

Gatinho Classificador: Uma Abordagem Gamificada para o Ensino Intuitivo de Algoritmos de *Boosting*

Luana Faxina Wenceslau¹, Maurilio Martins Campano Junior^{1,2},
Linnyer Beatrys Ruiz Aylon¹

¹Manna Team

Universidade Estadual de Maringá (UEM)

Programa de Pós-Graduação em Ciência da Computação (PCC)

Maringá - PR - Brazil

²Engenharia de Software – UniCesumar

Maringá – PR – Brazil

maurilio.campanojr@gmail.com, lbruiz@uem.br

Abstract. *Teaching Ensemble Learning algorithms poses pedagogical challenges due to the mathematical abstraction involved. This paper presents “Gatinho Classificador”, an educational game that visually simulates the operation of the CatBoost algorithm. The proposal gamifies the training, prediction, and validation stages, allowing students to interact with the concept of Boosting through an autonomous agent. The evaluation conducted with 18 undergraduate students, using the PAJDE and Pro-AvaliaJS protocols, indicated excellent acceptance (average 4.7), validating the tool as an effective resource for the intuitive introduction of complex Artificial Intelligence concepts.*

Resumo. *O ensino de algoritmos de Ensemble Learning impõe desafios pedagógicos devido à abstração matemática envolvida. Este artigo apresenta o “Gatinho Classificador”, um jogo educativo que simula visualmente o funcionamento do algoritmo CatBoost. A proposta gamifica as etapas de treinamento, predição e validação, permitindo ao aluno interagir com o conceito de Boosting através de um agente autônomo. A avaliação realizada com 18 graduandos, utilizando os protocolos PAJDE e Pro-AvaliaJS, indicou excelente aceitação (média 4,7), validando a ferramenta como um recurso eficaz para a introdução intuitiva de conceitos complexos de Inteligência Artificial.*

1. Introdução

A educação contemporânea tem passado por transformações significativas, impulsionada pela inserção de tecnologias digitais que possibilitam práticas pedagógicas mais interativas e personalizadas. Segundo a Unicep (2024), esses recursos contribuem para a criação de espaços de aprendizado dinâmicos, capazes de atender a diferentes ritmos cognitivos e democratizar o acesso a conteúdos de qualidade, independentemente do contexto geográfico ou socioeconômico do estudante.

No âmbito da Ciência da Computação, o uso de tais ferramentas torna-se essencial, especialmente no ensino de tópicos abstratos como a Inteligência Artificial (IA). Compreender os fundamentos da IA é um requisito crescente na formação tecnológica; no

entanto, a área enfrenta desafios pedagógicos relacionados à dificuldade de visualização dos processos matemáticos envolvidos, o que frequentemente resulta na desmotivação discente e na dificuldade de consolidação do aprendizado [Russell and Norvig 2020].

A barreira da abstração teórica reflete-se diretamente nos índices de retenção acadêmica. Esse cenário de evasão elevada não é um caso isolado, sendo um desafio sistêmico amplamente reportado na literatura de educação em computação [Mundim et al. 2025]. Estudos locais corroboram essa tendência, como o realizado por Fukao et al. (2023) na Universidade Estadual de Maringá (UEM), que indicam taxas de evasão acumulada superiores a 50% em cursos de Ciência da Computação e Informática. A pesquisa evidencia que a maior parte das desistências ocorre nos anos iniciais, sugerindo que dificuldades pedagógicas e a falta de adaptação às exigências de base, onde se situam os fundamentos lógicos e matemáticos, são fatores determinantes para o abandono.

Nesse cenário, torna-se imperativo explorar abordagens que promovam o engajamento e facilitem a compreensão conceitual. Os jogos digitais educativos surgem como alternativas promissoras, pois combinam aprendizado ativo com elementos lúdicos (*Game-Based Learning*). Conforme aponta Prensky (2001), essa abordagem favorece a assimilação de conceitos complexos por meio de experiências acessíveis e interativas, transformando a abstração técnica em prática concreta.

O problema específico abordado neste trabalho reside na dificuldade de ensino de algoritmos de classificação supervisionada baseados em *Ensemble Learning*, como o *CatBoost*. Apesar de sua relevância e eficiência na indústria, o *CatBoost* apresenta alta complexidade teórica, envolvendo conceitos de árvores de decisão e otimização de gradiente que são desafiadores para estudantes iniciantes.

Como solução, este artigo propõe o desenvolvimento e avaliação do jogo educativo “Gatinho Classificador”. A ferramenta busca ensinar o funcionamento do algoritmo *CatBoostClassifier* de forma visual, onde o jogador atua como tutor de um personagem que aprende a classificar objetos (brinquedos) com base em atributos visuais. Diferentemente de abordagens que tratam o algoritmo como uma “caixa preta”, a proposta simula as etapas de treinamento, predição e validação (*feedback*), permitindo que o aluno visualize o processo de aprendizado incremental.

O objetivo deste trabalho é demonstrar como a gamificação pode atuar como um recurso eficaz para reduzir a complexidade percebida de algoritmos de *Machine Learning*, contribuindo para o ensino prático de IA e, potencialmente, para a mitigação das dificuldades de aprendizado nos cursos de computação.

2. Fundamentação Teórica e Trabalhos Relacionados

O desenvolvimento de ferramentas de apoio ao ensino de Computação exige a integração entre teorias pedagógicas e tecnologias de implementação. O uso de jogos digitais (*Game-Based Learning*) consolidou-se como estratégia para promover o engajamento e a aprendizagem significativa. Segundo Prensky (2001), jogos estimulam o pensamento crítico e a resolução de problemas, enquanto Gee (2003), defende que a experimentação por tentativa e erro em ambientes simulados favorece a compreensão de conceitos técnicos complexos.

Para a construção do artefato proposto, utilizou-se a *Unity Engine* e a língua-

gem C#, ferramentas que permitem a criação de interfaces gráficas interativas e sistemas de *feedback* imediato, elementos essenciais para a eficácia de jogos sérios educativos [Schell 2008].

2.1. Fundamentos: Aprendizado Supervisionado e CatBoost

O Aprendizado de Máquina Supervisionado baseia-se no treinamento de algoritmos com exemplos rotulados, visando a generalização para novos dados [Russell and Norvig 2020]. Dentro deste espectro, destacam-se os métodos de *ensemble learning*, que combinam múltiplos modelos para aumentar a precisão preditiva.

O foco deste trabalho recai sobre o *CatBoost (Categorical Boosting)*, um algoritmo baseado em Árvores de Decisão com *Gradient Boosting (GBDT)*. O *CatBoost* diferencia-se por tratar nativamente variáveis categóricas e utilizar esquemas de permutação para evitar *overfitting* [Prokhorenkova et al. 2018]. A escolha deste algoritmo como objeto de ensino justifica-se por sua alta relevância na indústria e pela dificuldade inerente em ensinar, de forma visual, o conceito de refinamento incremental de erros (*boosting*).

2.2. Trabalhos Relacionados e Análise Comparativa

A literatura acadêmica evidencia uma consolidação do uso de jogos digitais como estratégia pedagógica no ensino de Inteligência Artificial. Michael e Chen (2005) argumentam que jogos sérios desempenham papel fundamental ao educar, treinar e informar estudantes sobre tarefas complexas. Essa percepção é corroborada por Wollowski et al. (2016), que em levantamento junto aos participantes da conferência *Educational Advances in Artificial Intelligence (EAAI)*, identificaram que 68% dos educadores já incorporavam jogos e quebra-cabeças em suas metodologias de ensino.

Nesse contexto, destacam-se iniciativas corporativas e ferramentas acessíveis. Sapci and Sapci (2020) analisam o ecossistema de ferramentas do Google, como o *Teachable Machine*, que permite a criação rápida de classificadores; o *Quick Draw*, que utiliza redes neurais para reconhecimento de desenhos; e o *AIY Projects*, focado em hardware de baixo custo para processamento de voz e imagem. Similarmente, Yim and Su (2025) catalogam diversas abordagens voltadas para o público infantil, integrando conceitos de IA a ambientes de programação como *Scratch* e *Python*.

No âmbito da pesquisa acadêmica aplicada ao Ensino Médio e Superior, propostas mais robustas têm sido desenvolvidas. Wang et al. (2022) apresentam o *ARIN-561*, um jogo educacional cujos estudos piloto indicaram uma correlação positiva entre o tempo de interação no jogo e a construção do conhecimento sobre IA. Já o projeto *Deep Neural Tower*, descrito por Petrov-Valchev et al. (2024), utiliza a *Unity Engine* e o *ML-Agents* para ensinar conceitos de redes neurais e sistemas de recompensa, demonstrando resultados positivos em engajamento e usabilidade com estudantes do ensino médio.

Entre as ferramentas mais difundidas, o *AI for Oceans* [Code.org 2026] introduz visualmente a classificação supervisionada através da diferenciação entre peixes e lixo marinho, enquanto o *Machine Learning for Kids* [Lane 2026] foca na integração técnica de modelos reais com blocos de programação.

Embora estas iniciativas sejam eficazes para a introdução geral ao tema, elas frequentemente abstraem o funcionamento interno do algoritmo (abordagem “caixa-preta”).

O “Gatinho Classificador” busca preencher esta lacuna ao simular visualmente o processo de *boosting*. A Tabela 1 apresenta uma comparação estruturada entre a proposta deste trabalho e as ferramentas correlatas.

Tabela 1. Comparação entre jogos educativos para ensino de IA

Característica	Gatinho Classificador	AI for Oceans	ML for Kids
Algoritmo Foco	<i>CatBoost</i> (Simulação de <i>boosting</i>)	Classificação Geral	Modelos Supervisionados
Visualização	Interna (Personagem aprende visualmente)	Externa (Apenas resultado)	Externa (Caixa preta)
Interação	Interação direta com agente autônomo	Classificação de imagens	Blocos de programação
Mecânica	<i>Drag-and-drop</i> e Reforço Incremental	Classificação rápida	Treinamento e Aplicação
Público-Alvo	Ensino Médio/Superior	Infantil/Iniciante	Ensino Fundamental/Médio

A análise da Tabela 1 evidencia que, enquanto as iniciativas anteriores focam na acessibilidade para o público infantil ou na aplicação prática de modelos prontos, o presente trabalho aprofunda a abordagem pedagógica ao simular o funcionamento de um algoritmo de *ensemble*.

O diferencial central reside na presença de um personagem autônomo que reage a erros e acertos. Essa mecânica, aliada ao *feedback* narrativo em *pop-ups*, reforça o entendimento de como modelos supervisionados ajustam seus parâmetros a partir de novos dados rotulados, promovendo uma imersão que favorece a fixação do conteúdo [Gee 2003, Prensky 2001]. Dessa forma, o jogo posiciona-se em uma camada intermediária entre a ludicidade acessível e o rigor técnico necessário para o ensino de Computação em níveis mais avançados.

3. Metodologia e Design do Jogo

A metodologia adotada para o desenvolvimento do jogo educativo “Gatinho Classificador” seguiu um modelo iterativo, compreendendo as etapas de concepção pedagógica, modelagem computacional e implementação técnica. O objetivo central da proposta é a transposição didática de conceitos complexos de Aprendizado de Máquina Supervisionado, especificamente o algoritmo *CatBoostClassifier*, para uma experiência lúdica e interativa. A seguir, são detalhadas as estratégias de design instrucional e a arquitetura técnica da solução.

3.1. Concepção Pedagógica e Metáforas Visuais

A premissa fundamental do jogo baseia-se na Aprendizagem Ativa, em que o jogador assume o papel de tutor de um agente inteligente (o personagem Gatinho). A narrativa foi estruturada para criar uma analogia direta entre as ações do jogador e o ciclo de vida de um modelo de classificação supervisionada. Para mitigar a abstração matemática de algoritmos de *Gradient Boosting*, foram estabelecidas as seguintes metáforas operacionais:

- O Agente (Modelo): O gatinho representa o classificador que inicia o processo sem conhecimento prévio e deve ter seus parâmetros ajustados.
- O Cenário (Dataset): Os objetos espalhados pelo ambiente representam as instâncias do conjunto de dados, possuindo atributos visuais (cor e forma) que servem como *features*.
- A Interação (Rotulagem): A ação de apresentar um brinquedo ao gatinho simula a etapa de rotulagem de dados, fornecendo exemplos positivos e negativos.

O *design* do jogo foi dividido em três fases distintas, alinhadas às etapas clássicas de um pipeline de *Machine Learning*: (1) Treinamento, na qual exemplos iniciais são fornecidos; (2) Exploração/Predição, em que o agente aplica o conhecimento adquirido em novos dados; e (3) Validação, onde o jogador fornece *feedback* corretivo, reforçando ou penalizando as decisões do modelo.

3.2. Mecânicas de Jogo e Simulação do Algoritmo

Dada a complexidade matemática do algoritmo *CatBoost* (baseado em árvores de decisão e otimização de gradiente), optou-se por uma implementação pedagógica que simula o comportamento conceitual do *boosting* em vez de uma execução numérica literal. O funcionamento técnico do agente foi estruturado em uma máquina de estados finitos que opera da seguinte maneira:

1. Fase de Observação (Treinamento Inicial): Utilizando a mecânica de *drag-and-drop*, o jogador apresenta dois objetos de classes distintas ao agente. O sistema registra os atributos (cor) na memória interna do agente, criando as regras de decisão iniciais.
2. Fase de Inferência (Predição): O agente explora o cenário de forma autônoma. Ao encontrar um objeto não classificado, o algoritmo interno compara seus atributos com os registros da memória. A decisão é baseada em uma heurística de similaridade: se a cor é conhecida e associada a uma classe positiva, o agente prediz que o objeto é um “brinquedo”; caso contrário, classifica como “não brinquedo” ou “desconhecido”.
3. Fase de Reforço (Correção): Após a predição, o sistema solicita a validação do jogador via *pop-ups*. Esta etapa é crucial para ilustrar o conceito de *boosting*: cada erro corrigido pelo jogador atualiza a base de conhecimento do agente, refinando as regras de decisão para iterações futuras. O acúmulo de acertos e a correção de erros simulam o ajustamento de pesos característico dos modelos de aprendizado sequencial.

3.3. Implementação Técnica

O protótipo foi desenvolvido utilizando a *Unity Engine* (versão 6.x) e a linguagem C#, escolhidas pela versatilidade no desenvolvimento 2D multiplataforma. A arquitetura do software foi concebida de forma modular para facilitar a manutenção e a escalabilidade.

Os principais componentes do sistema incluem:

- Controlador do Agente: *Script* responsável pela lógica de IA, incluindo a máquina de estados (exploração, análise e aguardo de validação) e a gestão da memória de aprendizado. Este componente calcula a “confiança” do agente baseada no histórico de interações.

- Gerenciador de Interação: Implementa a física de arrastar e soltar (*drag-and-drop*) e detecta a colisão entre os objetos manipuláveis e o agente durante a fase de treinamento.
- Interface de *Feedback*: Gerencia a exibição dinâmica de mensagens e *pop-ups*. Este módulo traduz os estados internos do algoritmo (ex: “analisando atributo cor”) para linguagem natural, permitindo que o jogador visualize o “raciocínio” do modelo.

Os recursos gráficos (*sprites* e cenários) foram desenvolvidos em estilo *pixel art* para garantir uma estética acessível e leve, compatível com diversos dispositivos, utilizando ferramentas de IA generativa para otimização do fluxo de produção de *assets*.

3.4. Roteiro de Aplicação em Sala de Aula

O “Gatinho Classificador” não foi projetado para substituir a instrução formal, mas para atuar como um *Organizador Prévio* ou ferramenta de fixação. Com base na experiência de desenvolvimento, sugere-se o seguinte roteiro para docentes de Inteligência Artificial:

1. **Introdução Teórica:** O docente apresenta o problema de classificação.
2. **Gameplay:** Os alunos jogam individualmente ou em duplas focando na “Fase de Validação”.
3. Discussão Dirigida: O professor conecta a mecânica do jogo à teoria questionando os alunos sobre o que acontece na memória do gatinho ao ser corrigido.
4. Formalização Matemática: Apresentação das fórmulas do *CatBoost*, agora com os alunos possuindo um modelo mental concreto do processo iterativo.

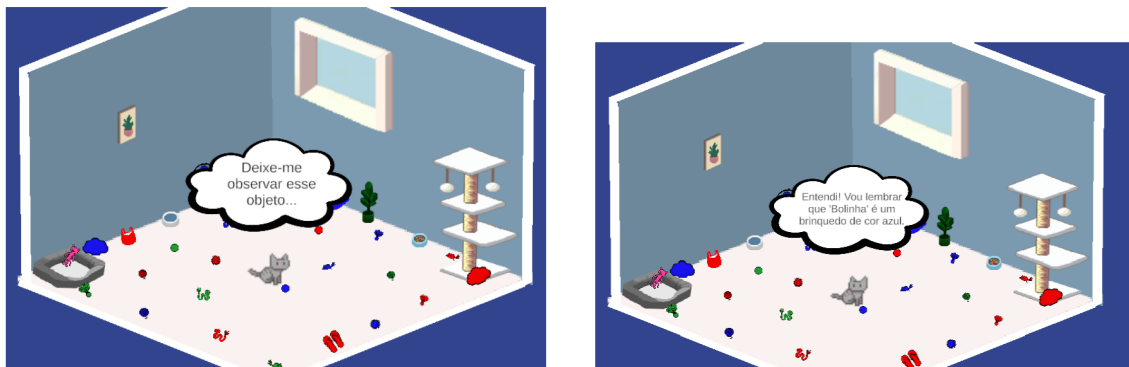
4. Implementação e Mecânicas de Aprendizagem

O protótipo do jogo “Gatinho Classificador” foi desenvolvido utilizando a *Unity Engine* (2D) e a linguagem C#. A implementação seguiu um modelo iterativo, resultando em uma arquitetura modular na qual o agente (personagem) opera através de uma máquina de estados finitos, simulando o comportamento de um classificador supervisionado. O ambiente gráfico e a interface foram projetados para reduzir a carga cognitiva, permitindo que o aluno foque exclusivamente na lógica do algoritmo. A jogabilidade foi estruturada para espelhar as etapas de um pipeline de *Machine Learning*: Treinamento, Inferência (Exploração) e Refinamento (Validação).

4.1. Fase de Treinamento e Representação Interna

Nesta etapa inicial, o jogador atua como um oráculo, fornecendo exemplos rotulados ao sistema. Através de uma mecânica de arrastar e soltar, o usuário apresenta objetos com atributos visuais distintos (cores) ao personagem. O sistema processa esses insumos e atualiza a “memória” do agente, criando uma associação preliminar entre o atributo (ex: cor azul) e a classe alvo (“brinquedo” ou “não brinquedo”).

Para tornar esse processo abstrato em algo concreto, o jogo utiliza metáforas visuais. Conforme ilustrado na Figura 1, ao analisar um objeto, o agente exibe um balão de pensamento contendo a regra de decisão recém-adquirida. Pedagogicamente, isso representa a atualização dos parâmetros do modelo, permitindo ao aluno visualizar o conceito de aprendizado incremental.



(a) Interação de rotulagem (Input).

(b) Representação do estado interno.

Figura 1. Fase de Treinamento: (a) O jogador apresenta o exemplo; (b) O agente visualiza a regra aprendida.

4.2. Exploração, Validação e o Conceito de Boosting

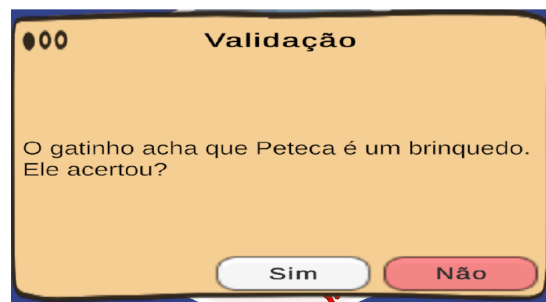
Após a fase inicial, o agente entra em modo autônomo (Fase de Exploração), navegando pelo cenário para identificar novos objetos. O algoritmo de movimentação simula a etapa de inferência, em que o modelo aplica as regras aprendidas em dados não vistos. Ao encontrar um objeto, o agente realiza uma predição baseada em uma árvore de decisão simplificada (ex: “Se cor conhecida → Classificar”).

O diferencial pedagógico da proposta reside na Fase de Validação. Após cada predição do agente, o jogador recebe um *pop-up* solicitando a confirmação do resultado (Figura 2).

- Erro e Refinamento: Se o agente erra e o jogador corrige, o sistema aciona uma rotina de retreinamento com o exemplo negativo.
- Conceito de *Boosting*: Após uma sequência de interações, o jogo introduz explicitamente o conceito de *Boosting*. O sistema explica, através de diálogos contextuais, que o erro anterior foi essencial para calibrar o modelo, traçando um paralelo direto com algoritmos que otimizam o aprendizado focando nas instâncias de difícil classificação, como o *CatBoost* [Prokhorenkova et al. 2018, Russell and Norvig 2020].



(a) Agente realizando inferência.



(b) Validação do conceito de *Boosting* pelo jogador.

Figura 2. Ciclo de Refinamento: (a) O agente aplica o conhecimento em novos dados; (b) O jogador valida a predição, ajustando o modelo.

Esta mecânica de validação ativa transforma o erro, frequentemente frustrante em jogos tradicionais, em uma ferramenta construtiva de aprendizado, alinhando-se aos princípios de *Game-Based Learning* defendidos por [Gee 2003].

5. Avaliação e Análise de Resultados

A validação do jogo “Gatinho Classificador” foi conduzida por meio de um estudo de caso exploratório com viés quantitativo e qualitativo. O objetivo foi mensurar a qualidade pedagógica, a usabilidade e o potencial de engajamento da ferramenta proposta.

5.1. Metodologia de Avaliação

Para garantir robustez na análise, adotou-se uma abordagem híbrida utilizando dois instrumentos consolidados na literatura de Informática na Educação:

- PAJDE (Programa de Avaliação de Jogos Digitais Educacionais): Focado no alinhamento entre mecânicas de jogo e objetivos pedagógicos [Santos and Alves 2019].
- Pro-AvaliaJS: Um protocolo voltado à experiência do jogador e eficácia de jogos sérios [de Oliveira et al. 2022].

O estudo contou com a participação de 18 estudantes de graduação dos cursos de Bacharelado em Informática e Ciência da Computação da Universidade Estadual de Maringá (UEM). A escolha deste público-alvo justifica-se pela familiaridade dos discentes com os conceitos fundamentais de lógica e IA, permitindo uma avaliação crítica sobre a transposição didática do algoritmo. A coleta de dados foi realizada via formulário digital após a interação com o protótipo.

5.2. Análise dos Indicadores

Os resultados obtidos demonstram uma convergência positiva entre as duas metodologias aplicadas. A Tabela 2 apresenta a síntese das médias (em uma escala *Likert* de 1 a 5), evidenciando que o jogo atingiu níveis de excelência nas dimensões pedagógicas e de engajamento.

Tabela 2. Síntese comparativa dos resultados (Escala 1–5)

Dimensão Avaliada	PAJDE	Pro-AvaliaJS	Classificação
Aspectos Pedagógicos	4,8	4,7	Excelente
Usabilidade	4,6	4,7	Muito Bom
Engajamento/Motivação	4,8	4,8	Excelente
Acessibilidade	4,0	4,1	Regular/Bom

A análise dos dados revela uma forte convergência entre os protocolos PAJDE e Pro-AvaliaJS, conferindo confiabilidade aos resultados obtidos. Destaca-se a dimensão Aspectos Pedagógicos como um dos pilares de sucesso do artefato, com médias de 4,8 e 4,7, respectivamente. A análise detalhada dos itens deste domínio aponta que a metáfora visual do gatinho foi eficaz na materialização de conceitos abstratos: o item referente à percepção de aprendizado alcançou média 4,9, enquanto a clareza do *feedback* imediato obteve 4,8. Esses índices validam a hipótese de que a visualização do “estado interno” do

agente auxilia na construção de modelos mentais corretos sobre o ciclo de erro e correção do algoritmo.

Simultaneamente, a dimensão Engajamento e Motivação apresentou consistência notável, atingindo média 4,8 em ambos os instrumentos de avaliação. Este resultado é particularmente relevante no contexto de evasão em cursos de Computação, sugerindo que a abordagem gamificada conseguiu mitigar a aridez típica do ensino de algoritmos matemáticos. A proximidade entre as notas pedagógicas e motivacionais indica que o jogo alcançou o equilíbrio desejado em *Serious Games*: ser educativo sem ser entediante, e ser lúdico sem perder o rigor conceitual.

No que tange à Usabilidade, as médias de 4,6 (PAJDE) e 4,7 (Pro-AvaliaJS) demonstram que a interface gráfica e as mecânicas de arrastar e soltar foram compreendidas intuitivamente pelos estudantes. A baixa barreira de entrada na interação permitiu que o esforço cognitivo dos alunos fosse direcionado para a compreensão da lógica do *Boosting*, e não para o aprendizado dos controles do jogo. A fluidez da navegação e a clareza dos elementos visuais foram fundamentais para sustentar a experiência de imersão relatada.

Por outro lado, a dimensão de Acessibilidade revelou-se o ponto crítico da avaliação, com médias de 4,0 e 4,1, classificadas apenas como “Regular/Bom”. Embora não comprometa a viabilidade do protótipo atual, essa discrepância em relação às demais dimensões evidencia uma dívida técnica importante. As observações qualitativas dos participantes corroboram os dados quantitativos, apontando a necessidade urgente de implementação de ajustes de contraste para daltônicos e suporte a tecnologias assistivas (leitores de tela), elementos que foram priorizados para os trabalhos futuros.

5.3. Alinhamento Conceitual e Limitações do Escopo

Uma discussão crítica necessária refere-se ao nível de abstração adotado. O algoritmo *CatBoost* real envolve cálculos complexos de gradiente e regularização que são inviáveis de serem gamificados fielmente sem comprometer a ludicidade. Portanto, os resultados validam a eficácia do jogo na transmissão da intuição conceitual do *Boosting* (aprendizado sequencial, correção de erros e pesos), e não da implementação técnica matemática.

A avaliação indica que o jogo funciona efetivamente como um “organizador prévio”, preparando o estudante cognitivamente para, posteriormente, aprofundar-se na matemática do algoritmo. A alta taxa de engajamento (4,8) sugere que a ferramenta é eficaz para mitigar a barreira inicial de desinteresse ou intimidação frequentemente associada ao ensino de tópicos avançados de Inteligência Artificial.

5.4. Ameaças à Validade

Embora os resultados sejam promissores, é necessário considerar limitações inerentes ao desenho experimental deste estudo de caso:

- Tamanho da Amostra: O grupo de 18 participantes, embora suficiente para uma avaliação preliminar de usabilidade e percepção pedagógica, não permite generalizações estatísticas robustas para a população de estudantes de computação em geral.
- Efeito Novidade: A alta taxa de engajamento pode ter sido influenciada pelo fator novidade, na qual o interesse dos alunos é amplificado temporariamente pelo uso

de uma tecnologia diferente da aula expositiva tradicional, e não necessariamente pela mecânica do jogo em si.

- **Viés de Seleção:** Como a participação foi voluntária e realizada com estudantes que já possuíam afinidade com a área de tecnologia, os resultados podem refletir um viés positivo em relação ao uso de ferramentas digitais.
- **Ausência de Grupo de Controle:** A avaliação focou na percepção subjetiva de aprendizado. A ausência de um pré-teste e pós-teste formal, comparando com um grupo que teve apenas aula teórica, impede a mensuração objetiva do ganho de conhecimento cognitivo sobre o algoritmo *CatBoost*.

O reconhecimento dessas ameaças direciona os trabalhos futuros para a condução de experimentos controlados randomizados, visando isolar a variável da gamificação no desempenho acadêmico.

6. Conclusão

O ensino de algoritmos complexos de Inteligência Artificial, como o *Boosting*, impõe desafios significativos devido à abstração matemática envolvida. Este trabalho apresentou o “Gatinho Classificador” como uma resposta a esse desafio, propondo uma abordagem baseada em gamificação e metáforas visuais para tornar tangível o conceito de aprendizado supervisionado incremental.

A principal contribuição da pesquisa foi a validação de que é possível simular a lógica interna de um algoritmo “caixa-preta” (*CatBoost*) através de mecânicas de jogo acessíveis. Os resultados obtidos junto a estudantes de computação indicam que a ferramenta foi eficaz em promover o engajamento (média 4,8 na escala *Likert*) e facilitou a percepção dos conceitos de treinamento, predição e refinamento de modelo. A alta avaliação nos aspectos pedagógicos (4,8) sugere que a visualização do “estado interno” do agente (através dos balões de pensamento e *feedbacks*) auxilia na construção de modelos mentais corretos sobre o funcionamento do algoritmo. Contudo, é importante ressaltar que o jogo atua como um facilitador da intuição conceitual, e não como um substituto para o estudo formal da matemática subjacente aos métodos de otimização de gradiente.

Como trabalhos futuros, visando a evolução da ferramenta e a ampliação de seu impacto educacional, propõem-se as seguintes ações:

- **Aprimoramento da Acessibilidade:** Implementação de modos de alto contraste e suporte a leitores de tela, endereçando as limitações apontadas na avaliação preliminar;
- **Expansão de Cenários:** Desenvolvimento de novas fases que introduzam variáveis adicionais (além da cor), aumentando a complexidade da árvore de decisão simulada;
- **Avaliação de Aprendizagem:** Realização de experimentos com pré-teste e pós-teste para mensurar quantitativamente o ganho de conhecimento teórico, superando a análise puramente perceptiva adotada nesta etapa inicial.

O “Gatinho Classificador” demonstra, portanto, que a interseção entre Design de Jogos e Educação em Computação oferece caminhos promissores para desmistificar a IA, preparando estudantes para lidar com tecnologias avançadas de forma mais intuitiva e motivadora.

Declaração sobre uso de Inteligência Artificial

O conteúdo deste trabalho é de autoria exclusiva dos pesquisadores, não tendo sido utilizadas ferramentas de IA Generativa em sua produção.

Agradecimentos

“Agradecimentos ao Manna Team, a Fundação Araucária de Apoio ao Desenvolvimento Científico e Tecnológico do Estado do Paraná (FA) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) - Brasil (Processo nº 421548/2022-3) pelo apoio”.

Referências

- Code.org (2026). Ai for oceans. Disponível em: <https://code.org/oceans>.
- de Oliveira, R. N., Belarmino, G. D., Minholi, F. S., Rodriguez, C., Goya, D., and Rocha, R. V. (2022). Pro-avaliajs: Protocolo para planejamento e execução da avaliação da reação e aprendizagem de jogos sérios. In *Anais do XXXIII Simpósio Brasileiro de Informática na Educação*, pages 517–527. SBC.
- Fukao, A. T., Colanzi, T. E., Martimiano, L. A., and Feltrim, V. D. (2023). Estudo sobre evasão nos cursos de computação da universidade estadual de maringá. In *Anais do III Simpósio Brasileiro de Educação em Computação*, pages 86–96. SBC.
- Gee, J. P. (2003). What video games have to teach us about learning and literacy. *Computers in entertainment (CIE)*, 1(1):20–20.
- Lane, D. (2026). Machine learning for kids. Disponível em: <https://machinelearningforkids.co.uk/>.
- Michael, D. R. and Chen, S. L. (2005). *Serious games: Games that educate, train, and inform*. Muska & Lipman/Premier-Trade.
- Mundim, P., Silva, T., e Silva, G. B., and Barbosa, D. (2025). Evasão em cursos superiores na área de computação: Um mapeamento sistemático da literatura. In *Anais do XXXIII Workshop sobre Educação em Computação*, pages 1055–1067, Porto Alegre, RS, Brasil. SBC.
- Petrov-Valchev, B., Núñez-Vidal, E., Hristov-Kalamov, N., Zapata-Cáceres, M., and Palacios-Alonso, D. (2024). Deep neural tower: A game to teach ai concepts. In *2024 IEEE 4th International Conference on Advanced Learning Technologies on Education Research (ICALTER)*, pages 1–4.
- Prensky, M. (2001). *Digital Game-Based Learning*. McGraw-Hill, New York.
- Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A. (2018). Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31.
- Russell, S. J. and Norvig, P. (2020). *Artificial Intelligence: A Modern Approach*. Pearson, 4 edition.
- Santos, W. and Alves, L. R. G. (2019). Pajed: Um programa de avaliação de jogos digitais educacionais. *Anais do Seminário de Jogos Eletrônicos, Educação e Comunicação*.

- Sapci, A. H. and Sapci, H. A. (2020). Artificial intelligence education and tools for medical and health informatics students: systematic review. *JMIR Medical Education*, 6(1):e19285.
- Schell, J. (2008). *The Art of Game Design: A book of lenses*. Morgan Kaufmann, Burlington.
- Unicep (2026). Educação e tecnologia: qual o impacto da tecnologia na educação moderna? Disponível em: <https://www.unicep.edu.br/post/educa>
- Wang, N., Greenwald, E., Montgomery, R., and Leitner, M. (2022). Arin-561: An educational game for learning artificial intelligence for high-school students. In *International Conference on Artificial Intelligence in Education*, pages 528–531. Springer.
- Wollowski, M., Selkowitz, R., Brown, L., Goel, A., Luger, G., Marshall, J., Neel, A., Neller, T., and Norvig, P. (2016). A survey of current practice and teaching of ai. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.
- Yim, I. H. Y. and Su, J. (2025). Artificial intelligence (ai) learning tools in k-12 education: A scoping review. *Journal of Computers in Education*, 12(1):93–131.