

Programação de Jogos com Google Colab e Chatgpt

Isabela Freitas¹, Pedro Coura¹, José Augusto Nacif¹, Ricardo Ferreira¹

¹Departamento de Informática, Centro de Ciências Exatas e Tecnológicas
Universidade Federal de Viçosa

{isabela.castro.freitas,pedro.coura,ricardo,jnacif}@ufv.br

Resumo. Para acessar o vídeo, clique [aqui](#) e para acessar os exemplos, clique [aqui](#). O rápido desenvolvimento de modelos generativos de linguagem tem aberto novas oportunidades para o ensino de programação, especialmente no contexto de desenvolvimento de jogos [Maleki and Zhao 2024, Akoury et al. 2023, Hu et al. 2024]. Jogos constituem instrumentos atraentes para o ensino de programação, permitindo estimular o raciocínio lógico e a aprendizagem de conceitos computacionais de forma interativa e envolvente. O objetivo deste trabalho é avaliar o desempenho de modelos generativos de linguagem, como o ChatGPT [OpenAI 2023], na geração de código didático e funcional para jogos, explorando um processo sistemático de criação, compreensão e modificação de códigos de jogos educacionais. A metodologia proposta compreende três etapas principais: (1) Exploração inicial de prompts para interação com modelos de linguagem e geração de jogos; (2) Solicitação de modificações no código gerado para adição de novas funcionalidades; (3) Análise e explicação do código para introdução manual de intervenções e modificações. Como ambiente de desenvolvimento, optou-se pelo Google Colab [Canesche et al. 2021, Sabino et al. 2024, Ferreira et al. 2023], que oferece portabilidade, independência de dispositivo e suporte a linguagens como Python e Javascript. Os resultados preliminares demonstram a capacidade dos modelos em gerar diversos tipos de jogos, desde clássicos como Space Invaders e Tetris até jogos de tabuleiro como Dama e Resta-um. Adicionalmente, explorou-se a portabilidade para plataformas com restrições de programação como Arduino, NodeMCU e Esp32, incluindo implementações com displays de baixo custo e painéis de LED. Os exemplos completos estão disponíveis de forma colaborativa no Google Colab. A seguir ilustraremos alguns exemplos.

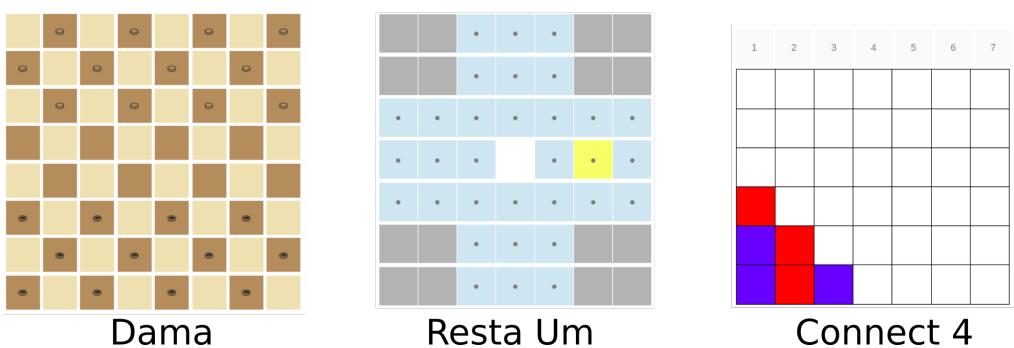


Figura 1. Jogos de Tabuleiro e Desafios: Dama, Resta-um e Connect 4.

A Figura 1 ilustra três jogos gerados com facilidade pela Chatgpt versão gratuita. Na elaboração do prompt foi solicitado o uso widgets do Python para iteratividade. Para jogos de ação, sugerimos que o prompt inicial solicite o uso

de código Javascript. Resultados preliminares evidenciam a geração com facilidade os jogos clássicos como Space Invaders, Tetris e breakout com Javascript como ilustrado na Figura 2.

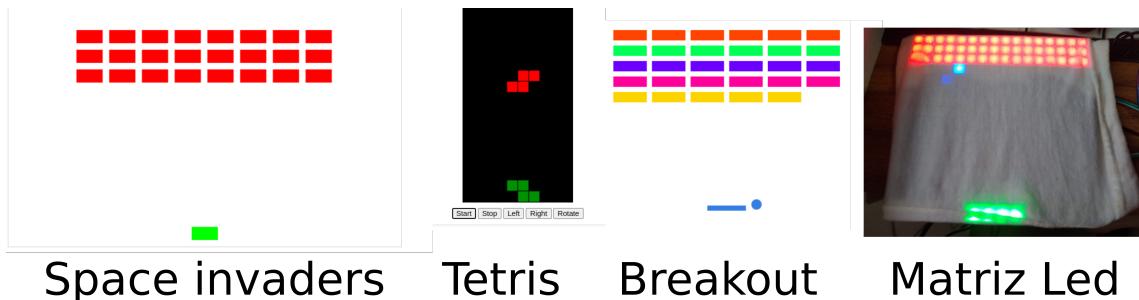


Figura 2. Jogos Clássicos em Javascript ou Arduino com painéis de Led.

O segundo passo da nossa metodologia é trabalhar com as versões ou variações dos jogos. Por exemplo fazer um jogo com duas snakes e observando as modificações nos códigos gerados para os jogos. Deve fazer parte da interação com as LLM (Large Languages Models) verificar o resultado. Jogos são ótimos casos de uso, onde a tarefa de verificação é bem lúdica e gera motivação nos alunos, para ajustar os erros com mais interações e novos prompts ou modificações diretamente no código. Na parte final da metodologia, os estudantes exploram o código gerado, solicitando explicações para as LLMs para manualmente introduzir intervenções nos jogos. Em nossos experimentos também avaliamos as possibilidades de migração do jogo para ambientes com mais restrições de programação que é o caso das plataformas "Arduino". Nossa sugestão é usar displays de baixo custo ou painéis de led, que são mais desafiadores devido a baixa resolução mas tem um impacto visual.



Figura 3. Portabilidade do Jogo para Arduino com controle no celular pelo Wifi

A figura 3 ilustra uma demonstração com uso Wifi com o controle do jogo no celular. O "Arduino" provê a "Internet", então pode ser usado até em ambientes sem Wifi com baixo custo criando um ambiente de laboratório de ensino móvel. Sugerimos solicitar aos "prompts" o uso de do protocolo MQTT [Sabino et al. 2024] ou Websockets para jogos de ação. Para acessar os exemplos, clique aqui. Apoio financeiro da Bolsa de Iniciação Científica da FAPEMIG e do CNPq programa Institucional, Bolsa PIBEN Funarbe/Pro-Reitoria de Ensino UFV, FAPEMIG APQ-01577-22, CNPq e Centro de Ciência Exatas e Tecnológicas da UFV.

Referências

- Akoury, N., Yang, Q., and Iyyer, M. (2023). A framework for exploring player perceptions of llm-generated dialogue in commercial video games. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2295–2311.
- Canesche, M., Bragança, L., Neto, O. P. V., Nacif, J. A., and Ferreira, R. (2021). Google colab cad4u: Hands-on cloud laboratories for digital design. In *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5. IEEE.
- Ferreira, R., Canesche, M., and Penha, J. (2023). Google colab para ensino de computação. In *Anais Estendidos do III Simpósio Brasileiro de Educação em Computação*, pages 46–47. SBC.
- Hu, C., Zhao, Y., and Liu, J. (2024). Generating games via llms: An investigation with video game description language. *arXiv preprint arXiv:2404.08706*.
- Maleki, M. F. and Zhao, R. (2024). Procedural content generation in games: A survey with insights on emerging llm integration. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 20, pages 167–178.
- OpenAI (2023). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Sabino, C. E., Nacif, J. A., and Ferreira, R. (2024). Acesso compartilhado à recursos físicos com google colab. In *Simpósio Brasileiro de Educação em Computação (EDU-COMP)*, pages 11–12. SBC.