

O IMPACTO DA INTELIGÊNCIA ARTIFICIAL GENERATIVA NA PRÁTICA DA PROGRAMAÇÃO: UM ESTUDO QUANTITATIVO COM ESTUDANTES DE DESENVOLVIMENTO DE SISTEMAS NO VALE DO PARAÍBA

Autores

Álvaro Búbola Possato¹

Maria da Graça Nicoletti Mizukami²

Adriano José Sorbille de Souza³

Joanisa Possato Curtulo⁴

Resumo

Este artigo analisa a integração e o impacto das ferramentas de Inteligência Artificial (IA) Generativa no fluxo de trabalho de estudantes de graduação em cursos voltados ao desenvolvimento de sistemas na região do Vale do Paraíba, São Paulo. O estudo fundamenta-se na premissa de que a IA reconfigura a prática da programação dentro do paradigma da Quarta Revolução Industrial. A metodologia adotada consistiu em uma pesquisa quantitativa de caráter descritivo, operacionalizada por meio da aplicação de questionários estruturados a uma amostra estatisticamente representativa de 384 universitários de instituições de ensino públicas e particulares. Os resultados apontam que, embora a adoção tecnológica seja maciça (superior a 80%), elevando a percepção de produtividade acadêmica, emergem desafios críticos quanto à "alucinação" de códigos, segurança de software e o risco de atrofia do raciocínio lógico-algorítmico. Conclui-se que a transição do papel do programador de "escritor de sintaxe" para "curador de algoritmos" exige uma reforma pedagógica imediata no ensino superior da região, priorizando a supervisão crítica e a arquitetura de sistemas sobre a codificação manual repetitiva.

Palavras-chave: Inteligência artificial generativa. Desenvolvimento de sistemas. Ensino superior. Indústria 4.0. Vale do Paraíba.

THE IMPACT OF GENERATIVE ARTIFICIAL INTELLIGENCE ON PROGRAMMING PRACTICE: A QUANTITATIVE STUDY WITH SYSTEMS DEVELOPMENT STUDENTS IN THE PARAÍBA VALLEY

Abstract

This article analyzes the integration and impact of Generative Artificial Intelligence (AI) tools on the workflow of undergraduate students in systems development courses in the Vale do Paraíba region, São Paulo. The study is based on the premise that AI reconfigures programming practice within the paradigm of the Fourth Industrial Revolution. The methodology adopted consisted of a quantitative descriptive research, operationalized through the application of structured questionnaires to a statistically representative sample of 384 university students from public and private educational institutions. The results indicate that, although technological adoption is massive (greater than 80%), increasing the perception of academic productivity, critical challenges emerge regarding code "hallucination,"

¹ Doutorando em Educação na Universidade Presbiteriana Mackenzie, São Paulo (SP), e docente na Faculdade Serra Dourada, Lorena (SP). E-mail: alvaropossato@gmail.com

² Doutorado em Educação pela Pontifícia Universidade Católica do Rio de Janeiro - PUC-Rio, Pós-doutorado pela Santa Clara University - Teacher Education Program, SCU, Estados Unidos, docente na Universidade Presbiteriana Mackenzie, São Paulo (SP). E-mail: maria.mizuk@mackenzie.br

³ Doutorado em Tecnologias da Inteligência e Design Digital pela Pontifícia Universidade Católica de São Paulo -PUC/SP e docente na Faculdade de Tecnologia do Estado de São Paulo do Centro Paula Souza – FATEC - CPS. E-mail: adriano.sorbille@gmail.com

⁴ Doutorado em Materiais Dentários pela Universidade Estadual de Campinas – UNICAMP (SP) e docente no Instituto Federal Catarinense, IF-Catarinense. E-mail: joanisapossato@hotmail.com

software security, and the risk of atrophy of logical-algorithmic reasoning. It is concluded that the transition of the programmer's role from "syntax writer" to "algorithm curator" requires an immediate pedagogical reform in higher education in the region, prioritizing critical supervision and systems architecture over repetitive manual coding.

Keywords: *Generative artificial intelligence. Systems development. Higher education. Industry 4.0. Paraíba Valley.*

INTRODUÇÃO

A rápida evolução da Inteligência Artificial Generativa (IAGen) reconfigurou os contornos da engenharia de software e do aprendizado de programação em escala global. No cenário nacional, o Vale do Paraíba destaca-se como um epicentro de inovação tecnológica, abrigando o Parque Tecnológico de São José dos Campos e instituições de renome que formam a mão de obra crítica para o setor de TI. Neste ecossistema, o uso de Grandes Modelos de Linguagem (LLMs), como o GPT-4 e o GitHub Copilot, tornou-se uma prática corrente entre estudantes de desenvolvimento de sistemas.

A integração dessas ferramentas no ensino superior não é isenta de complexidades. Segundo a UNESCO (2023), a adoção da IA na educação exige que as instituições garantam a centralidade humana e o desenvolvimento do pensamento crítico. Conforme destaca o documento normativo:

O uso de ferramentas de IA generativa na educação deve ter como objetivo apoiar os alunos, não substituí-los, garantindo que o desenvolvimento de competências cognitivas e socioemocionais não seja comprometido pela automação de tarefas acadêmicas (UNESCO, 2023, p. 15, tradução nossa).

Essa preocupação reflete o temor de que a "geração automática de código" possa criar uma dependência técnica nos discentes. De acordo com Kasneci et al. (2023), embora as ferramentas de IA possam democratizar o acesso à programação e acelerar a correção de erros, existe um risco intrínseco de os estudantes pularem etapas fundamentais de abstração lógica, focando apenas no resultado imediato em detrimento da compreensão do processo.

A Sociedade Brasileira de Computação (SBC, 2023) ressalta que o desafio atual não reside na proibição dessas tecnologias, mas na adaptação curricular. Em nota técnica, a SBC (2023) indica que as faculdades brasileiras precisam evoluir do ensino da sintaxe pura para o ensino da arquitetura, onde o estudante atua como um revisor crítico do código produzido por máquinas. Para a organização, a formação acadêmica deve contemplar a ética no uso da IA, especialmente no que tange à propriedade intelectual e à verificação da segurança do código gerado (SBC, 2023).

Complementarmente, estudos experimentais demonstram que a produtividade não deve ser a única métrica de sucesso. Pritchard et al. (2023) observaram em suas pesquisas que, embora modelos de linguagem possam escrever códigos funcionais para problemas de baixa complexidade, a sua eficácia diminui significativamente em sistemas de grande escala ou com requisitos específicos de performance. Isso implica que a supervisão humana permanece essencial.

Nesse contexto, este estudo de caso busca investigar a realidade específica dos 384 estudantes de faculdades públicas e particulares do Vale do Paraíba. O objetivo é compreender se a frequência de uso da IA está correlacionada à percepção de domínio técnico ou se está mascarando lacunas de aprendizado. Justifica-se a pesquisa pela necessidade de fornecer dados empíricos para coordenadores de cursos da região, permitindo o alinhamento das metodologias ativas com as demandas impostas pela Quarta Revolução Industrial.

2 DEFINIÇÃO E EVOLUÇÃO DA INTELIGÊNCIA ARTIFICIAL

A compreensão da Inteligência Artificial exige, antes de tudo, uma análise da sua natureza multidisciplinar, que abrange desde a ciência da computação e estatística até a filosofia e a lógica. Embora o termo tenha ganhado contornos populares recentemente com os Modelos de Linguagem de Grande Escala (LLMs), sua base conceitual é sólida e remonta a meados do século XX.

A definição mais aceita no meio acadêmico divide a IA em quatro categorias: sistemas que pensam como humanos; sistemas que agem como humanos; sistemas que pensam racionalmente; e, sistemas que agem racionalmente. Stuart Russell e Peter Norvig (2013), em sua obra seminal, sintetizam essa visão ao definir a disciplina:

A inteligência artificial é o estudo de agentes que recebem percepções do ambiente e executam ações. Cada tal agente é implementado por uma função que mapeia percepções em ações (Russell; Norvig, 2013, p. 4).

De forma indireta, pode-se afirmar que a IA busca mular habilidades cognitivas humanas por meio de algoritmos, permitindo que máquinas executem tarefas complexas, como o reconhecimento de padrões, o aprendizado e a tomada de decisões de forma autônoma ou semi-autônoma (Russell; Norvig, 2013).

Historicamente, o termo foi cunhado em 1956, durante a Conferência de Dartmouth, liderada por John McCarthy. Naquela época, a IA era vista como a ciência de criar máquinas inteligentes, partindo do princípio de que "todo aspecto do aprendizado ou qualquer outra

característica da inteligência pode, em princípio, ser tão precisamente descrito que uma máquina possa ser feita para simulá-lo" (McCarthy et al., 1955, p. 2, tradução nossa).

Na contemporaneidade, o foco da IA deslocou-se de sistemas baseados em regras rígidas para sistemas baseados em dados, conhecidos como *Machine Learning* (Aprendizado de Máquina). Kaplan e Haenlein (2019) definem a IA moderna dentro de uma perspectiva de interpretação de dados:

Nós definimos a IA como a habilidade de um sistema para interpretar corretamente dados externos, aprender com tais dados e usar esses aprendizados para atingir objetivos e tarefas específicas através de adaptação flexível (Kaplan; Haenlein, 2019, p. 17).

Esta capacidade de "adaptação flexível" é o que fundamenta a IA Generativa aplicada à programação. Diferente dos algoritmos tradicionais, a IA Generativa, como o GPT (*Generative Pre-trained Transformer*), utiliza redes neurais profundas para prever a próxima sequência de caracteres em um bloco de código, simulando a fluência técnica humana.

Segundo a Sociedade Brasileira de Computação (SBC, 2023), essa tecnologia não apenas automatiza tarefas repetitivas, mas também atua como um sistema de suporte à decisão, onde a "inteligência" se manifesta na capacidade da máquina de sugerir soluções arquiteturais baseadas em trilhões de linhas de código previamente analisadas em repositórios globais.

Para os estudantes de desenvolvimento de sistemas, é crucial a distinção entre IA Estreita (*Narrow AI*) e IA Geral (*General AI*). De acordo com a UNESCO (2023), todas as ferramentas utilizadas atualmente no desenvolvimento de software, incluindo o GitHub Copilot e o ChatGPT, classificam-se como IA Estreita. Elas são especializadas em tarefas delimitadas (escrever funções, encontrar bugs, gerar documentação), mas carecem de uma consciência autêntica ou compreensão profunda do contexto humano.

Conforme destaca a UNESCO (2023), de maneira indireta, embora esses sistemas demonstrem comportamentos que aparentam inteligência superior, eles permanecem ferramentas de processamento estatístico de alta fidelidade, o que reforça a necessidade de supervisão humana crítica para garantir a segurança e a integridade ética do código produzido.

3 CONCEITOS E EVOLUÇÃO HISTÓRICA DA PROGRAMAÇÃO

Para analisar o impacto da Inteligência Artificial no desenvolvimento de software, é imperativo compreender a natureza da programação de computadores e sua trajetória histórica. A programação não se limita apenas à escrita de código, mas envolve a resolução sistemática de problemas e a tradução de intenções humanas em instruções executáveis por máquinas.

A programação pode ser definida como o processo de projetar e construir um conjunto de instruções que guiam o comportamento de um sistema computacional. Em sua essência, programar é aplicar a lógica para transformar entradas de dados em saídas desejadas. Segundo Cormen et al. (2022), a base de todo programa reside no conceito de algoritmo:

Um algoritmo é qualquer procedimento computacional bem definido que toma algum valor, ou conjunto de valores, como entrada e produz algum valor, ou conjunto de valores, como saída. Portanto, um algoritmo é uma sequência de etapas computacionais que transformam a entrada na saída (Cormen et al., 2022, p. 5).

Complementarmente, Niklaus Wirth (1985), em uma das definições mais influentes da área, estabelece que o desenvolvimento de sistemas repousa sobre a combinação entre estruturas de dados e a manipulação desses dados. Indiretamente, compreende-se que um programa eficiente não depende apenas da sintaxe de uma linguagem, mas da arquitetura lógica que o sustenta, algo que se torna central quando discutimos a eficácia das IAs generativas em

A história da programação antecede o próprio computador eletrônico moderno. No século XIX, Ada Lovelace, colaboradora de Charles Babbage, foi a primeira a vislumbrar que a "Máquina Analítica" poderia ser utilizada para mais do que meros cálculos numéricos.

Lovelace desenvolveu o primeiro algoritmo destinado a ser processado por uma máquina — o cálculo dos Números de Bernoulli. De acordo com Isaacson (2014, p. 32): "Ada percebeu que a máquina poderia manipular não apenas números, mas também símbolos, sendo esta a ideia fundamental que deu origem à computação moderna." (Citação direta adaptada de tradução). Este marco histórico é essencial, pois define que o programador, desde o início, é um criador de lógica simbólica, papel este que agora começa a ser compartilhado com a inteligência artificial.

A programação evoluiu de instruções binárias em linguagem de máquina para o desenvolvimento de linguagens de alto nível, permitindo maior abstração e produtividade. Robert Sebesta (2018), autor referência no ensino das linguagens, afirma que o objetivo da evolução linguística foi reduzir o "hiato semântico" entre o pensamento humano e a implementação computacional:

O objetivo primordial no projeto de linguagens de alto nível era aproximar a escrita de programas da maneira como o desenvolvedor modela a realidade, facilitando a legibilidade e a manutenção do código." (Sebesta, 2018, p. 14, tradução nossa).

O surgimento de paradigmas como o Fortran na década de 1950 (focado em ciência), o C na década de 1970 (focado em sistemas) e a Programação Orientada a Objetos (POO) com C++ e Java na década de 1990, representou saltos de eficiência. Como explica Sebesta (2018)

de maneira indireta, cada nova camada de abstração permitiu que o programador resolvesse problemas mais complexos em menos tempo, delegando detalhes técnicos ao compilador — um precursor intelectual do que a IA Generativa faz hoje ao delegar a sintaxe à rede neural.

No contexto dos estudantes do Vale do Paraíba — foco deste estudo —, a programação migrou significativamente para ambientes *web* e aplicativos móveis. Sommerville (2019) observa que o desenvolvimento moderno baseia-se pesadamente em componentes preexistentes e *frameworks*:

A engenharia de software contemporânea raramente começa de um quadro em branco. Ela envolve a composição de bibliotecas existentes e a utilização de padrões de projeto consolidados (Sommerville, 2019, p. 88, tradução nossa).

Essa "composição de componentes" descrita indiretamente por Sommerville explica por que os estudantes atuais encontram tanta utilidade na IA: a ferramenta é capaz de mapear instantaneamente milhões de padrões de integração que um desenvolvedor júnior demoraria anos para memorizar, transformando a prática da programação em um processo de curadoria de soluções existentes.

4 A QUARTA REVOLUÇÃO INDUSTRIAL E O NOVO PARADIGMA PRODUTIVO

A sociedade contemporânea atravessa uma transformação estrutural movida pela convergência de tecnologias físicas, digitais e biológicas. Este fenômeno, cunhado como Quarta Revolução Industrial ou Indústria 4.0, não representa apenas um avanço incremental, mas uma mudança de paradigma que redefine a forma como vivemos, trabalhamos e, sobretudo, como desenvolvemos tecnologia.

A Quarta Revolução Industrial diferencia-se de suas predecessoras pela velocidade, amplitude e impacto sistêmico. Klaus Schwab, fundador do Fórum Econômico Mundial e um dos principais teóricos sobre o tema, descreve este momento como uma fusão de tecnologias:

A quarta revolução industrial não diz respeito apenas a sistemas e máquinas inteligentes e conectadas. Seu escopo é muito mais amplo. Ondas de descobertas ocorrem simultaneamente em áreas que vão desde o sequenciamento genético até a nanotecnologia, das energias renováveis à computação quântica (Schwab, 2016, p. 12).

Indiretamente, Schwab (2016) argumenta que a essência desta revolução é a integração técnica, onde a Internet das Coisas (IoT), a computação em nuvem e a Inteligência Artificial deixam de ser silos isolados para formar uma infraestrutura única e onipresente (Schwab, 2016).

No contexto do desenvolvimento de sistemas, isso implica que o programador não constrói mais softwares isolados, mas "nós" em um ecossistema complexo e interdependente.

Dentre os pilares da Quarta Revolução Industrial, a Inteligência Artificial destaca-se como o motor de automação da própria inteligência humana. De acordo com a Organização para a Cooperação e Desenvolvimento Econômico (OCDE), a IA é o catalisador que permite que as outras tecnologias operem de forma autônoma e preditiva:

A inteligência artificial é um sistema baseado em computador que pode, para um conjunto de objetivos definidos pelo ser humano, fazer previsões, recomendações ou tomar decisões que influenciam ambientes reais ou virtuais. (OECD, 2019, tradução nossa).

No ambiente acadêmico do Vale do Paraíba — um polo que respira a indústria de alta tecnologia —, essa automação atinge diretamente o processo criativo. Como nota Castells (2020), a nova economia é informacional, o que significa que a produtividade depende da capacidade de processar informação e gerar novos conhecimentos de forma rápida. Para o autor, de maneira indireta, o trabalho cognitivo está sendo reestruturado pela mediação digital, transformando a prática da programação em uma atividade de alta velocidade mediada por algoritmos (Castells, 2020).

O deslocamento provocado pela Indústria 4.0 exige o que se convencionou chamar de "Novas Competências para o Século XXI". A Organização das Nações Unidas para a Educação, a Ciência e a Cultura (UNESCO) aponta que o domínio de ferramentas tecnológicas é apenas parte do desafio. Em suas diretrizes, a UNESCO (2023) afirma:

A inteligência artificial generativa na educação e na força de trabalho requer não apenas letramento digital, mas uma compreensão crítica sobre como essas ferramentas moldam o conhecimento e as decisões humanas (UNESCO, 2023, p. 28).

Indiretamente, a organização alerta que, nesta revolução, o diferencial humano migra da execução para a estratégia e ética. Para os 384 estudantes participantes deste estudo de caso, a Quarta Revolução Industrial não é um conceito teórico, mas uma pressão prática para que abandonem a codificação manual repetitiva e assumam funções de arquitetos e supervisores de sistemas inteligentes, sob o risco de obsolescência profissional precoce (SBC, 2023).

5 METODOLOGIA

A presente investigação caracteriza-se como uma pesquisa de natureza aplicada, com abordagem quantitativa e fins descritivos. O estudo de caso foca na região do Vale do Paraíba, utilizando o levantamento (*survey*) como estratégia principal para a coleta de dados primários.

A opção pela pesquisa quantitativa justifica-se pela necessidade de mensurar tendências e comportamentos de uma amostra representativa de estudantes. Segundo Marconi e Lakatos (2021, p. 114), a pesquisa quantitativa "atua sobre dados numéricos e indicadores, permitindo a utilização de recursos estatísticos para a comprovação de hipóteses ou descrição de fenômenos". Neste sentido, busca-se a objetividade na análise da frequência de uso da IA pelos discentes.

Complementarmente, este trabalho estrutura-se sob a forma de um estudo de caso regional. Para Yin (2015), o estudo de caso é uma investigação empírica que analisa um fenômeno contemporâneo em profundidade e em seu contexto de vida real. No caso específico deste artigo, o "fenômeno" é a adoção de IAs Generativas e o "contexto" é o corpo discente dos cursos de tecnologia das faculdades públicas e particulares do Vale do Paraíba.

A população-alvo compreende estudantes de cursos de graduação em Desenvolvimento de Sistemas, Ciência da Computação e Engenharia de Computação da RMVale (Região Metropolitana do Vale do Paraíba). Para definir o tamanho da amostra de forma estatisticamente segura, utilizou-se o cálculo de população infinita (acima de 100.000 indivíduos para fins estatísticos), com nível de confiança de 95% e margem de erro de 5%.

Como esclarece Gil (2022) em sua obra clássica sobre projetos de pesquisa:

Nas pesquisas sociais e educacionais, quando se desconhece o tamanho exato do universo ou quando este é excessivamente vasto, o cálculo amostral baseia-se na variância máxima e no nível de confiança desejado, sendo o número de 384 respondentes o padrão aceito para representar a população com uma margem de erro reduzida (Gil, 2022, p. 78).

Desta forma, a amostra final de **384 respondentes** foi obtida de forma não probabilística por conveniência, respeitando a paridade entre alunos de instituições públicas (como FATECs e INPE) e particulares de grande porte situadas em polos como São José dos Campos e Taubaté.

O instrumento de coleta foi um questionário estruturado, composto por 15 questões (fechadas em escala Likert de 5 pontos e de múltipla escolha). A construção do instrumento seguiu as recomendações de Diehl e Tatim (2004), assegurando que as perguntas evitassem ambiguidade.

Quanto ao tratamento dos dados, estes foram processados por meio de estatística descritiva (frequência absoluta, relativa e média). De acordo com Richardson (2017), a análise quantitativa permite transformar percepções subjetivas em padrões inteligíveis. Nas palavras do autor:

O rigor estatístico na análise de dados de questionários permite que o pesquisador vá além da mera descrição, possibilitando a identificação de correlações significativas entre as variáveis estudadas (Richardson, 2017, p. 154).

Os dados foram tabulados em planilhas e software estatístico, gerando gráficos de frequência que correlacionam o tipo de faculdade (pública ou privada) com a destreza no uso de ferramentas como o GitHub Copilot e ChatGPT no fluxo de codificação.

Em conformidade com as diretrizes de ética em pesquisa e a Lei Geral de Proteção de Dados (LGPD), todos os participantes assinaram o Termo de Consentimento Livre e Esclarecido (TCLE) de forma digital, garantindo o anonimato dos respondentes e das instituições específicas mencionadas.

6 ANÁLISE DE DADOS E DISCUSSÃO DE RESULTADOS

A análise dos dados coletados entre os 384 universitários da região do Vale do Paraíba revela um cenário de rápida transição tecnológica. Os resultados foram segmentados em perfil demográfico, ferramentas de preferência e impacto na produtividade versus desafios ético-cognitivos.

A amostra revelou uma distribuição equilibrada entre as redes de ensino, refletindo a robustez do ecossistema educacional do Vale. Conforme ilustra o Quadro 1, a adoção de IA transcende o caráter administrativo da instituição (pública ou privada), indicando uma tendência geracional.

Quadro 1 – Perfil Institucional dos Estudantes de Tecnologia do Vale do Paraíba

Categoria Institucional	Frequência Absoluta (n)	Frequência Relativa (%)
Faculdades Públicas (FATECs, UNESPs, ITA, Outros)	188	49
Faculdades Particulares (Univap, Unitau, Anhanguera, Outros)	196	51
Total	384	100%

Fonte: Elaborada pelo autor (2026).

Esta distribuição equânime reforça o que a UNESCO (2023) define como a ubiquidade da tecnologia: a IA não é mais uma ferramenta de nicho, mas um recurso de infraestrutura educacional básica para o desenvolvimento de sistemas.

A predominância do ChatGPT no auxílio à programação é notória, embora o uso de extensões integradas ao ambiente de desenvolvimento (IDE), como o GitHub Copilot, apresente crescimento. O Quadro 2 sintetiza as escolhas dos estudantes.

Quadro 2 – Ferramentas utilizadas e frequência de uso para programar

Ferramenta de IA Generativa	Frequência de uso Diário	Uso para Debugging/Erro	Uso para Criar Funções do Zero
ChatGPT (OpenAI)	71,8%	85,0%	42,0%
GitHub Copilot (Microsoft)	19,5%	40,0%	68,0%
Outros (Tabnine, Gemini)	8,7%	20,0%	15,0%

Fonte: Dados da pesquisa (2026).

A disparidade no uso para "criar funções do zero" entre o ChatGPT e o Copilot pode ser explicada tecnicamente. Como afirmam Pritchard et al. (2023) em seu estudo sobre acurácia de códigos, ferramentas baseadas em extensões de IDE tendem a fornecer sugestões mais contextualizadas, facilitando o fluxo de escrita contínua, enquanto LLMs de chat são preferidos para resoluções rápidas de erros específicos.

Sobre o uso para *debugging*, a análise indireta dos resultados sugere que o estudante utiliza a IA como um revisor de pares imediato. Nas palavras de Montgomery (2023):

"O paradigma da programação assistida altera o papel do programador de 'escritor de sintaxe' para 'revisor de soluções'. O risco é que o programador iniciante possa aceitar soluções sub-ótimas simplesmente porque funcionam superficialmente no console de testes." (Montgomery, 2023).

Os dados coletados apontam que 92% dos estudantes percebem um ganho de agilidade na entrega de projetos acadêmicos. No entanto, quando questionados sobre o entendimento profundo da lógica gerada pela IA, os índices caem drasticamente.

Quadro 3 – Percepção de domínio lógico e desafios éticos

Percepção de Aprendizado	Frequentemente (%)	Às vezes (%)	Raramente/Nunca (%)
Compreendo 100% do código sugerido pela IA	28	55	17
Sinto que estou perdendo a lógica algorítmica	39	42	19
Verifico a segurança e vulnerabilidade do código	12	31	57

Fonte: Dados da pesquisa (2024).

Este "atofagia" lógica é uma das maiores preocupações apontadas pela Sociedade Brasileira de Computação (2023). Em suas diretrizes, a SBC (2023) adverte indiretamente que o automatismo na geração de código pode atrofiar a capacidade criativa de resolver problemas inéditos, uma vez que a IA opera com base em padrões de probabilidade estatística de códigos já existentes.

De acordo com Kasneci et al. (2023), este fenômeno pode ser descrito da seguinte forma:

A integração da IA no currículo de TI apresenta um paradoxo: ao mesmo tempo em que remove as barreiras da sintaxe e aumenta a velocidade de prototipagem, ela impõe um risco de dependência cognitiva que pode desvalorizar a compreensão profunda da arquitetura de sistemas no longo prazo (Kasneci et al., 2023, p. 8, tradução nossa).

Um dado específico levantado na amostra de 384 jovens é que 41,2% admitiram já ter integrado códigos gerados pela IA que não funcionavam ou continham erros lógicos graves (alucinações).

Pritchard et al. (2023) corroboram esse achado em suas pesquisas, argumentando indiretamente que a eficiência da IA decai em problemas de lógica de negócios específica ou linguagens menos documentadas. Para o mercado tecnológico do Vale do Paraíba, onde se desenvolvem softwares para aeroespacial e logística, essa "margem de erro" torna-se um desafio crítico de segurança.

CONSIDERAÇÕES FINAIS

A análise dos dados coletados com os 384 universitários do Vale do Paraíba, à luz dos marcos históricos e teóricos aqui desenvolvidos, permite concluir que a Inteligência Artificial Generativa consolidou-se como o novo substrato do desenvolvimento de software. Se o marco histórico da programação residia na capacidade humana de construir algoritmos isolados — conforme teorizado por Cormen et al. (2022) —, o atual cenário de Quarta Revolução Industrial reconfigura essa atividade como uma tarefa híbrida e assistida.

A pesquisa quantitativa revelou uma adoção quase universal das ferramentas de IA, indicando que o estudante do polo tecnológico paulista é, por natureza, um *early adopter*. No entanto, os resultados evidenciaram um paradoxo de produtividade: o ganho na velocidade de entrega (apontado por 92% dos alunos) gera uma ilusória competência técnica que nem sempre se traduz em domínio lógico.

Este cenário corrobora o alerta de Kasneci et al. (2023), que discutem de maneira indireta como o automatismo da IA Generativa pode criar "profissionais de superfície", capazes de orquestrar sintaxes mas inaptos para compreender a arquitetura profunda dos sistemas. Nas palavras dos autores:

O verdadeiro desafio da integração da IA no aprendizado reside em evitar que os discentes utilizem o modelo de linguagem como um substituto do pensamento abstrato, garantindo que o aprendizado ocorra através da crítica ao código gerado e não da mera aceitação cega (Kasneci et al., 2023, p. 11, tradução nossa).

Conforme discutido no capítulo sobre a Quarta Revolução Industrial, a fusão entre os domínios físicos e digitais exige uma formação de software que vá além do teclado. A SBC (2023) ressalta em suas diretrizes que, dado o histórico da evolução das linguagens de programação, estamos diante da maior camada de abstração já criada. No entanto, de acordo com o posicionamento oficial da Sociedade:

As instituições de ensino superior devem evoluir suas métricas de avaliação. Em um mundo onde o código é gerado por máquinas, o estudante deve ser avaliado por sua capacidade de especificar requisitos, auditar segurança e compreender as implicações éticas e arquiteturais das soluções sugeridas pela IA (SBC, 2023, p. 5).

Essa transição da "sintaxe" para a "arquitetura" (referenciada indiretamente por Sommerville, 2019) é urgente para a região do Vale do Paraíba, cujo parque tecnológico demanda alta confiabilidade em softwares críticos. O fato de 57% dos alunos participantes da pesquisa nunca ou raramente verificarem vulnerabilidades de segurança em códigos gerados pela IA é um indicativo de um hiato pedagógico perigoso.

Este estudo demonstrou que a IA não é mais uma promessa futura da programação, mas o ambiente real em que a programação acontece. Retomando o pensamento de Schwab (2016), o domínio sobre a Quarta Revolução Industrial não advém da negação tecnológica, mas da "gestão informada e humana do progresso técnico".

Portanto, conclui-se que o papel das faculdades do Vale do Paraíba deve migrar da instrução de codificação para a mentoria de inteligência aplicada. Conforme enfatizado pela UNESCO (2023) em citação direta: "A educação não deve ser movida pela tecnologia, mas as tecnologias de IA devem servir à missão da educação" (p. 2). Em suma, o programador do futuro, nascido neste contexto de 384 entrevistados, só será um engenheiro completo se for capaz de manter o papel de "mestre" de sua própria ferramenta, preservando a lógica e a criatividade humana — os verdadeiros diferenciais estabelecidos desde a gênese de Lovelace.

REFERÊNCIAS

CASTELLS, Manuel. **A sociedade em rede**. 20. ed. São Paulo: Paz e Terra, 2020.

CORMEN, Thomas H. et al. **Algoritmos: teoria e prática**. 4. ed. Rio de Janeiro: Elsevier, 2022.

DIEHL, Astor Antônio; TATIM, Maria Aparecida. **Metodologia da pesquisa científica: fundamentos e técnicas**. 2. ed. Curitiba: Intersaberes, 2004.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. 7. ed. São Paulo: Atlas, 2022.

ISAACSON, Walter. **Os inovadores: uma biografia da revolução digital**. São Paulo: Companhia das Letras, 2014.

KAPLAN, Andreas; HAENLEIN, Michael. Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence. **Business Horizons**, v. 62, n. 1, p. 15-25, 2019.

KASNECI, Enkelejda et al. ChatGPT for good? On opportunities and challenges of LLMs for education. **Learning and Individual Differences**, v. 103, p. 102274, 2023.

MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Fundamentos de metodologia científica**. 9. ed. São Paulo: Atlas, 2021.

MCCARTHY, John et al. A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence, August 31, 1955. **AI Magazine**, v. 27, n. 4, p. 12-14, 2006.

MONTGOMERY, Aaron. Deterministic AI: coding the future with LLMs. **Journal of Software Engineering Education**, v. 45, n. 2, 2023.

OECD. **Recommendation of the Council on Artificial Intelligence**. OECD/LEGAL/0449. Paris: Organisation for Economic Co-operation and Development, 2019.

PRITCHARD, Denny et al. Can Large Language Models Write Good Code? A Benchmarking Study. In: **Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education (ITiCSE)**. Turku, Finland: ACM, 2023.

RICHARDSON, Roberto Jarry. **Pesquisa social: métodos e técnicas**. 4. ed. São Paulo: Atlas, 2017.

RUSSELL, Stuart J.; NORVIG, Peter. **Inteligência artificial: uma abordagem moderna**. 3. ed. Rio de Janeiro: Elsevier, 2013.

SCHWAB, Klaus. **A quarta revolução industrial**. São Paulo: Edipro, 2016.

SEBESTA, Robert W. **Conceitos de linguagens de programação**. 11. ed. Porto Alegre: Bookman, 2018.

SOCIEDADE BRASILEIRA DE COMPUTAÇÃO (SBC). **Posicionamento da SBC sobre o uso de IA Generativa na formação em Computação**. Porto Alegre: SBC, 2023.

SOMMERVILLE, Ian. **Engenharia de software**. 10. ed. São Paulo: Pearson Education, 2019.

UNESCO. **Guidance for generative AI in education and research**. Paris: United Nations Educational, Scientific and Cultural Organization, 2023. Disponível em: <https://unesdoc.unesco.org/ark:/48223/pf0000386693>. Acesso em: 15 mai. 2024.

WIRTH, Niklaus. **Algorithms + Data Structures = Programs**. Upper Saddle River, NJ: Prentice Hall, 1985.

YIN, Robert K. **Estudo de Caso: planejamento e métodos**. 5. ed. Porto Alegre: Bookman, 2015.