Análise comparativa de IAs generativas como ferramentas de apoio à programação segura

Danilo Ataide, Eduardo Alves, Emerson Quinaia, Alexandre Braga

CPQD

Pergunta

Assistentes de programação com IA generativa são capazes de apoiar tarefas de programação segura?

Objetivos

Avaliar a efetividade dos ***assistentes de programação apoiados por IAs generativas*** na realização de tarefas de programação segura.

Será que os assistentes de programação conseguem:

- Reconhecer vulnerabilidades em códigos inseguros?
- Corrigir vulnerabilidades (simples) identificadas?
- Escrever códigos seguros?

Avaliação das respostas das IAs

A efetividade pode ser medida a partir da qualidade das respostas.

Avaliada com os critérios a seguir:

- **Produtividade**: velocidade e eficiência com que o programador auxiliado pela ferramenta identifica e corrige vulnerabilidades.
- **Consistência**: homogeneidade de respostas ao longo do tempo, considerando diferentes códigos e linguagens de programação.
- **Assertividade**: quantidade de falsos positivos ou falsos negativos, bem como avaliação de métricas na detecção de vulnerabilidades.
- **Confiabilidade**: códigos sem vulnerabilidades e testes não destrutivos (p. ex., teste de SQLi com “DROP TABLE”).

Modelos de prompt para análise código

Verifique se o código a seguir apresenta falhas de segurança e vulnerabilidades.

Desenvolva uma solução segura para a vulnerabilidade de segurança apresentada.

Implemente um código de teste para a vulnerabilidade do código.

Metodologia de trabalho

Duas grandes etapas:

- Coleta de dados com aplicação de prompts
- Análise manual das respostas
  - Testadores e validadores

Bases de conhecimento

- Common Weakness Enumeration (CWE)
  - 34 exemplares
- SEI CERT Oracle Coding Standard for Java
  - 34 exemplares
- SEI CERT C Coding Standard
  - 148 exemplares

IAs avaliadas e ambiente de teste

- IAs generativas
  - Microsoft Copilot (v1.7, LLM GPT 4-Turbo) *
  - Codium (v0.9, LLM GPT 4) *
  - Amazon Code Whisperer (Amazon Q v1.9.0) *
  - Claude Ai (Chatgpt 3.5) +
  - Gemini (1.5 Flash) +
  - Codeium (proprietary model, third party OpenAI APIs) +

- 4 IAs generativas utilizadas com extensão do IDE VS Code
  - Github Copilot, Codium , Amazon Q e Codeium

- 2 IAs Generativas utilizadas diretamente no navegador
  - Gemini e Claude AI

*** no artigo**
+ posteriores

Análise por métricas de classificação

$$\text{Precision} = \frac{VP + VN}{VP+FP+VN+FN}$$

$$\text{Recall} = \frac{VP}{VP+FN}$$

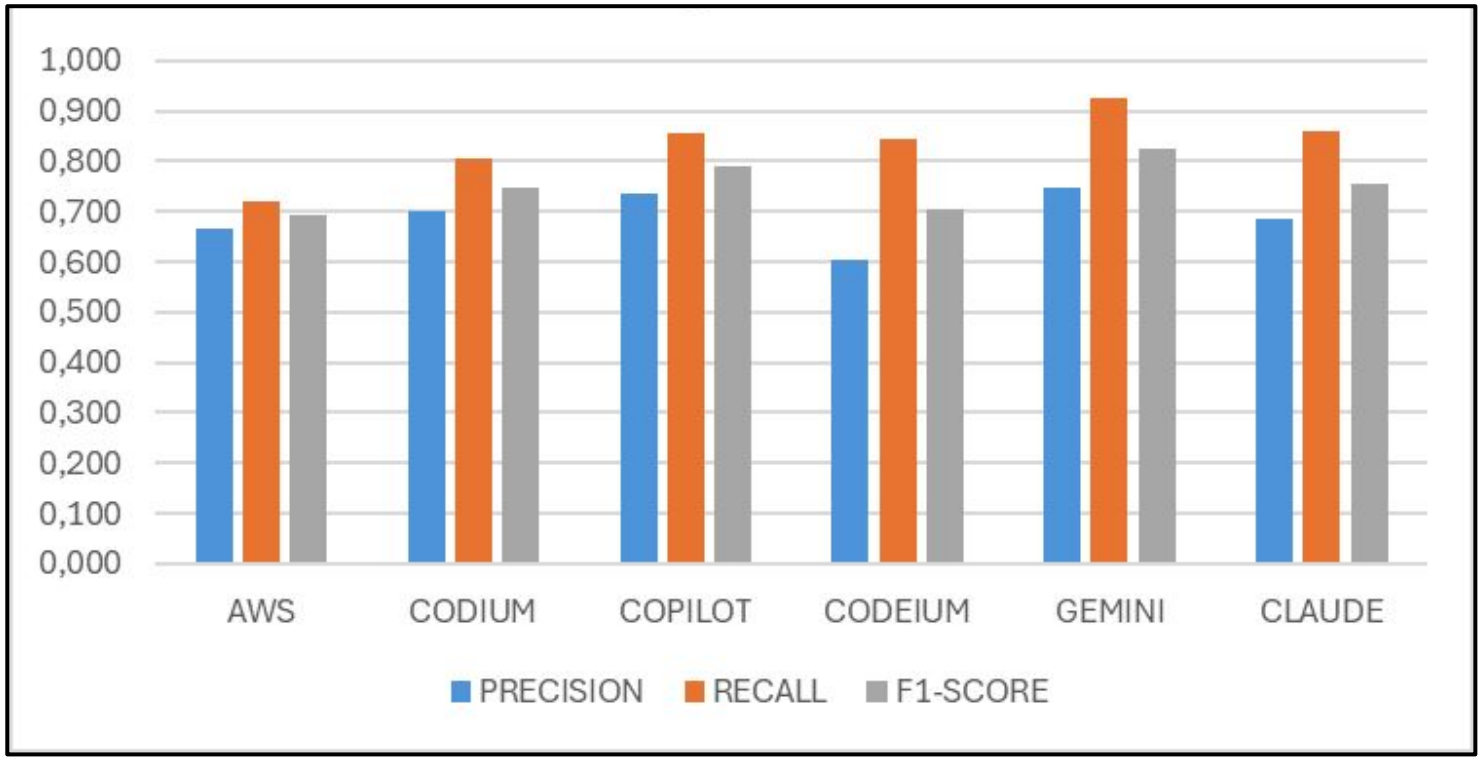
$$\text{F1-Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision}+\text{Recall}}$$

Legenda:

- Verdadeiro Positivo (VP)
- Falso Positivo (FP)
- Verdadeiro Negativo (VN)
- Falso Negativo (FN)

Recall mede a capacidade de identificar todos os exemplos positivos. ***Precision*** é proporção de classificados positivos que são realmente positivos. ***F1-score*** é média harmônica de Precision e Recall.

Avaliação quantitativa - resultados gerais



Observações e avaliação qualitativa

- As **métricas** Precision, Recall e F1-score estão **acima de 0,60**
    - Eficácia **comparável à** de ferramentas **SAST open source**
- IAs tendem a apontar todas as vulnerabilidades possíveis no código, não só o que se esperava (**alarmes falsos**)
- Em alguns casos, vários erros são apontados, mas não o que se esperava (**alarme falso + omissão**)
- Respostas prolixas, confusas e imprecisas **amplificam o efeito dos alarmes falsos** e fazem o desenvolvedor perder mais tempo
- Qualidade da resposta oscila ao longo do tempo (**não determinismo**).

Considerações finais

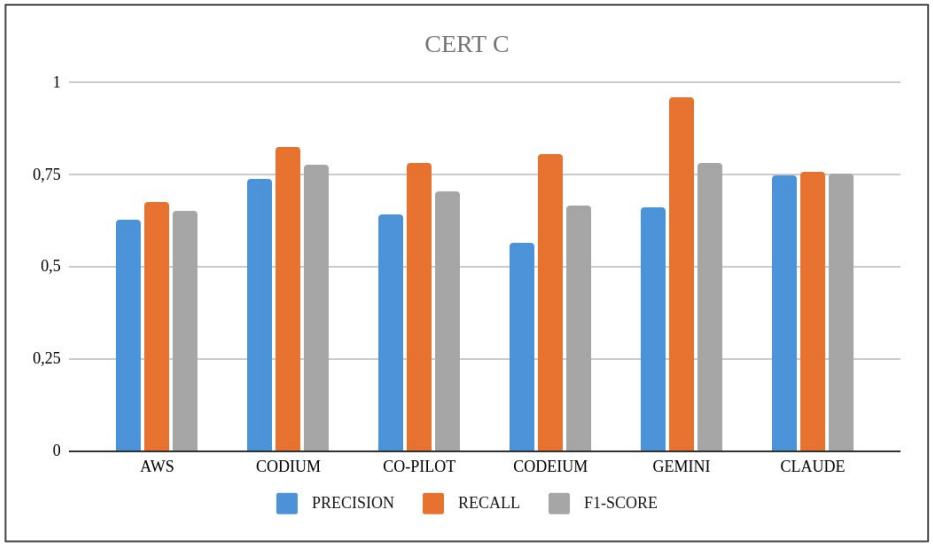
- **Assistentes de programação** com IA generativa podem estar **relacionados a código inseguro**
- Desenvolvedor tende a confiar na segurança do código sugerido pela IA (**otimista com a tecnologia**)
  - Análise da resposta da **IA pode decepcionar**
  - Falta uma engenharia de prompt seguro?
- Com base no **F1-score** apenas, **Gemini é a melhor escolha** entre os assistentes avaliados
- **Copilot** tem um **F1-score** quase tão **bom e usabilidade** melhor
  - Uso mais intuitivo pela **IDE** e com respostas mais objetivas
- **Ainda há muito por fazer em avaliação e comparação das IAs**

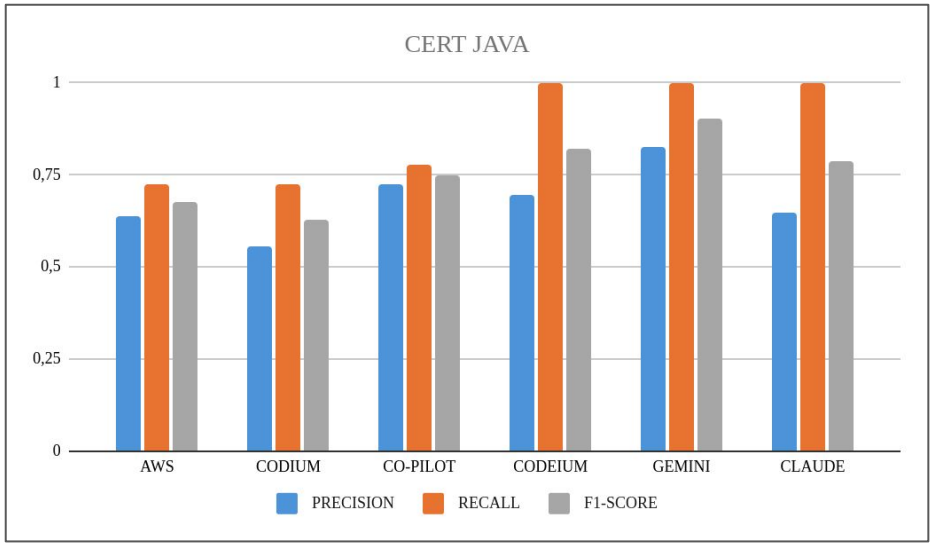
Obrigado!

- Alexandre Braga
- Contato(s)
  - ambraga@cpqd.com.br
  - linkedin.com/in/alexmbraga

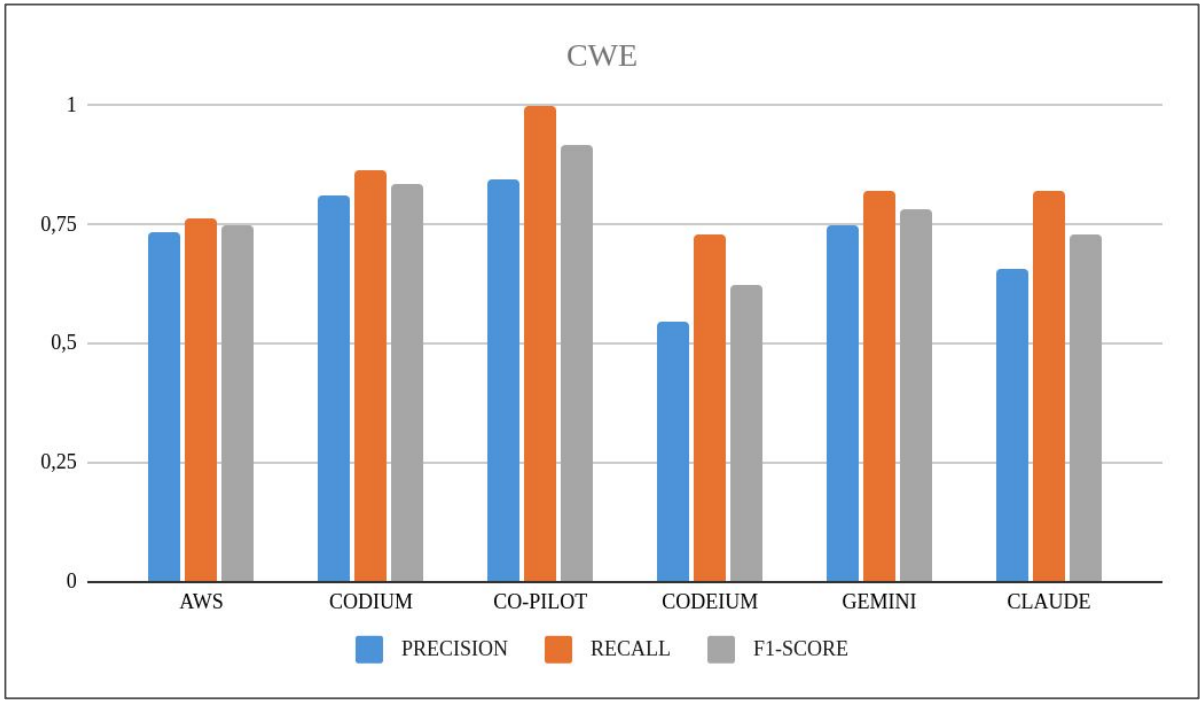
Este projeto foi apoiado pelo Ministério da Ciência, Tecnologia e Inovações, com recursos da Lei no 8.248, de 23 de outubro de 1991, no âmbito do PPI-SOFTEX, coordenado pela Softex e publicado PDI 03, DOU 01245.023862/2022-14.

Avaliação - resultados detalhados





Avaliação - resultados detalhados



Patrocinadores do SBSeg 2024!

nic.br

cgi.br

Google

Tempest

CAPES

SiDi

FAPESP

zscaler™

BugHunt

CNPq

C.E.S.A.R

FACULDADE IBPTECH