

LLMs Como Tutores Educacionais em Dispositivos de Borda: Um Benchmark de 16 Modelos Compactos em Raspberry Pi 5

Antônio A. Brum¹, Tiago T. Primo¹, Paulo C. R. Pinho¹, Ulisses B. Correa¹

¹ Universidade Federal de Pelotas (UFPEL)

{aadbrum, tiago.primo, pcrpinho, ulisses}@inf.ufpel.edu.br

Abstract. *Large Language Models (LLMs) may help reduce educational inequalities in Global South countries, where teacher shortages coexist with limited infrastructure and connectivity. This work evaluates compact LLMs as potential components of educational tutoring systems in low-resource environments. Sixteen models with fewer than 4 billion parameters were tested on a Raspberry Pi 5 using elementary-level multiple-choice exams in four subject areas. Results show a trade-off between accuracy and computational cost. Preliminary findings suggest that smaller models are viable for some educational domains, whereas structured reasoning tasks may require larger models.*

Keywords: Large Language Models; Educational Tutoring Systems; Edge Computing; Raspberry Pi; Low-Resource Environments.

Resumo. *Grandes Modelos de Linguagem (LLMs) podem ajudar a reduzir desigualdades educacionais em países do Sul Global, onde a escassez de professores convive com limitações de infraestrutura e conectividade. Este trabalho avalia LLMs compactos como possíveis componentes de sistemas de tutoria educacional em ambientes com poucos recursos. Dezesesseis modelos com menos de 4 bilhões de parâmetros foram testados em uma Raspberry Pi 5 usando provas de múltipla escolha de nível fundamental em quatro áreas do conhecimento. Os resultados mostram um trade-off entre acurácia e custo computacional. Modelos menores funcionam para alguns domínios, enquanto tarefas que exigem raciocínio estruturado tendem a demandar modelos maiores.*

Palavras-Chave: Grandes Modelos de Linguagem; Sistemas de Tutoria Educacional; Computação de Borda; Raspberry Pi; Inteligência Artificial na Educação.

1. Introdução

O acesso ao ensino de qualidade ainda é um problema em países do Sul Global. No Brasil, as turmas da educação pública são frequentemente grandes demais para um único professor, que acaba sobrecarregado e sem condições de oferecer atendimento individualizado. O docente termina dividido entre dezenas de alunos com ritmos e dificuldades distintos, e os estudantes podem ficar desassistidos. Essa realidade pesa mais nos países em desenvolvimento. Nesse contexto, os Grandes Modelos de Linguagem (LLMs, do inglês *Large Language Models*) podem ajudar em sistemas de tutoria: eles armazenam conhecimento, adaptam respostas e podem assumir parte do atendimento que hoje recai sobre o professor.

LLMs têm uma vantagem sobre abordagens tradicionais: personalizam respostas com base na linguagem do aluno e direcionam o estudo a partir do conhecimento prévio do estudante. O problema é o custo. Os melhores modelos exigem boa conectividade com a internet ou hardware caro de adquirir e manter. Como muitas escolas públicas têm baixa ou nenhuma conexão e poucos recursos para GPUs e outros equipamentos, alternativas mais baratas passaram a ser investigadas. O paradigma AIED Unplugged [Rodrigues et al. 2025] propõe tecnologias educacionais baseadas em IA que operem em ambientes desconectados, usando hardware de baixo custo. Estudos recentes mostraram a viabilidade técnica de executar LLMs localmente, investigando latência em deploy offline em smartphones, trade-offs entre qualidade e desempenho em hardware móvel, aplicação de alfabetização em Raspberry Pi com inferência local e interface de voz educacional em Raspberry Pi 5 [Santos et al. 2024, Monteiro et al. 2025, Pinho et al. 2025, Ramires and Primo 2025].

Este trabalho amplia essa linha com um benchmark maior. Avaliamos **16 modelos** de linguagem com até 4 bilhões de parâmetros, de cinco famílias arquiteturais distintas (Qwen, Gemma, Granite, DeepSeek e LFM), executados em uma Raspberry Pi 5 com 16 GB de RAM via Ollama. A Raspberry Pi 5 foi escolhida pelo custo baixo, pelo ecossistema de software disponível e pela possibilidade de expansão com vídeo, monitor, microfone e câmera. A RAM de 16 GB, combinada com LLMs pequenos, permite manter vários modelos ativos ao mesmo tempo, além de outros aplicativos necessários em um ambiente de tutoria.

O conjunto de testes foi construído a partir do Exame Nacional para Certificação de Competências de Jovens e Adultos (ENCCEJA), avaliação oficial do governo brasileiro disponível em <https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/encceja/provas-e-gabaritos>, que certifica competências equivalentes ao Ensino Fundamental e Médio. Usamos questões de nível fundamental de três anos (2018, 2019 e 2020), totalizando 258 questões válidas de múltipla escolha em quatro áreas do conhecimento: Matemática, Ciências Naturais, Língua Portuguesa e História & Geografia.

As contribuições são: (1) benchmark de 16 LLMs leves em Raspberry Pi com métricas de acurácia e desempenho computacional; (2) proposta do Coeficiente Custo-Desempenho (CPC, do inglês *Cost-Performance Coefficient*), métrica integrada que quantifica o trade-off entre qualidade e eficiência computacional.

Este trabalho concentra-se na avaliação objetiva das respostas, isto é, na capacidade do modelo de selecionar a alternativa correta em questões de múltipla escolha. Não avaliamos a qualidade didática das explicações geradas, a adequação pedagógica da linguagem nem a capacidade do modelo de adaptar o ensino ao perfil do estudante. Esses pontos são necessários para avaliar um tutor educacional completo, mas ficaram fora do escopo deste estudo.

O restante deste artigo está organizado da seguinte forma. A Seção 2 revisa os trabalhos relacionados. A Seção 3 detalha a metodologia de seleção dos modelos, o protocolo de avaliação, as métricas coletadas e o ambiente experimental. A Seção 4 apresenta os resultados quantitativos. A Seção 5 discute as implicações dos achados. A Seção 6 reconhece as limitações do estudo. A Seção 7 conclui com recomendações práticas e

direções futuras.

2. Trabalhos Relacionados

Tecnologias educacionais inteligentes em contextos de recursos limitados têm recebido mais atenção na literatura, especialmente quanto ao uso de modelos de linguagem em hardware de borda. Esta seção revisa três linhas de pesquisa relacionadas a este trabalho: Sistemas Tutores Inteligentes e sua evolução com LLMs, execução de modelos de linguagem em dispositivos de borda, e avaliação de LLMs em português.

Os Sistemas Tutores Inteligentes (ITS) tradicionais dependiam de modelos especializados por domínio, o que limitava sua escalabilidade [Silva and Andrade 2025]. Com os LLMs, abordagens híbridas passaram a combinar capacidade linguística generalista com tutoria humana, aumentando o engajamento dos estudantes [Pereira et al. 2024, Pereira et al. 2025]. Entre as estratégias de tutoria com LLMs estão a geração de feedback automatizado para ensino de programação [Pimentel et al. 2025] e diálogos socráticos para estímulo metacognitivo [Camelo and Siebra 2025]. Um estudo testou cinco modelos locais via Ollama com o GPT-4o como controle remoto: os modelos locais foram consistentemente inferiores ao de nuvem [Pimentel et al. 2025]. Camelo e Siebra construíram um tutor metacognitivo com um modelo de 70 bilhões de parâmetros acessado via API em nuvem. Apesar do valor pedagógico, ambas as abordagens dependem de infraestrutura de nuvem, o que as torna inacessíveis em regiões sem conectividade persistente.

Executar LLMs em dispositivos de borda é uma resposta a essa limitação. Um estudo avaliou três modelos leves (TinyLlama, RedPajama e Qwen2) em um smartphone fazendo deploy offline em Android [Santos et al. 2024]. Embora tecnicamente viáveis, os modelos tiveram latências de 1 a 2 minutos e consumo de memória entre 3,2 e 4,1 GB, próximo ao limite da RAM do dispositivo. Outro estudo expandiu essa linha com seis modelos compactos no mesmo dispositivo, medindo desempenho computacional e qualidade das respostas em 180 questões de biologia [Monteiro et al. 2025]. Encontraram um trade-off claro: modelos mais leves respondiam mais rápido com menor completude; modelos maiores eram mais acurados com latências elevadas. Outros trabalhos testaram o TinyLlama em uma Raspberry Pi 3 com apenas 1 GB de RAM, obtendo tempos de resposta de 40 minutos, e propuseram um pipeline de interação por voz combinando Whisper com Gemma 3 quantizada em hardware simulando uma Raspberry Pi 5, com tempos de resposta entre 5 e 45 segundos e consumo de aproximadamente 7 GB de RAM [Pinho et al. 2025, Ramires and Primo 2025]. Apesar desses avanços, esses trabalhos compartilham lacunas: avaliação restrita a no máximo seis modelos e falta de um índice integrado de custo-benefício.

A avaliação de LLMs em português é a terceira linha relevante. Um estudo avaliou três modelos leves em questões de Ciências da Terra com 196 avaliadores humanos que julgaram acurácia, completude e legibilidade em escala Likert [Barros et al. 2025]. A legibilidade foi significativamente inferior à acurácia, indicando que respostas corretas podem ser linguisticamente complexas demais para o público-alvo, e o Qwen2 foi o modelo mais equilibrado. Outro trabalho investigou a geração de questões de matemática em português com o Gemma-2-2b-it em smartphone Android, com taxa de erro de 24% nas questões geradas [Santos et al. 2025]. Também foram explorados SLMs com RAG para geração de questões do POSCOMP, mostrando que o RAG beneficia especialmente mo-

delos menores ao fornecer contexto de domínio especializado [Araujo et al. 2025]. Esses trabalhos compartilham limitações: ausência de métricas de hardware detalhadas, falta de protocolos de avaliação objetiva em múltiplas áreas, e foco em tarefas isoladas sem integrar desempenho acadêmico e eficiência computacional.

Este trabalho parte dessas lacunas. Em relação aos estudos de tutoria com LLMs, removemos a dependência de nuvem com execução local completa em hardware de baixo custo. Quanto aos trabalhos em dispositivos edge, ampliamos a escala para 16 modelos e incluímos métricas de throttling térmico e um índice integrado de custo-benefício. Em relação à avaliação em português, usamos um protocolo objetivo em quatro áreas do conhecimento e coletamos métricas computacionais na mesma execução. Com isso, educadores e gestores públicos passam a ter dados comparáveis para decidir quais modelos adotar em contextos de recursos limitados.

3. Metodologia

A metodologia foi organizada para permitir replicação. Ela combina critérios de seleção de modelos, um conjunto de questões de múltipla escolha, um protocolo de avaliação padronizado e a coleta de métricas educacionais e computacionais para cada modelo.

As métricas foram organizadas em três categorias. As de **Qualidade** compreendem a acurácia objetiva (comparação com gabarito). As de **Inferência** são coletadas em duas camadas: métricas de experiência do usuário, que incluem o tempo total (*wall-clock*) e o tempo até o primeiro token (TTFT), e métricas internas reportadas pela API Ollama (durações de *prefill* e geração, contagem de tokens de entrada e saída). O *throughput* global é a razão entre o total de tokens de saída gerados e o tempo acumulado de geração em todas as questões. As de **Hardware** são amostradas a cada 500 ms por sondas que monitoram: uso de CPU do processo Ollama e subprocessos, memória RAM atribuída ao modelo somada ao consumo basal do sistema operacional, temperatura da CPU, frequência de operação e estado de *throttling*.

A seleção dos 16 modelos seguiu três critérios: adequação ao ambiente de execução, disponibilidade no Ollama e diversidade arquitetural. Consideramos apenas modelos com até 4 bilhões de parâmetros. Modelos maiores podem ser executados na Raspberry Pi 5 com 16 GB de RAM, mas a ausência de aceleração por GPU resulta em tempos de inferência elevados, comprometendo a responsividade necessária para tutoria. A restrição à plataforma Ollama garante reprodutibilidade do deployment e padronização da execução. Selecionamos modelos de cinco famílias: Qwen (Alibaba), Gemma (Google), Granite (IBM), DeepSeek (DeepSeek-AI) e LFM (Liquid AI). Os modelos abrangem entre 350 milhões e 4 bilhões de parâmetros em diferentes configurações fornecidas por seus desenvolvedores, permitindo analisar o equilíbrio entre qualidade e eficiência computacional sem limitar-se a uma família específica.

O conjunto de dados foi construído a partir do ENCCEJA. As provas foram obtidas em PDF e as questões de múltipla escolha foram extraídas por scripts automatizados. Questões com imagens ou gráficos essenciais para a resolução foram excluídas. Questões com tabelas foram formatadas para representação textual. Essas exclusões e formatações garantem comparabilidade justa entre modelos, eliminando a variável de processamento visual, que não está no escopo deste trabalho. A Tabela 1 detalha a distribuição das questões por área do conhecimento.

Tabela 1. Distribuição das questões por área do conhecimento.

Área do Conhecimento	Questões Originais	Questões Válidas
Matemática	90	66
Ciências Naturais	90	63
Língua Portuguesa	90	70
História & Geografia	90	59
Total	360	258

O uso de três anos reduz o viés de um único exame, que poderia conter questões atípicas ou particularmente fáceis ou difíceis. Também aumenta o número de questões avaliadas e permite observar melhor o desempenho de cada modelo em diferentes domínios.

O protocolo de avaliação foi padronizado para garantir consistência e isolamento entre experimentos. Cada modelo respondeu sequencialmente a todas as questões usando um template de persona “tutor”. Esse template instrui o modelo a assumir o papel de tutor educacional para crianças de 6 a 13 anos, explicando a questão em linguagem simples com até três passos curtos, relacionando a explicação à alternativa correta e mantendo um formato estruturado com seções “Explicação:” e “Resposta correta:”. O uso de personas e templates estruturados se apoia em evidências de que personas alteram o comportamento dos modelos de linguagem [Luz de Araujo and Roth 2025, Yeo et al. 2025] e de que formatos de saída restritos melhoram a consistência e a extração de respostas [Liu et al. 2024]. A estratégia de explicação passo a passo segue técnicas de raciocínio estruturado que melhoram a qualidade das respostas [Wei et al. 2022]. A adaptação ao público-alvo (crianças em idade escolar) baseia-se em princípios de personalização pedagógica em chatbots educacionais [Palahan 2025].

A execução dos experimentos foi automatizada por um *pipeline* de *scripts* que itera sobre os 16 modelos e, para cada modelo, sobre as 12 provas (4 áreas $\times$ 3 anos), totalizando 192 execuções. Cada execução tem quatro etapas: (1) coleta de *baseline* de hardware ocioso; (2) pré-carregamento (*warmup*) do modelo, para não incluir o tempo de carregamento dos pesos nas métricas de inferência; (3) execução sequencial de todas as questões da prova com monitoramento contínuo de recursos; e (4) descarregamento do modelo da memória. Entre execuções consecutivas, um procedimento de *cooldown* térmico aguarda a temperatura da CPU retornar abaixo de 50 °C. Isso garante isolamento térmico entre experimentos, de modo que o delta de temperatura de cada execução reflita o impacto térmico do modelo avaliado.

Como as respostas dos modelos continham raciocínio e alternativa final, um script de extração identifica automaticamente a alternativa escolhida a partir de padrões textuais na resposta gerada. Após a extração, a resposta é comparada ao gabarito oficial para o cálculo da acurácia.

3.1. Métrica Derivada: CPC

O *Cost-Performance Coefficient* (CPC) integra qualidade e eficiência em uma única métrica que quantifica o custo-benefício de cada modelo:

$$\text{CPC} = \frac{w_{\text{acc}} \cdot \text{acc}_{\text{raw}} + w_{\text{tok}} \cdot \text{tps}_{\text{norm}}}{\alpha \cdot t_{\text{norm}} + \beta \cdot (1 - f_{\text{norm}}) + \gamma \cdot m_{\text{norm}}} \quad (1)$$

onde acc_{raw} é a acurácia, tps_{norm} é o throughput normalizado, t_{norm} é o tempo total normalizado, f_{norm} é a frequência relativa ao limite (indicadora de *throttling*) e m_{norm} é o uso de RAM relativo ao total. Os pesos adotados foram: $\alpha = 0,5$, $\beta = 0,2$, $\gamma = 0,3$ (custo); $w_{\text{acc}} = 0,8$, $w_{\text{tok}} = 0,2$ (benefício). A normalização do CPC é relativa ao *dataset* completo.

O CPC mede “acertos por unidade de recurso computacional”, de forma análoga ao conceito de eficiência energética. O numerador representa o valor educacional: acurácia (80% do peso) como capacidade de responder corretamente e throughput (20% do peso) como fluidez da interação. O denominador representa o custo computacional: tempo total de execução (50% do peso) como latência agregada, *throttling* (20% do peso) penalizando modelos que forçam a CPU a reduzir frequência por superaquecimento, e uso de memória (30% do peso) quantificando a pressão sobre os recursos do dispositivo.

Um CPC maior indica melhor custo-benefício: o modelo entrega mais acertos por menos recurso computacional. A métrica compara modelos de escalas distintas em termos de eficiência relativa e mostra que o modelo mais acurado nem sempre é o mais viável. Os pesos são uma escolha de projeto e afetam diretamente o ranking produzido pelo CPC. Portanto, a métrica não deve ser lida como medida absoluta de qualidade, mas como uma função de decisão dependente das prioridades de implantação. Em cenários onde a acurácia é prioritária, w_{acc} pode ser aumentado; em dispositivos com memória limitada, γ pode ser elevado para penalizar maior consumo de RAM. Outra ponderação poderia alterar a ordem dos modelos no ranking.

4. Resultados

A análise está organizada em quatro dimensões: desempenho acadêmico global e por área, eficiência computacional e de inferência, impacto térmico no hardware, e a métrica integrada CPC.

Para interpretar os resultados, usamos parâmetros de referência ligados ao contexto educacional e ao uso de recursos computacionais. A nota do ENCCCEJA é calculada pela Teoria de Resposta ao Item (TRI). Estima-se que um estudante precisa acertar entre 12 e 16 questões em 30 para obter a certificação; por isso, tratamos acurácia superior a 50% como satisfatória. Para fluidez, consideramos melhor um TTFT menor e um throughput maior. Para recursos, consideramos desejável que o sistema ocupe pouca RAM e opere longe do limite térmico de 85 °C.

4.1. Acurácia por Área do Conhecimento

Na Tabela 2 estão os 3 modelos mais acurados em cada área. As famílias **Qwen** e **Gemma** concentram quase todas as primeiras posições. Se o critério for apenas precisão, esses modelos são os mais indicados.

A Tabela 3 mostra os modelos com acurácia acima de 50% por área. Há variação significativa de dificuldade entre as áreas: Ciências Naturais e História & Geografia têm as maiores acurácias (máximos de 95,7% e 96,6%), enquanto todos os modelos tiveram dificuldade em Língua Portuguesa. Além disso, poucos modelos foram bem em matemática,

Tabela 2. Três melhores modelos por área do conhecimento.

Área	1º	2º	3º
Ciências Naturais	Qwen3.5:4B (95,7%)	Qwen3:4B (94,3%)	Gemma4:e4B (92,9%)
História & Geografia	Gemma4:e4B (96,6%)	Qwen3:4B (94,9%)	Qwen3.5:4B (93,2%)
Língua Portuguesa	Qwen3.5:4B (79,4%)	Gemma4:e4B (76,2%)	Qwen3:4B (76,2%)
Matemática	Qwen3:4B (87,9%)	Gemma4:e4B (81,8%)	Qwen3.5:4B (80,3%)

sugerindo que o raciocínio necessário para esse tipo de questão ainda não foi incorporado aos modelos pequenos.

Tabela 3. Modelos com acurácia superior a 50% em cada área do conhecimento.

Modelo	Mat	CN	LP	H&G
Qwen3:4B	87,9	94,3	76,2	94,9
Qwen3.5:4B	80,3	95,7	79,4	93,2
Gemma4:e4B	81,8	92,9	76,2	96,6
Gemma4:e2B	62,1	80,0	73,0	83,1
Gemma3:4B	—	72,9	71,4	74,6
Granite4:3B	—	71,4	57,1	79,7
Qwen3:1.7B	—	68,6	58,7	69,5
Qwen3.5:2B	—	58,6	58,7	76,3
Granite3.1-MoE	—	55,7	—	57,6
Granite4:1B	—	55,1	50,8	70,7
Qwen3:0.6B	—	52,9	—	—
Qwen3.5:0.8B	—	—	57,1	—

4.2. Acurácia Global

A Figura 1 apresenta a acurácia média global de cada modelo. Os melhores resultados foram de Qwen3:4B (88,3%), Qwen3.5:4B (87,2%) e Gemma4:e4B (86,9%). O gráfico mostra que a quantidade de parâmetros, sozinha, não explica a qualidade: o **Gemma4:e2B**, por exemplo, fica à frente de vários modelos de 4B e 3B.

4.3. Métricas de Eficiência de Hardware

A Tabela 4 apresenta as métricas de hardware e de inferência agregadas por modelo, ordenadas por throughput médio de tokens por segundo. Os dados consolidam as execuções em todas as áreas do conhecimento e anos de prova.

Tamanho e throughput têm correlação negativa: o Qwen3:0.6B gera 10,3 tok/s; o Qwen3:4B gera apenas 2,5 tok/s. No entanto, arquiteturas especializadas como o Granite3.1-MoE (3B) alcançam 9,4 tok/s, superando modelos menores. O TTFT varia de 5,6 s no Qwen3:0.6B a 70,7 s no Granite4:1B. O consumo de RAM oscila entre 1,2 GB e 10,2 GB (Gemma4:e4B).

A Figura 2 apresenta o tempo médio por questão. O Qwen3:0.6B é o modelo mais rápido (13,5 s/questão), seguido por Gemma3:1B (22,0 s) e Granite4:350M (26,9 s). No extremo oposto, o Qwen3:4B, que também é o mais preciso, requer 671,4 s/questão,

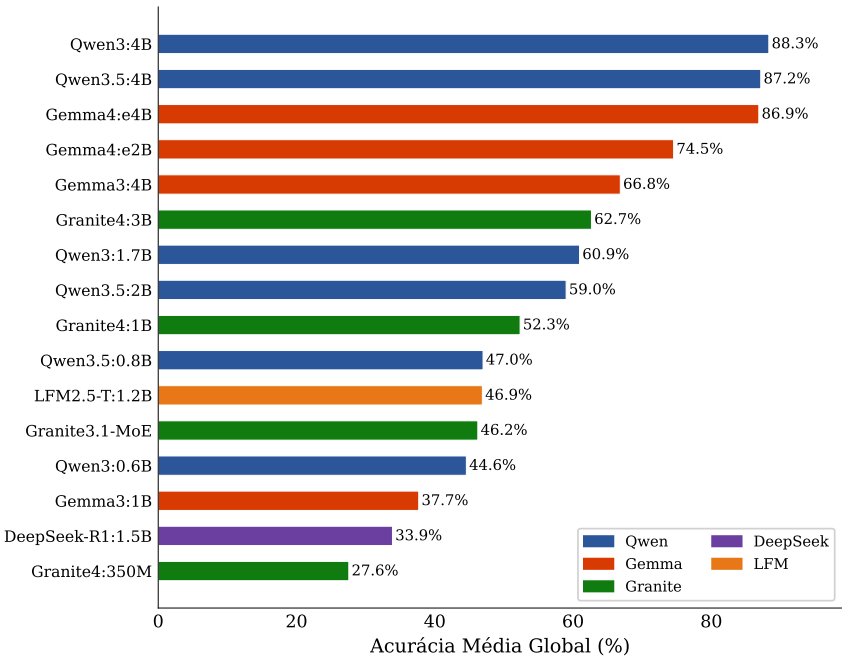


Figura 1. Acurácia média global por modelo. Cores indicam famílias arquiteturais.

cerca de 50 vezes mais que o modelo mais rápido. Outros modelos de alta precisão como Qwen3.5:4B e Gemma4:e4B seguem esse padrão.

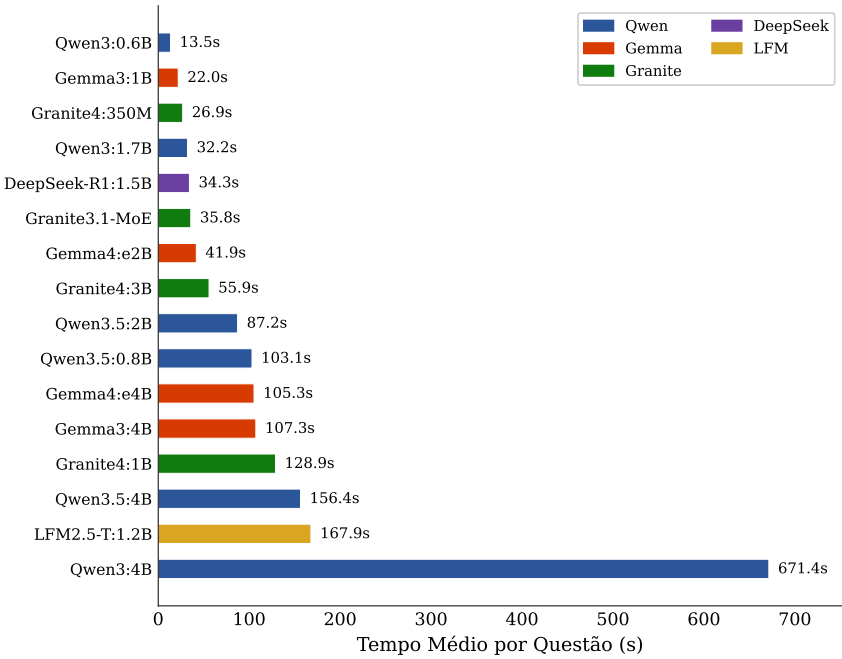


Figura 2. Tempo médio por questão por modelo. Modelos ordenados do mais rápido (topo) ao mais lento. Cores indicam famílias arquiteturais.

Tabela 4. Métricas de hardware consolidadas: throughput, TTFT, RAM, temperatura média e frequência.

Modelo	tok/s	TTFT (s)	RAM (GB)	T _{méd} (°C)	Freq (MHz)
Qwen3:0.6B	10,3	5,6	1,2	69,6	2400
LFM2.5-T:1.2B	9,4	15,1	1,3	80,2	2352
Granite3.1-MoE	9,4	10,8	2,7	74,0	2400
Granite4:350M	6,9	12,9	1,2	72,4	2400
DeepSeek-R1:1.5B	7,8	12,2	1,5	79,3	2306
Gemma3:1B	6,5	8,7	1,8	72,5	2400
Qwen3:1.7B	5,2	13,4	2,0	77,5	2386
Gemma4:e2B	4,3	16,5	7,6	77,8	2367
Granite4:3B	2,9	28,8	2,8	79,4	2352
Qwen3.5:2B	3,0	18,3	4,0	74,3	2399
Qwen3.5:0.8B	2,7	39,5	3,1	75,0	2397
Gemma3:4B	2,5	30,8	5,6	81,8	2253
Gemma4:e4B	2,5	34,5	10,2	81,6	2235
Qwen3:4B	2,5	36,7	3,5	82,3	2300
Qwen3.5:4B	1,7	49,3	5,5	77,6	2398
Granite4:1B	1,5	70,7	3,9	78,5	2373

4.4. Impacto Térmico e Throttling

A Tabela 4 mostra a temperatura média do hardware para cada modelo. O Qwen3:0.6B opera a apenas 69,6°C; modelos de 4B atingem 81–82°C em média.

Throttling térmico é o mecanismo de proteção pelo qual o processador reduz automaticamente sua frequência de operação quando a temperatura atinge limites críticos. Na Raspberry Pi 5, a frequência nominal máxima é de 2400 MHz. Modelos maiores apresentam redução perceptível: Gemma4:e4B opera a 2235 MHz (queda de 6,9%), Gemma3:4B a 2253 MHz (queda de 6,1%), e Qwen3:4B a 2300 MHz (queda de 4,2%).

4.5. Cost-Performance Coefficient (CPC)

A Figura 3 apresenta o CPC de cada modelo. Um CPC mais alto indica melhor equilíbrio entre qualidade (acurácia) e custo computacional (tempo, memória, throttling). Modelos com CPC elevado acertam mais sem sobrecarregar tanto o hardware; modelos com CPC baixo ou têm acurácia insuficiente ou custam demais para executar.

O Qwen3:0.6B lidera o ranking com CPC = 16,82 porque combina inferência muito rápida, throughput de 10,3 tok/s e baixo consumo de RAM (1,2 GB, sem throttling). Essa liderança, contudo, deve ser interpretada com cautela: sua acurácia global é de 44,6%, abaixo do limiar de 50% adotado neste estudo. Portanto, o CPC deve ser lido junto com um piso mínimo de acurácia. Se a métrica atribuísse peso ainda maior à acurácia, o ranking tenderia a favorecer modelos como Qwen3:1.7B, Gemma4:e2B ou os modelos de 4B, apesar do custo computacional mais elevado. O Qwen3:1.7B (CPC = 9,22) tem acurácia mais alta (60,9%), throughput de 5,2 tok/s e consumo de apenas 2,0 GB de RAM.

Os modelos com maior benefício bruto (Qwen3:4B, Qwen3.5:4B, Gemma4:e4B) são penalizados pelo alto custo computacional. Modelos de custo mínimo como, por

exemplo, Granite4:350M têm benefício baixo por causa da acurácia reduzida. Pelo CPC, os modelos intermediários ficam na faixa mais equilibrada.

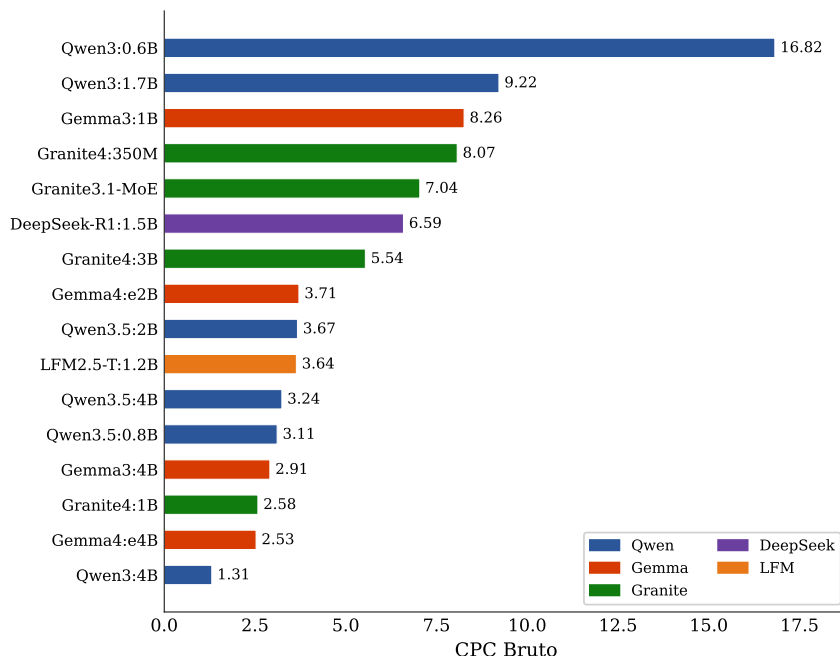


Figura 3. CPC bruto por modelo. Cores indicam famílias arquiteturais.

5. Discussão

Os resultados mostram que o número de parâmetros, sozinho, não determina a qualidade de um modelo como componente de sistemas de tutoria. Há uma correlação geral entre escala e acurácia: modelos de 4B alcançam 83–88%, modelos de 2B ficam entre 59–75%, e modelos abaixo de 1B raramente superam 47%. Ainda assim, a arquitetura pesa bastante. O Gemma4:e2B, com 2 bilhões de parâmetros, supera o Gemma3:4B em mais de 7 pontos percentuais (74,4% contra 66,7%), indicando que avanços geracionais e escolhas arquiteturais podem compensar diferenças de escala.

De forma análoga, o Granite3.1-MoE (3B) atinge 9,4 tok/s, throughput comparável ao de modelos com um terço do seu tamanho, ainda que sua acurácia de 46,1% não deva ser interpretada isoladamente como adequada para uso pedagógico. Isso mostra que a seleção de modelos para educação em hardware limitado não pode se guiar apenas pela contagem de parâmetros.

O CPC explicita a tensão entre qualidade e custo operacional. A métrica indica que a melhor opção prática não está nos extremos: os modelos mais acurados são caros demais para uso interativo, enquanto os mais leves erram demais. O Qwen3:1.7B exemplifica esse equilíbrio: com 60,9% de acurácia, 5,2 tok/s e apenas 2,0 GB de RAM, entrega a segunda melhor relação custo-benefício do benchmark (CPC = 9,22), atrás apenas do Qwen3:0.6B (CPC = 16,82), cuja acurácia de 44,6% fica abaixo do limiar de certificação que utilizamos. O Gemma4:e2B tem acurácia superior (74,5%), mas seu consumo de 7,6 GB de RAM o exclui de dispositivos com menos de 8 GB. Modelos intermediários, portanto, são a faixa com maior potencial de adoção prática.

O estresse térmico também importa. Modelos que operam por longos períodos em temperaturas elevadas aceleram a degradação dos componentes e aumentam o risco de falhas prematuras, um problema grave em escolas onde a reposição de hardware é lenta ou inviável. No nosso benchmark, os três modelos mais acurados (Qwen3:4B, Qwen3.5:4B e Gemma4:e4B) operaram com temperaturas médias entre 77,6 e 82,3°C, acionando throttling térmico que reduziu a frequência da CPU em até 6,9%. Mesmo assim, permaneceram próximos do limite de 85°C da plataforma. Em contrapartida, o Qwen3:1.7B manteve temperatura média de 77,5°C com throttling desprezível (2386 MHz contra 2400 MHz nominais), indicando operação termicamente sustentável. Privilegiar só a acurácia pode reduzir a vida útil do equipamento, um custo oculto que o CPC incorpora parcialmente pelo termo de throttling.

Mesmo os modelos acima do limiar de certificação do ENCCEJA ainda erram bastante. O Qwen3:1.7B, apesar do bom equilíbrio geral, erra aproximadamente 4 em cada 10 questões; o Gemma4:e2B erra cerca de 1 em cada 4. Em um cenário real de tutoria, respostas incorretas podem consolidar conceitos equivocados, sobretudo em Matemática. A acurácia, ainda que aceitável para um benchmark, não basta para garantir segurança pedagógica. Estratégias de calibração de confiança podem ajudar estudantes e professores a tratar respostas incertas com cautela. Em implantação escolar, recomenda-se que respostas dos modelos sejam apresentadas como apoio ao estudo, não como fonte final de validação. Professores devem poder revisar respostas, selecionar atividades adequadas, definir quando o sistema pode ser usado autonomamente e orientar estudantes diante de respostas de baixa confiança. Também é recomendável registrar interações problemáticas para auditoria pedagógica e melhoria posterior do sistema. A supervisão humana periódica continua sendo a camada mais robusta de validação, visto que esse tipo de sistema deve ser tratado como ferramenta de apoio e nunca como substituto de um professor.

6. Limitações

Este trabalho tem três limitações principais. Primeiro, o dataset se restringe a um único exame (ENCCEJA), embora expandido para três anos. Questões de nível fundamental podem não refletir a complexidade de tarefas educacionais mais avançadas; a generalização para outros contextos requer validação adicional.

Segundo, a avaliação foi puramente objetiva, comparando respostas com o gabarito oficial. Essa abordagem não avalia a qualidade explicativa: um modelo pode acertar por razões erradas ou fornecer explicações pedagogicamente inadequadas. A ausência de avaliação humana limita a compreensão da eficácia pedagógica em sistemas de tutoria.

Terceiro, a interface Ollama padroniza a execução e a coleta de métricas, mas restringe a seleção de modelos aos disponíveis na plataforma. Como consequência, modelos potencialmente relevantes, como aqueles treinados especificamente em português mas indisponíveis no ecossistema Ollama, não foram incluídos.

7. Conclusão

Este trabalho avaliou o potencial de 16 LLMs compactos em uma Raspberry Pi 5 como auxiliares em sistemas de tutoria educacional offline para o Ensino Fundamental. Os resultados confirmam a viabilidade técnica da abordagem, mas mostram que o gargalo

determinante não é a memória, e sim a velocidade de inferência: a RAM de 16 GB comportaria modelos maiores, mas a ausência de acelerador dedicado torna proibitivo o tempo de resposta de qualquer modelo acima de 4 bilhões de parâmetros para uso interativo. Os modelos mais acurados atingem até 88% de acertos, mas com TTFT entre 30 e 70 segundos, latência que descaracteriza a tutoria em tempo real.

Sem estratégias de adaptação como fine-tuning ou Retrieval-Augmented Generation (RAG), os modelos atuais ainda não oferecem ensino com a responsividade e a segurança pedagógica que o contexto exige. Em hardware limitado, a decisão de deployment envolve uma troca direta: privilegiar acurácia sacrifica fluidez, mas priorizar velocidade reduz a qualidade. Modelos intermediários, como o Qwen3:1.7B, aparecem como a opção mais equilibrada, com acurácia acima do limiar de certificação, latência menor e consumo térmico aceitável.

Apesar das limitações, LLMs compactos podem apoiar sistemas de tutoria educacional em regiões com infraestrutura precária. Trabalhos futuros devem avaliar subjetivamente as respostas geradas, considerando didática, coesão e adequação da linguagem ao público infantil. Também é necessário testar motores de inferência alternativos ao Ollama, tanto para ampliar o catálogo de modelos quanto para reduzir o overhead de orquestração. Por fim, modelos promissores podem ser adaptados com fine-tuning em dados educacionais em português ou RAG, buscando elevar a acurácia e a qualidade pedagógica sem depender apenas do aumento de parâmetros.

Agradecimentos

Os autores agradecem o apoio financeiro do Instituto Nacional de Ciência e Tecnologia em Inteligência Artificial na Educação Desplugada (INCT IA.Edu), financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), processo nº 408483/2024-5, e pela Fundação de Amparo à Pesquisa do Estado de Alagoas (FAP-PEAL), processo nº 60030.0000001481/2025. Agradecem também ao Instituto Kunumi pelo apoio financeiro concedido e por seu compromisso com o avanço da pesquisa científica.

8. Declaração sobre uso de Inteligência Artificial

Foram utilizados modelos de Inteligência Artificial para apoio à revisão linguística, correção gramatical, reorganização textual e auxílio na estruturação de trechos do artigo. As decisões metodológicas, análises, resultados, interpretação dos dados e versão final do texto permaneceram sob responsabilidade integral dos autores.

Referências

- Araujo, Jr., J. R. S., Marinho, L. B., Campos, L. S., Lima, K. N. S., Pereira, D. E., Cavalcanti, H. B., Ramos, A. L. C., and Araújo, E. C. (2025). Smarter questions, smaller models: RAG-enhanced multiple-choice question generation for POSCOMP. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.
- Barros, A., Monteiro, M., Rodrigues, L., Dermeval, D., Isotani, S., and Bittencourt, I. I. (2025). Evaluating large language model quality in resource-constrained environments: An educational stakeholders' survey on accuracy, completeness, and readability in Brazil. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.

- Camelo, V. C. S. and Siebra, C. A. (2025). Estimulando a metacognição de estudantes por meio da tutoria virtual com IA generativa. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.
- Liu, M. X., Liu, F., Fiannaca, A. J., Koo, T., Dixon, L., Terry, M., and Cai, C. J. (2024). “we need structured output”: Towards user-centered constraints on large language model output. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA '24)*. ACM.
- Luz de Araujo, P. H. and Roth, B. (2025). Helpful assistant or fruitful facilitator? investigating how personas affect language model behavior. *PLOS ONE*, 20(6):e0325664.
- Monteiro, M., Barros, A., Rodrigues, L., Dermeval, D., Bittencourt, I. I., Isotani, S., and Macario, V. (2025). Small Device, Big Decision: Comparing lightweight LLMs’ computational performance and output quality for AIED unplugged. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.
- Palahan, S. (2025). Pythonpal: Enhancing online programming education through chatbot-driven personalized feedback. *IEEE Transactions on Learning Technologies*.
- Pereira, A. J., Gomes, A. S., and Primo, T. T. (2024). Avaliação de sistema de tutoria inteligente para cooperar com tutores humanos. *Gestão.Org – Revista Eletrônica de Gestão Organizacional*, 22:1–17.
- Pereira, A. J., Gomes, A. S., and Primo, T. T. (2025). Evaluation of human-artificial hybrid tutoring system for mediation of engagement in e-learning. *Revista Brasileira de Informática na Educação (RBIE)*, 33:216–246.
- Pimentel, E. P., Silva, L. D. L., Takeuti, R. H., and Braga, J. C. (2025). Abordagem com LLM para geração automatizada de feedback no ensino de programação de computadores. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.
- Pinho, P. C. R., Silva, R. Z., Primo, T. T., Dermeval, D., Bittencourt, I. I., and Isotani, S. (2025). Win-win situation: Generative AI in educational constrained environments. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.
- Ramires, G. and Primo, T. T. (2025). Uma interface de voz para inferência de IA offline: Avaliação da viabilidade de modelos modernos de IA para computação de borda. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.
- Rodrigues, L. et al. (2025). A long-term AIED unplugged framework. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.
- Santos, M. M., Barros, A. P., Santos, E., Silva, J. G., Isotani, S., Bittencourt, I. I., Macario, V., Rodrigues, L., and Dermeval, D. (2025). Geração de questões com LLMs leves: Um estudo inicial sobre a percepção de educadores. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.
- Santos, M. M., Dermeval, D., Barros, A., Primo, T., Rodrigues, L., Bittencourt, I. I., and Isotani, S. (2024). Near Feasibility, Distant Practicality: Empirical analysis of deploying and using LLMs on resource-constrained smartphones. In *Proceedings of the 13th International Conference on Information and Communication Technologies and Development (ICTD)*. ACM.

- Silva, J. V. and Andrade, T. L. (2025). A efetividade de sistemas tutores inteligentes no ensino: um estudo baseado no mapeamento sistemático da literatura. In *Anais do XXXVI Simpósio Brasileiro de Informática na Educação (SBIE)*. SBC.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., and Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems (NeurIPS)*, 35.
- Yeo, G., Tan An Ting, F., Jaidka, K., Furniturewala, S., Fanyou, W., Xu, W., Jain, V., Chadha, A., Liu, Y., and Ng, S. K. (2025). PHAnToM: Persona-based prompting has an effect on theory-of-mind reasoning in large language models. *Proceedings of the International AAAI Conference on Web and Social Media*, 19(1):2124–2142.