

**INTELIGÊNCIA ARTIFICIAL PARA A OTIMIZAÇÃO DE CÓDIGO DO
SANDBOX ATÉ A PERFORMANCE COLETIVA: UMA ANÁLISE QUALITATIVA**

**ARTIFICIAL INTELLIGENCE FOR CODE OPTIMIZATION FROM SANDBOX
TO COLLECTIVE PERFORMANCE: A QUALITATIVE ANALYSIS**

**INTELIGENCIA ARTIFICIAL PARA LA OPTIMIZACIÓN DE CÓDIGO DESDE
EL SANDBOX HASTA EL DESEMPEÑO COLECTIVO: UN ANÁLISIS
CUALITATIVO**

 <https://doi.org/10.56238/arev7n9-036>

Data de submissão: 03/08/2025

Data de publicação: 03/09/2025

Lucas José Gasparin Corrêa Rufino

Especialização em Inteligência Artificial

Instituição: Faculdade Senai de Taubaté

E-mail: lucasjose.gasparin@gmail.com

Orcid: <https://orcid.org/0009-0002-4564-9631>

Marcello Pereira Benevides

Mestrando em Engenharia Mecânica

Instituição: Universidade de Taubaté

E-mail: marcello.benevides@sp.senai.br

Orcid: <https://orcid.org/0009-0004-2683-2275>

Karina Daniela Garcia Benevides

Mestranda em Engenharia Mecânica

Instituição: Universidade de Taubaté

E-mail: karinadaniela@uol.com.br

Orcid: <https://orcid.org/0009-0007-7322-1209>

David Felipe Alves dos Santos

Graduando em Gestão Produção Industrial

Instituição: Fatec São José dos Campos

E-mail: davidfelipe2707@gmail.com

Orcid: <https://orcid.org/0009-0002-8198-666X>

Alex Pisciotta

Doutorando em Engenharia

Instituição: Faculdade Senai de Taubaté

E-mail: alex.pisciottta@sp.senai.br

Orcid: <https://orcid.org/0000-0003-3898-4263>

Marco Rogério Silva Richetto

Doutorando em Engenharia Produção

Instituição: Escola Senai de Taubaté

E-mail: marco.richetto@unesp.br

Orcid: <https://orcid.org/0000-0003-3755-4341>

Márcia Regina de Oliveira
Doutora em Desenvolvimento Urbano
Instituição: Universidade de Taubaté
E-mail: oliveira.marcia@unitau.br
Orcid: <https://orcid.org/0000-0002-9462-9445>

Roque Antônio de Moura
Doutor em Engenharia
Instituição: Fatec São José dos Campos
E-mail: roque.moura@fatec.sp.gov.br
Orcid: <https://orcid.org/0000-0002-3036-7116>

RESUMO

Sandbox significa um ambiente seguro em que linhas de programação ou códigos são executados, testados e até validados em espaço seguro e isolado. O espaço do desenvolvimento atua como local funcional e onde podem ser os códigos, linhas de programa podem ser ajustados sem interferir no produto. A sandbox regulatória possibilita inovações e operações controladas. Por exemplo, a crescente integração da inteligência artificial (IA) no desenvolvimento de software impulsionou a otimização de processos e a refatoração de códigos. Neste sentido, esta pesquisa investiga a eficácia de diferentes IAs na otimização de códigos quanto ao desempenho final, comparando o desempenho de códigos funcionais desenvolvidos por humanos com versões otimizadas por quatro modelos de IA que abrangem as áreas de front-end e back-end em diferentes níveis de complexidade ciclomática. A metodologia envolveu a submissão de códigos a um prompt padronizado de refatoração, com a eficiência avaliada por um quociente que considera acurácia, tempo de execução e tamanho do código ($Q = A.t/S$). Os resultados indicaram um aumento substancial na eficiência dos códigos otimizados pelas IAs em comparação com os originais. Notou-se, contudo, variações de desempenho entre as IAs e as frentes de desenvolvimento. Conclui-se que a IA é um recurso para a otimização de código, mas sua eficácia é influenciada pelo contexto da tarefa e pelo modelo específico utilizado, destacando-se a necessidade de uma aplicação consciente e contextual da tecnologia.

Palavras-chave: Inteligência Artificial. Front-end. Back-end. Otimização Computacional. Sandbox.

ABSTRACT

A sandbox is a secure environment where programming lines or codes are executed, tested, and even validated in a safe and isolated space. The development space acts as a functional space, where code can be modified and adjusted without affecting the product. The regulatory sandbox enables innovation and controlled operations. For example, the increasing integration of artificial intelligence (AI) in software development has driven process optimization and code refactoring. In this sense, this research investigates the effectiveness of different AIs in optimizing code performance, comparing the performance of functional codes developed by humans with versions optimized by four AI models covering the front-end and back-end areas at different levels of cyclomatic complexity. The methodology involved submitting code to a standardized refactoring prompt, with efficiency assessed by a quotient that considers accuracy, execution time, and code size ($Q = A.t/S$). The results indicated a substantial increase in the efficiency of AI-optimized codes compared to the original ones. However, performance variations were observed between AIs and development fronts. The conclusion is that AI is a resource for code optimization, but its effectiveness is influenced by the task context and the specific model used, highlighting the need for a conscious and contextual application of the technology.

Keywords: Artificial Intelligence. Front-end. Back-end. Computational Optimization. Sandbox.

RESUMEN

Un sandbox es un entorno seguro donde se ejecutan, prueban e incluso validan líneas de programación o códigos en un espacio seguro y aislado. El espacio de desarrollo actúa como un espacio funcional, donde el código puede modificarse y ajustarse sin afectar al producto. El sandbox regulatorio facilita la innovación y el control de las operaciones. Por ejemplo, la creciente integración de la inteligencia artificial (IA) en el desarrollo de software ha impulsado la optimización de procesos y la refactorización de código. En este sentido, esta investigación analiza la eficacia de diferentes IA para optimizar el rendimiento del código, comparando el rendimiento de códigos funcionales desarrollados por humanos con versiones optimizadas por cuatro modelos de IA que abarcan las áreas front-end y back-end con diferentes niveles de complejidad ciclomática. La metodología implicó someter el código a una solicitud de refactorización estandarizada, cuya eficiencia se evaluó mediante un cociente que considera la precisión, el tiempo de ejecución y el tamaño del código ($Q = A \cdot t / S$). Los resultados indicaron un aumento sustancial en la eficiencia de los códigos optimizados por IA en comparación con los originales. Sin embargo, se observaron variaciones de rendimiento entre las IA y los frentes de desarrollo. La conclusión es que la IA es un recurso para la optimización de código, pero su eficacia se ve influenciada por el contexto de la tarea y el modelo específico utilizado, lo que resalta la necesidad de una aplicación consciente y contextualizada de la tecnología.

Palabras clave: Inteligencia Artificial. Front-end. Back-end. Optimización Computacional. Sandbox.

1 INTRODUÇÃO

A inteligência artificial (IA) transformou o desenvolvimento de aplicativos, programas e *softwares* desde o *sandbox* até o uso coletivo e personalizado da sociedade para reduzir esforços e com a percepção de não exceder limites cognitivos e evitar fadigas, principalmente em atividades repetitivas e monótonas que levam a fadiga (Silva *et al.*, 2024; Antônio *et al.*, 2024).

A IA revolucionou o desenvolvimento de *software* e sua aplicação na otimização de código emerge como um tema central na busca por eficiência e performance, ou seja, desde o ambiente de testes (*sandbox*) até a sua aplicação em grande escala. Contudo há barreiras devido falta de familiarização com o mundo digital e suas tecnologias e acessibilidade (Fernández-Herrero *et al.*, 2024; Benevides *et al.*, 2024).

A otimização de código, antes um processo manual e complexo, é agora potencializada pelos recursos da IA com sua biblioteca, ferramentas e algoritmos de aprendizado de máquina que analisam padrões e gargalos que seriam difíceis de serem percebidas por um desenvolvedora ou desenvolvedor humano (Antônio *et al.*, 2024) além de acelerar o ciclo de desenvolvimento, reduz o tempo gasto na depuração e melhora a qualidade do *software* (Becker *et al.*, 2023).

No *sandbox*, um ambiente controlado e isolado, a IA pode ser treinada para otimizar trechos de código específicos e é neste estágio, o foco é na experimentação e na validação de modelos simuladores (Egara; Mosimege, 2024). A análise qualitativa das melhorias de performance e da confiabilidade do código torna-se essencial antes da concepção, durante o desenvolvimento e principalmente na operacionalização dos comandos em um ambiente de produção (Da Silva Filho *et al.*, 2025; Moura *et al.*, 2024; Oliveira *et al.*, 2025).

A transição para a performance coletiva marca a próxima fase da otimização. Neste cenário, a IA atua em sistemas complexos, analisando o comportamento de múltiplos componentes para otimizar o desempenho de um sistema como um todo. A IA não apenas ajusta o código-fonte, mas também otimiza a alocação de recursos, o escalonamento e a arquitetura de software, garantindo que o sistema funcione de maneira eficiente mesmo em condições de alta carga (Cong, 2024).

Segundo (Borghoff; Minas; Mönch, 2025) no mundo do desenvolvimento algoritmizado, a crescente presença da inteligência artificial como ferramenta nos meios de trabalho é considerada uma facilitação para heurísticas esporádicas e, se usada com fundamentos, possibilita agilidade nos processos de criação e desenvolvimento, além de minimizar a obsolescência de processos de categoria *chore* que tratam semanticamente da classificação de códigos para tarefas corriqueiras, o que facilita em diferentes níveis e faces do desenvolvimento com formas diferentes de apresentação da solução utilizando por exemplo, o desenvolvimento *back-end* de um sistema complexo.

O proposto é apresentar para diferentes inteligências, códigos categorizados em complexidade e com contextos das faces de desenvolvimento (*front-end* e *back-end*) comparando e analisando a diferença de performance do código funcional feito por humano com o código e performance processual da IA conforme ilustra a Figura 1.

Figura 1. Comparativo entre humano e IA no desenvolvimento de um software.



Fonte: Autores (2025).

O uso da IA de forma contextual para redução de tempo de processamento, não é apenas uma forma de analisar o código com um viés técnico, mas também algo inevitável para o *future-proofing* do processo e do produto (Seo *et al.*, 2021).

2 REVISÃO DA LITERATURA

A IA tem facilitado o aprendizado tornando a forma de educar mais prazerosa ao possibilitar a personalização no processo ensino-aprendizado, principalmente para os chamados cinquentões que estão voltando de forma crescente aos bancos escolares em virtude de contínuas ofertar de empregos e a necessidade para trabalharem com a tecnologia digital (Benevides *et al.*, 2025; Costa *et al.*, 2025).

A análise qualitativa é um ponto-chave no desenvolvimento de um software, pois, embora métricas de desempenho sejam importantes, a análise qualitativa avalia como a IA transforma o processo de desenvolvimento, a colaboração da equipe e a inovação, zelando pelo envolvimento e a compreensão das decisões tomadas pela IA (Hermann, 2021), a interpretabilidade de seus modelos e o impacto em longo prazo na manutenção do código. Mais importante que o resultado, é saber como a metodologia qualitativa torna o processo robusto e possibilita a repetibilidade (Miguel *et al.*, 2018).

2.1 BACK-END E FRONT-END

Front-end é tudo o que vimos, ou seja, a fachada do *software* ou a parte que o cliente percebe. No desenvolvimento de *software* é a parte do sistema com a qual o usuário interage diretamente, incluindo a *interface* gráfica, os botões para clicar, os menus de navegação e todo o *design* visual.

O *back-end* se comunica com o *front-end* para enviar e receber informações, usando linguagens e tecnologias como *Python*, *Java*, *PHP*, *Ruby*, *Node.js* e outras conforme ilustra a Figura 2.

Figura 2. Front-end e back-end de um software.



Fonte: Autores (2025).

O *front-end* é construído com linguagens como HTML na estrutura, CSS para o estilo e *design* e a interatividade com o *JavaScript*. O *back-end* é o que cliente não vê. É o sistema que fica por trás da *interface* responsável por processar dados, armazenar informações e assegurar que todo o sistema funcione corretamente (Adel *et al.*, 2024; Cinar; Bilodeau, 2024; Seo *et al.*, 2021).

3 METODOLOGIA

A metodologia adotada nesta pesquisa tem natureza exploratória e aplicada, com foco no desenvolvimento de *software* comparando o desenvolvimento com e sem o uso de IA. A metodologia combina uma revisão bibliográfica sistemática utilizando as publicações e estudos de casos citados na literatura (Miguel *et al.*, 2018).

Uma base de conhecimento substancial foi utilizada para esta pesquisa usando *AI Code Review* para se introduzir uma visão analítica para a utilização de IA como ferramenta de *pair-programming* (Almeida *et al.*, 2024; Fan *et al.*, 2025).

Estudos sobre programação aplicada com inteligência artificial para educação, apresentaram uma boa fundamentação para a aplicação de análise de performance (Lewowski; Madeyski, 2022).

Coelho *et al.* (2017) citam ser relevante criar um espaço isolado e seguro para promover o desenvolvimento e implementar sem risco a tecnologia simulacional. Serão desenvolvidos códigos categorizados por complexidade ciclomática, seguindo o índice de complexidade de McCabe e separados por frente de desenvolvimento, assim então serão submetido à métrica de eficiência do estudo conforme Equação 1:

$$Q = \frac{A}{t.s} \quad (1)$$

Onde:

Q = quociente de eficiência

A = acurácia (%)

t = tempo de execução (μs)

S = size ou tamanho (bytes)

O funcionamento do código requer que ele seja funcional para a gradação correta, portanto, os códigos otimizados que não fornecem o mesmo resultado e tipagem que os originais, são reduzidos na forma de acurácia, isto é, se duas das quatro inteligências apresentarem códigos que não funcionam, ou seja 0% de cumprimento, a acurácia média das IAs em adaptar o código é de 50%.

Para códigos que não apresentarem resultados melhores que os originais, mas realizam o mesmo funcionamento, a acurácia é afetada em até 50%, portanto, se uma inteligência apresentou um código ineficiente, a média se torna 87,5%.

O quociente de eficiência será responsável por prever e verificar uma perspectiva da performance de todos os códigos criados, valorizando-se a otimização com destaque para a funcionalidade e ao volume ou tamanho de alocação conforme ilustra a Figura 3.

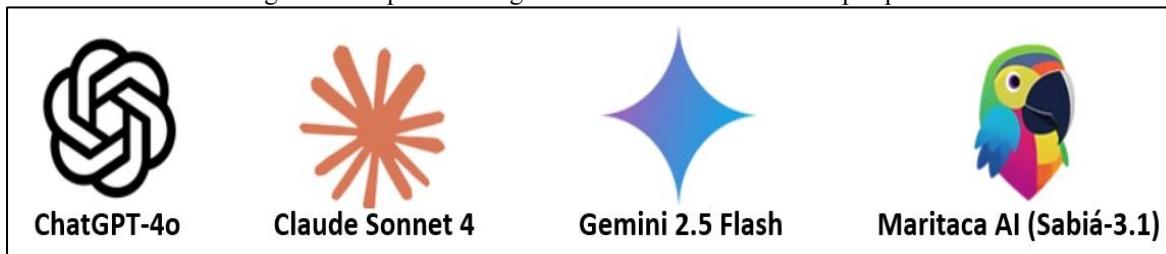
Figura 3. Separação de código por frente e complexidade (McCabe)

Região	Baixa complexidade	Média complexidade	Alta complexidade
Front-End	Código F#1	Código F#2	Código F#3
Back-End	Código B#1	Código B#2	Código B#3

Fonte: Autores (2025).

Foi solicitado por meio de um *prompt* padronizado acionando quatro inteligências artificiais conforme ilustra a Figura 4.

Figura 4. As quatro inteligências artificiais usadas nesta pesquisa.



Fonte: Autores (2025).

As IAs revisaram e realizaram uma refatoração com otimização dos códigos preliminarmente fornecidos, sem um viés pré-existente visado otimizar o tempo de execução e reduzir o tamanho, mantendo a funcionalidade original com solicitação para que se reescreva o código.

Os dados obtidos dos testes com as diferentes inteligências artificiais serão expostos em dado único, comparando a performance de otimização entre inteligências, e também serão utilizadas as médias dos dados na aplicação da fórmula de quociente de eficiência, para comparação com os códigos não otimizados.

Para transferir os resultados da eficiência para valores comparativos em porcentagem, no gráficos de radar, foi utilizada a Equação 2.

$$Porcentagem = \frac{Valor\ Atual}{Valor\ Máximo} \times 100 \quad (2)$$

Os códigos criados para submissão as inteligências foram criados utilizando a linguagem Javascript, e a eficiência média das regiões não otimizadas sendo:

Códigos *Front-End* = $Q^* = 0,93$

Códigos *Back-End* = $Q^* = 1,53$

*Sendo Q o quociente de eficiência

Os valores são os mais próximos do experimento e resultados ao valor de 1, eficiência normal, pois para captação de valores precisos, foi utilizada uma técnica de iterações de aquecimento e iterações de medição, sendo respectivamente o aquecimento: 1000 (mil) iterações e a medição de 1000000000 (um bilhão) de iterações. Utilizando essa estratégia, captando uma média do tempo extremamente sensível ao tempo de processamento, é possível captar as flutuações de tempo na métrica de nanossegundos, atingidos índices de complexidade de McCabe [código (1 → 1; 2 → 17; 3 → 36)].

4 RESULTADOS E DISCUSSÕES

4.1 PECULIARIDADES DE CADA INTELIGÊNCIA USADA NESTA PESQUISA

Como resultado foram obtidos dados comparativos de eficiência, explicitados graficamente assim como a performance individual de cada inteligência, com a otimização. Foi realizado uma análise das principais funcionalidade para uso e particularidades de cada uma das quatro IAs usadas nesta pesquisa conforme Figura 5.

Figura 5. As quatro inteligências artificiais usadas nesta pesquisa e suas características funcionais.

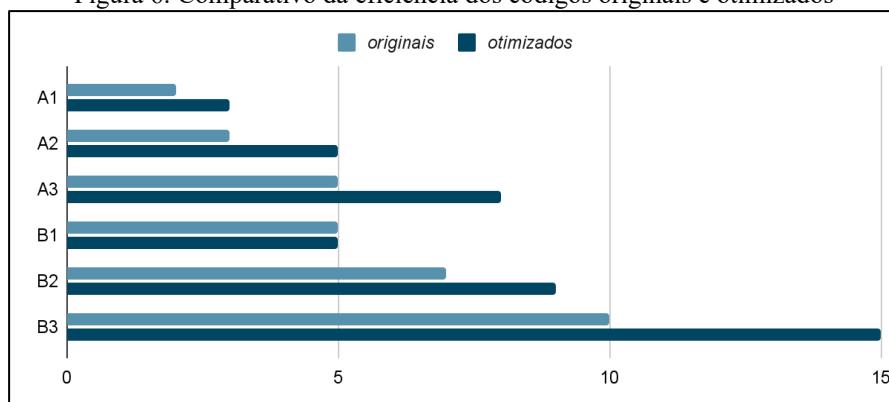
IA	Particularidades e funcionalidade
ChatGPT-4o	A principal característica é a capacidade multimodal nativa. Pode-se processar e gerar conteúdo em uma única rede neural para texto, áudio e imagens. Consegue por exemplo, analisar uma imagem e responder com áudio ou traduzir uma conversa em tempo real com pouca latência.
Gemini 2.5 Flash	Se destaca pela velocidade e eficiência de custo. Projetado para ser um modelo ágil e de trabalho para tarefas de alto volume e sensíveis à latência. Sua peculiaridade está em ser um modelo pensante que raciocina por seus pensamentos antes de responder, o que melhora a precisão. Ideal para aplicações em tempo real, como chatbots e resumos rápidos.
Maritaca AI (Sabiá-3.1)	A peculiaridade está no português brasileiro . Diferente de modelos globais que são majoritariamente treinados em inglês e depois adaptados, o Sabiá-3.1 foi criado e treinado em português do Brasil, o que possibilita que ele compreenda nuances, gírias, expressões regionais e o contexto cultural brasileiro. É uma IA especializada para o mercado brasileiro.
Claude Sonnet 4	Possui equilíbrio entre desempenho e praticidade. Otimizado como modelo de tamanho médio que oferece alta qualidade em tarefas de codificação com um custo mais baixo do que seus pares. Realiza raciocínio híbrido, fornecendo respostas instantâneas ou, se necessário entrar em modo pensamento estendido para resolver problemas mais complexos passo a passo. Versátil para agentes de IA e fluxos de trabalho que exigem precisão.

Fonte: Autores (2025).

4.2 RESULTADOS COMPARATIVOS DA EFICIÊNCIA DOS CÓDIGOS

Percebe-se que houve um aumento substancial na eficiência dos códigos otimizados em comparação aos originais, todos obtiveram um resultado funcional, o que graficamente podem ser verificados. Na Figura 6 um comparativo da eficiência de cada IA.

Figura 6. Comparativo da eficiência dos códigos originais e otimizados

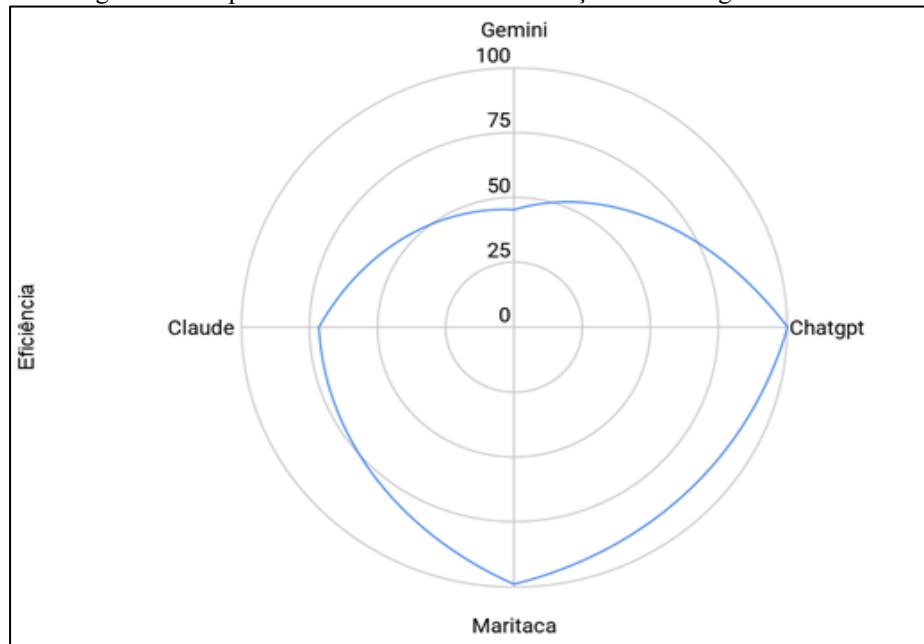


Fonte: Autores (2025).

4.3 RESULTADOS COMPARATIVOS DA OTIMIZAÇÃO COM CÓDIGOS *FRONT-END*

Na Figura 7 ilustra-se que as inteligências Claude e ChatGPT tiveram uma performance melhor em comparação às outras, entretanto comparativamente, considerando a porcentagem 0% para uma transferência do valor 0. Nesse quesito o Gemini teve a performance menor que 50%.

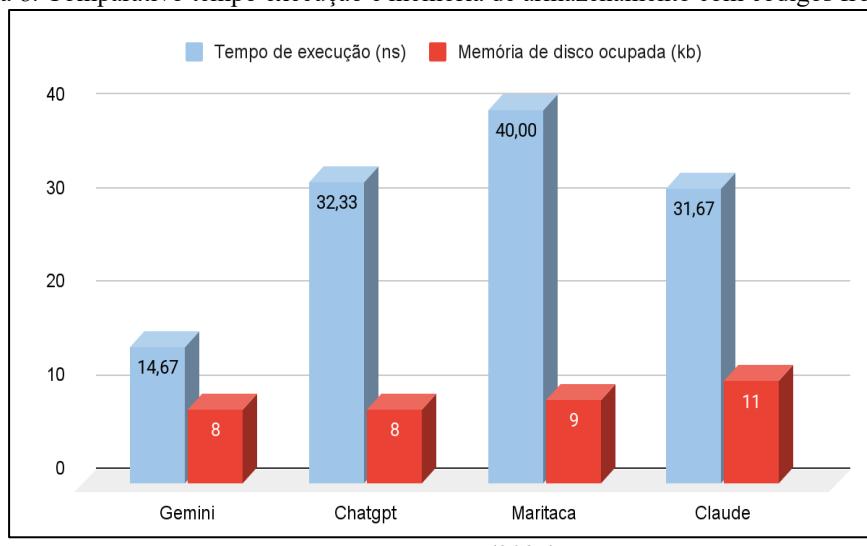
Figura 7. Comparativo da eficiência da otimização com códigos front-end.



Fonte: Autores (2025).

Na Figura 8 ilustra-se Comparativo tempo execução e memória de armazenamento com códigos *front-end*.

Figura 8. Comparativo tempo execução e memória de armazenamento com códigos front-end.

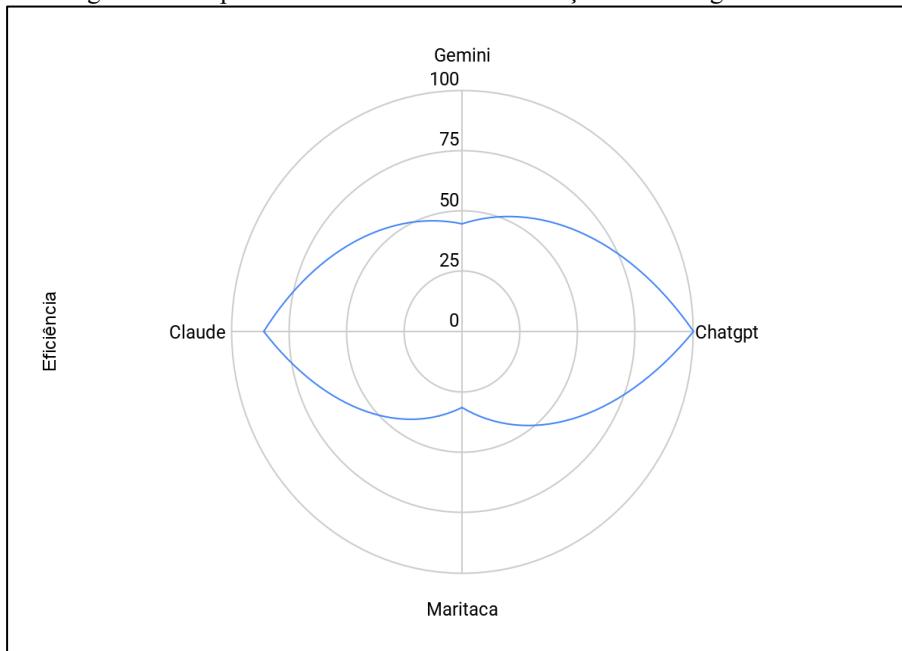


Fonte: Autores (2025).

4.4 RESULTADOS COMPARATIVOS DA OTIMIZAÇÃO COM CÓDIGOS BACK-END

A Figura 9 ilustra que ao se comparar, a IA ChatGPT tem a acurácia mais baixa e melhor resultado para velocidade. Apenas 60% dos códigos foram funcionais da mesma forma que eram originalmente.

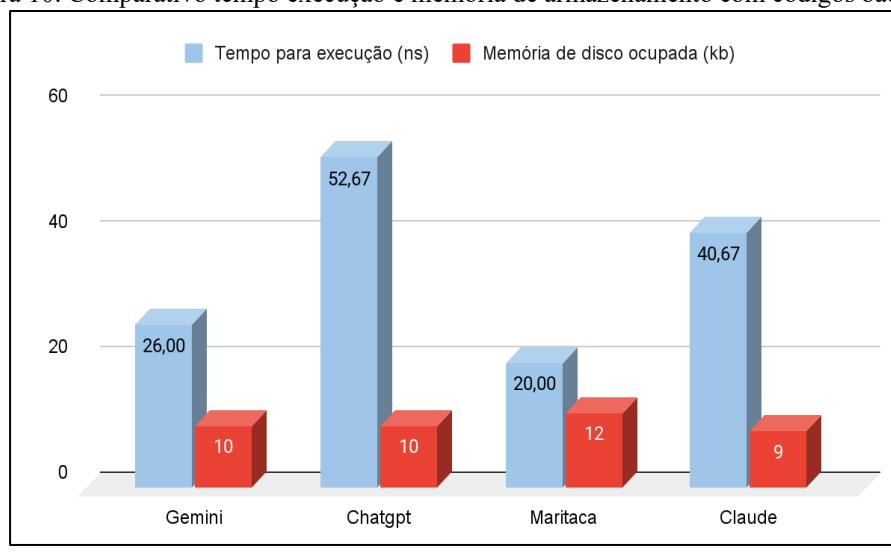
Figura 9. Comparativo da eficiência da otimização com códigos back-end.



Fonte: Autores (2025).

Na Figura 10 ilustra-se Comparativo tempo execução e memória de armazenamento com códigos *back-end*.

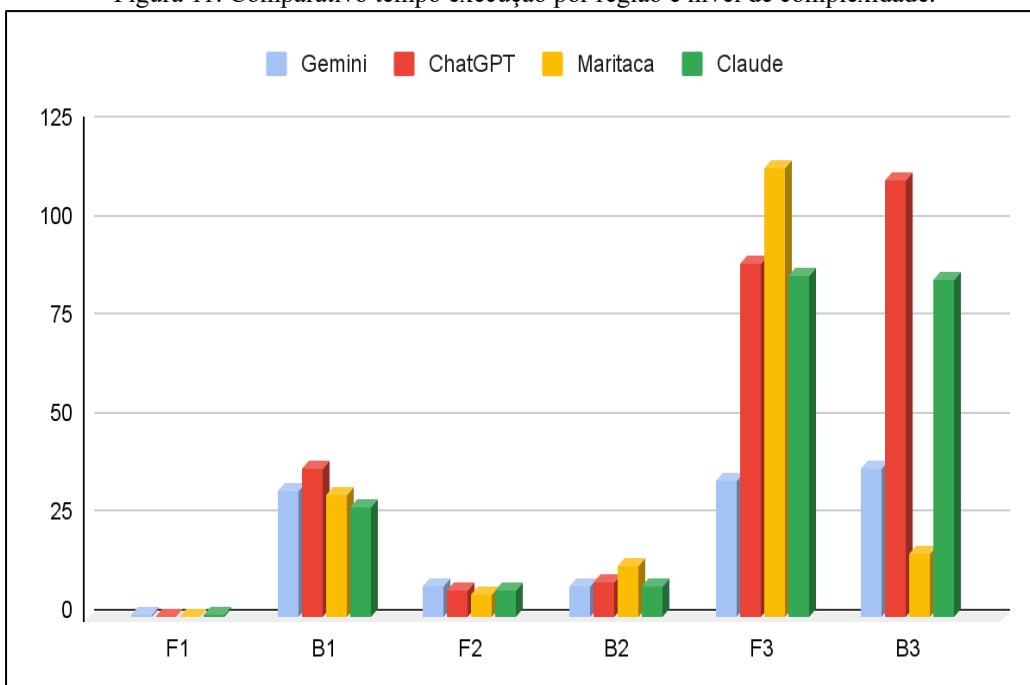
Figura 10. Comparativo tempo execução e memória de armazenamento com códigos back-end.



Fonte: Autores (2025).

Na Figura 11 ilustra-se um comparativo de tempo para execução categorizados por inteligências, agrupados por região de desenvolvimento e nível de complexidade.

Figura 11. Comparativo tempo execução por região e nível de complexidade.



Fonte: Autores (2025).

5 CONSIDERAÇÕES FINAIS

As inteligências artificiais são ferramentas importantes para o uso no dia a dia do desenvolvimento de *software* que se utilizadas como auxiliares, podem otimizar processos e refatorar códigos obsoletos ou de difícil compreensão humana.

É válido ressaltar as diferentes eficiências nas frentes de desenvolvimento e diferentes inteligências, essas nuances abrem espaço para um estudo experimental mais aprofundado.

Como pesquisas futuras recomenda-se pesquisar outras metodologias na criação e desenvolvimento de *software* com IA usando o *framework* Scrum.

REFERÊNCIAS

- ADEL, A.; AHSAN, A.; DAVISON, C. (2024). **ChatGPT Promises and Challenges in Education: Computational & Ethical Perspective.** Education Science., 14, 814. DOI 10.3390/educsci14080814.
- ALMEIDA, Y.; ALBUQUERQUE, D.; DANTAS FILHO, E.; MUNIZ, F.; SANTOS, K. F.; PERKUSICH, M.; ALMEIDA, H.; PERKUSICH, A. **AICodeReview: Advancing code quality with AI-enhanced reviews.** Software X, v. 26, 2024. DOI: <https://doi.org/10.1016/j.softx.2024.101677>
- ANTÔNIO, M. R., REGINA, O. M., GOUSSAIN, B. G. C. S., SILVA, M. B. (2024). **Neuroergonomics approach in the workplace aiming to standardize movements and increase workers' sense of well-being.** 24(10), 472–482. <https://doi.org/10.53660/CLM-3313-24H27>
- BECKER, B. A., DENNY, P., FINNIE-ANSLEY, J., LUXTON-REILLY, A., PRATHER, J., & SANTOS, E. A. (2023). **Programming is hard or at least it used to be: Educational Opportunities and Challenges of AI Code Generation.** SIGCSE 2023. <https://doi.org/10.1145/3545945.3569759>.
- BENEVIDES, K. D. G.; RUFINO, L. J. G. C.; DOS SANTOS, D. F. A.; BENEVIDES, M. P.; PIMENTA, C. D.; DE OLIVEIRA, M. R.; DE MOURA, R. A. (2025). **Inteligência artificial na educação de indivíduos adultos com 50 anos de idade ou mais:** uma abordagem assistiva. **ARACÊ**, [S. l.], v7, n8, p.e7160, 2025. DOI: 10.56238/arev7n8-085. Disponível em: <https://periodicos.newsciencepubl.com/arace/article/view/7160>. Acesso em: 27 ago. 2025.
- BENEVIDES, M. P.; XAVIER, K. R. S. L.; et al. (2024) **Sign talk assistive technology: real-time recognition of the libras typical alphabet using artificial intelligence.** RGSA, v. 18, n. 12, p. e010610, 2024. DOI: <https://doi.org/10.24857/rgsa.v18n12-214>
- BORGHOFF, U. M.; MINAS, M.; MÖNCH, K. (2025). **Generative AI in student software development projects:** A user study on experiences and self-assessment. In: Proceedings of the 6th European Conf. Software Eng. Education, 2025. DOI: <https://doi.org/10.1145/3723010.3723012>.
- CINAR, A.B., & BILODEAU, S. (2024). **Incorporating AI into the Inner Circle of Emotional Intelligence for Sustainability.** Sustainability, 16, 6648. <https://doi.org/10.3390/su16156648>.
- COELHO, M. H.; SARTOR, M.; MANENTE, M. T.; FRIGO, L. B. e POZZEBON, E. (2017). **Tecnologia, inovação e educação: caminhando juntas para o desenvolvimento de smart cities.** In: Revista novas tecnologias na Educação – UFRGS. v.15, n. 2.
- CONG, Y. (2024). **AI Language Models: An Opportunity to Enhance Language Learning.** Informatics, 11, 49. <https://doi.org/10.3390/informatics11030049>.
- COSTA, J. C. L.; SANTOS, D. F. A.; OLIVEIRA, M. R. de; MOURA, R. A. (2025). **Aprendizagem com solução de problemas reais para aprimoramento discente na injunção socioprofissional.** Revista CLCS, [S l], v18, n 2, p. e15288, 2025. DOI: 10.55905/revconv.18n.2-100.

DA SILVA FILHO, A. L.; BENEVIDES, M. P.; NOHARA, E. L.; DE MOURA, R. A. (2025). **Engenharia mecânica na construção de máquina-ferramenta portátil para usinar peças de até 1200 milímetros de diâmetro.** ARACÊ, [S. l.], v7, n7, p.40298–40314. DOI: 10.56238/arev7n7-295. Disponível em: <https://periodicos.newsciencepubl.com/arace/article/view/6788>. Acesso: 28ago2025.

EGARA, F. O., & MOSIMEGE, M. (2024). **Exploring the integration of artificial intelligence-based ChatGPT into mathematics instruction:** Perceptions, Challenges, and Implications for Educators. Education Sciences, 14, 742. <https://doi.org/10.3390/educsci14070742>.

FAN, G. et al. **The impact of AI-assisted pair programming on student motivation, programming anxiety, collaborative learning, and programming performance:** a comparative study with traditional pair programming and individual approaches. International Journal of STEM Education, v. 12, n. 1, p. 16, 2025.

FERNÁNDEZ-HERRERO, J. (2024). **Evaluating Recent Advances in Affective Intelligent Tutoring Systems:** A Scoping Review of Educational Impacts and Future Prospects. Educ. Sci., 14, 839. <https://doi.org/10.3390/educsci14080839>.

HERMANN, E. **Artificial intelligence and mass personalization of communication content—an ethical and literacy perspective.** New Media & Society, v. 24, p. 1258-1277, 2021. DOI: <https://doi.org/10.1177/14614448211022702> .

LEWOWSKI, T.; MADEYSKI, L. (2022). **Code smells detection using artificial intelligence techniques: A business-driven systematic review.** Developments in Information & Knowledge Management for Business Applications, p. 285-319, 2022.

MIGUEL, P. A. C.; FLEURY, A.; MELLO, C. H. P.; NAKANO, D. N.; LIMA, E. P.; TURRIONI, J. B.; HO, L. L.; MORABITO, R.; MARTINS, R. A.; SOUSA, R.; COSTA, S. E. G.; PUREZA, V. 2018. **Metodologia de pesquisa em engenharia de produção e gestão de operações.** 3^a Ed. Rio de Janeiro: Editora GEN LTC. 2018. ISBN 978-853529134-6. ISBN 13 – 978-8535291346.

MOURA, RA DE, SANTOS, DFA, BENEVIDES, MP, RICCHETTO, MRS, OLIVEIRA, MR DE, & SILVA, MB (2024). **Neurociência e ergonomia aplicadas como ciências comportamentais profissionais para longevidade saudável.** Revista De Gestão - RGSA , 18 (12), e09741. <https://doi.org/10.24857/rgsa.v18n12-077>

OLIVEIRA, M. R.; BENEVIDES, K. D. G.; RUFINO, L. J. G. C.; SANTOS, D. F. A.; BENEVIDES, M. P.; MOURA, R. A. (2025). **Direito Digital e sua limitação no uso da inteligência artificial hodierna: um ponto para reflexão e ações requeridas.** CLCS, [S. l.], v. 18, n. 7, p. e19679, 2025. DOI: 10.55905/revconv.18n.7-341. <https://ojs.revistacontribuciones.com/ojs/index.php/clcs/article/view/19679>

SEO, K., TANG, J., ROLL, I., FELS, S., & YOON, D. (2021). **The impact of artificial intelligence on learner-instructor interaction in online learning.** International Journal of Educational Technology in Higher Education, 18, 54. <https://doi.org/10.1186/s41239-021-00292-9>.

SILVA, E. A.; CAMARGO, A. A.; SILVA, M. B.; MOURA, R. A. (2024). Neuroergonomía y Tecnologías inmersivas para lograr un envejecimiento saludable sin dolor y además sin ortesis. Revista Exatas. V.30. UNITAU. DOI: <https://doi.org/10.69609/1516-2893.2024.v30.n2.a3916>