoras. A investigação aponta para a necessidade de os educadores implementarem estratégias de ensino que não apenas integrem tecnologias avançadas, mas também promovam um ambiente de aprendizado que estimule ativamente o interesse e o envolvimento dos alunos com problemas mais complexos.

6 CONSIDERAÇÕES FINAIS E PERSPECTIVAS FUTURAS

A conclusão extraída desta pesquisa revela que o uso de ferramentas de Inteligência Artificial Generativa, como o ChatGPT, por exemplo, para o ensino de programação de computadores em cursos superior de Computação, Engenharia e áreas correlatas apresenta resultados promissores. O estudo, que utilizou uma estrutura experimental com grupos de controle e teste, demonstrou que os estudantes que utilizaram o ChatGPT tiveram um incremento nas habilidades de pensamento computacional e na motivação para as aulas de programação.

O uso do ChatGPT contribuiu para melhorias em diversas dimensões, incluindo criatividade, pensamento algorítmico, cooperação, pensamento crítico e resolução de problemas. Este resultado sugere que a integração de ferramentas de IA Generativa no processo educacional não apenas facilita a aprendizagem de tarefas de programação,

mas também melhora a atitude individual e as expectativas dos alunos em relação ao aprendizado de programação.

Por outro lado, o estudo ressaltou que, embora se tenham observado benefícios em várias das dimensões avaliadas, não se identificaram diferenças significativas na motivação dos estudantes frente a desafios de programação mais complexos entre os grupos controle e experimental. Isso sugere que, apesar de o ChatGPT ser uma ferramenta eficiente para diversos aspectos do aprendizado de programação, ele não substitui integralmente a necessidade de estratégias motivacionais, que devem ser implementadas pelos educadores para estimular o engajamento dos alunos em tarefas mais desafiadoras.

Considerando os resultados obtidos nesta pesquisa, futuros estudos poderiam investigar a aplicação da IA Generativa em outras disciplinas das áreas de Engenharia e Computação, com o objetivo de expandir o uso dessas tecnologias em campos que exigem intenso raciocínio lógico e analítico. Seria também benéfico avaliar os impactos a longo prazo dessas ferramentas no desempenho acadêmico e na capacidade de inovação dos estudantes. Adicionalmente, examinar como diferentes perfis de estudantes respondem a essas inovações poderia orientar o desenvolvimento de métodos de ensino mais personalizados e eficientes.

REFERÊNCIAS

ELKHODR, Mahmoud *et al.* ICT students' perceptions towards ChatGPT: An experimental reflective lab analysis. **STEM Education**, v. 3, n. 2, p. 70-88, 2023. DOI: <https://www.aimspress.com/article/doi/10.3934/steme.2023006>.

GOODFELLOW, Ian J. *et al.* Generative adversarial nets. **Advances in Neural Information Processing Systems (NIPS 2014)**. Montreal, QC, p. 2672-2680, 2014. DOI: <https://doi.org/10.48550/arXiv.1406.266>.

IQBAL, Nayab; AHMED, Hassaan; AZHAR, Kaukab Abid. Exploring teachers' attitudes towards using Chat GPT. **Global Journal for Management and Administrative Sciences**, v. 3, n. 4, p. 97-111, 2022. DOI: <https://doi.org/10.46568/gjmas.v3i4.163>.

MENTA, Eziquiel; BRITO, Glauca da Silva. O Papel da Inteligência Artificial no Ensino Tecnológico: Implicações Emergentes. **Educitec - Revista de Estudos e Pesquisas sobre Ensino Tecnológico**, Manaus, v. 10, 2024. DOI: <https://doi.org/10.31417/educitec.v10.2325>.

PARK, Sung-Wook *et al.* Review on Generative Adversarial Networks: Focusing on Computer Vision and Its Applications. **Electronics**, v. 10, n. 10, p. 1216, 2021. DOI: <https://doi.org/10.3390/electronics10101216>.

RAHMAN, Mostafizer; WATANOBE, Yutaka. ChatGPT for education and research: Opportunities, threats, and strategies. **Applied Sciences**, v. 13, n. 9, 2023. DOI: <https://doi.org/10.3390/app13095783>.

RODRIGUES, Olira Saraiva; RODRIGUES, Karoline Santos. A inteligência artificial na educação: os desafios do ChatGPT. **Texto Livre**, v. 16, 2023. DOI: <https://doi.org/10.1590/1983-3652.2023.45997>.

SURAMEERY, Nigar M. Shafiq; SHAKOR, Mohammed Y. Use ChatGPT to Solve Programming Bugs. **International Journal of Information Technology and Computer Engineering**, v. 03, n. 01, 2023. DOI: <https://doi.org/10.55529/ijitc.31.17.22>.

YILMAZ, Ramazan; YILMAZ, Fatma Gizem Karaoglan. Augmented intelligence in programming learning: Examining student views on the use of ChatGPT for programming learning. **Computers in Human Behavior: Artificial Humans**, v. 01, 2023. DOI: <https://doi.org/10.1016/j.chbah.2023.100005>.

APPLICATIONS OF GENERATIVE AI IN TEACHING PROGRAMMING FOR ENGINEERING AND COMPUTER SCIENCE COURSES: AN INNOVATIVE APPROACH TO DEVELOPING CODING SKILLS

Abstract: *The implementation of Generative Artificial Intelligence (GenAI) in computer programming education emerges as a promising solution to the challenges found in Engineering, Computing, and related fields, characterized by high dropout and failure rates. This article explores the integration of Adaptive AI technologies, including chatbots, intelligent games, online judges, intelligent assistants, and intelligent tutoring systems. The analysis focuses on the impact of these tools on the personalization of learning, student motivation, and the effectiveness in acquiring programming skills, proposing a more adaptive and interactive teaching paradigm.*

Keywords: *Generative Artificial Intelligence, Programming Education, Adaptive Technologies, Learning Personalization, Student Engagement.*

