On-Premises vs. APIs de Nuvem para Modelos de Linguagem de Grande Porte (LLMs) em Sistemas Agênticos: Uma Análise Comparativa de Desempenho, Requisitos de Hardware e Viabilidade Econômica em 2026

Joao Pedro Sansao

https://doi.org/10.1590/SciELOPreprints.16747

Submetido em: 2026-06-30
Postado em: 2026-07-02 (versão 1)
(AAAA-MM-DD)

A moderação deste preprint recebeu o(s) endosso(s) de:

- Michel Leles (ORCID: https://orcid.org/0000-0001-7399-7444)

On-Premises vs. APIs de Nuvem para Modelos de Linguagem de Grande Porte (LLMs) em Sistemas Agênticos: Uma Análise Comparativa de Desempenho, Requisitos de Hardware e Viabilidade Econômica em 2026

On-Premises vs. Cloud APIs for Large Language Models (LLMs) in Agentic Systems: A Comparative Analysis of Performance, Hardware Requirements, and Economic Viability in 2026

João Pedro Hallack Sansão

https://orcid.org/0000-0003-0095-2629*

30 de Junho de 2026

Resumo

Este artigo apresenta uma análise comparativa sobre a viabilidade técnica, operacional e financeira da implantação de Modelos de Linguagem de Grande Porte (LLMs) localmente (*on-premises*) em comparação com o uso de APIs de nuvem (comerciais e de código aberto agregadas). O estudo foca na aplicação desses modelos em sistemas agênticos, caracterizados por loops de execução contínuos e alta densidade de chamadas. Detalhamos os requisitos de hardware e VRAM necessários para executar modelos representativos das famílias Llama, Qwen e Gemma nas faixas de 8B, 32B, 70B e 405B de parâmetros. Apresentamos dois estudos de caso

*E-mail: joao@ufsj.edu.br. Filiação: Departamento de Tecnologia em Engenharia Civil, Computação, Automação, Telemática e Humanidades, Campus Alto Paraopeba, Universidade Federal de São João del-Rei, Ouro Branco, MG, Brasil.

quantitativos detalhados (estação de trabalho com 2x RTX 4090 e servidor HGX com 8x H100 SXM5) para deduzir o custo real por milhão de tokens (MTok) sob diferentes níveis de utilização (10%, 50% e 100%). Por fim, deduzimos as equações de *break-even* econômico, revelando insights financeiros sobre o mercado de nuvem e custos de eletricidade locais no cenário atual de 2026.
Palavras-chave: LLM. On-Premises. APIs de Nuvem. Sistemas Agênticos. Custo Total de Propriedade (TCO). Break-Even.

Abstract

This article presents a comparative analysis of the technical, operational, and financial feasibility of deploying Large Language Models (LLMs) on-premises compared to using cloud APIs (both commercial and aggregated open-source). The study focuses on the application of these models in agentic systems, characterized by continuous execution loops and high query density. We detail the hardware and VRAM requirements needed to run representative models from the Llama, Qwen, and Gemma families across the 8B, 32B, 70B, and 405B parameter ranges. We present two detailed quantitative case studies (a workstation with 2x RTX 4090 and an HGX server with 8x H100 SXM5) to derive the actual cost per million tokens (MTok) under different utilization levels (10%, 50%, and 100%). Finally, we formulate the economic break-even equations, revealing financial insights into the cloud market and local electricity costs in the current 2026 landscape.
Keywords: LLM. On-Premises. Cloud APIs. Agentic Systems. Total Cost of Ownership (TCO). Break-Even.

1 Introdução

O emprego de sistemas agênticos baseados em inteligência artificial tem se expandido para a automação de tarefas complexas que demandam múltiplos ciclos de execução. Diferente de sistemas de conversação lineares, os agentes autônomos operam de forma recursiva por meio de metodologias como ReAct [1], *Chain of Thought* (CoT) [2] e sistemas multiagentes coordenados [3]. Esse padrão de arquitetura resulta em uma alta densidade de consultas a modelos de linguagem, em que uma única requisição de usuário pode demandar dezenas de chamadas sequenciais.

Neste cenário, a seleção da infraestrutura tecnológica apropriada, seja por meio da aquisição de hardware dedicado para implantação local (*on-premises*), seja pelo consumo de capacidade de processamento via interfaces de programação de aplicação (APIs) de nuvem, constitui uma decisão estratégica. Essa escolha envolve o balanceamento de fatores como latência de rede, soberania de dados, flexibilidade de customização e viabilidade econômica.

Este trabalho apresenta uma análise comparativa sobre a viabilidade técnica e financeira de ambas as alternativas no cenário tecnológico atual. Inicialmente, são apresentados os conceitos e definições no referencial teórico. Em seguida, discutem-se os fatores de escolha de infraestrutura, os requisitos de hardware e a memória de acesso aleatório de vídeo (VRAM) para os modelos abertos Llama [4], Qwen [5] e Gemma [6]. Por fim, sistematiza-se a relação entre custo e desempenho dos modelos, com base em estudos de caso e modelagem de ponto de equilíbrio (*break-even*).

2 Referencial Teórico

Para a adequada compreensão da análise de infraestrutura de inteligência artificial descrita neste artigo, faz-se necessária a definição de quatro conceitos fundamentais: modelos de linguagem de grande porte, agentes autônomos, tokens e sistemas agênticos.

2.1 Modelos de Linguagem de Grande Porte (LLMs)

Modelos de Linguagem de Grande Porte (*Large Language Models* — LLMs) constituem arquiteturas de redes neurais profundas baseadas predominantemente no mecanismo de atenção da arquitetura Transformer. Esses modelos são pré-treinados de forma auto-supervisionada em corpora de texto massivos, com o objetivo primário de prever a probabilidade da ocorrência de um token subsequente dado um contexto linguístico prévio. A escalabilidade dessas arquiteturas, caracterizada por dezenas ou centenas de bilhões de parâmetros, viabiliza o surgimento de capacidades cognitivas avançadas, como raciocínio lógico, compreensão de linguagem natural e geração de código computacional, servindo como o motor de inferência central para arquiteturas agênticas.

2.2 Agentes Autônomos

De acordo com o levantamento de Wang et al. [7], agentes autônomos baseados em Modelos de Linguagem de Grande Porte (LLMs) constituem entidades de software capazes de perceber o ambiente que as cerca, tomar decisões baseadas em objetivos predefinidos e executar ações de forma independente, sem a necessidade de intervenção humana contínua. Esses agentes utilizam a capacidade cognitiva do modelo de linguagem para planejar ações, interagir com ferramentas externas e adaptar seu comportamento a partir de observações do ambiente.

2.3 Tokens

No processamento de linguagem natural, tokens correspondem às unidades elementares de processamento de texto geradas a partir de algoritmos de segmentação (tokenização), como o *Byte-Pair Encoding* (BPE). Os tokens podem representar palavras completas, subpalavras ou caracteres individuais. Em modelos autorregressivos, a inferência consiste na previsão probabilística sequencial do próximo token a partir de uma sequência prévia, sendo o token a métrica básica de tarifação e de consumo de memória no contexto de inferência.

2.4 Sistemas Agênticos

Sistemas agênticos referem-se a arquiteturas de software que integram um ou mais agentes autônomos para solucionar problemas complexos. Em vez de operar sob uma lógica linear de entrada e saída, os sistemas agênticos organizam loops de execução contínuos que alternam etapas de raciocínio, consulta a bases de conhecimento externas, execução de código e chamada de ferramentas de terceiros por meio de metodologias estruturadas como o ReAct [1], o *Chain-of-Thought* (CoT) [2] ou arquiteturas de conversação multiagentes coordenadas [3].

3 Recursos Computacionais para LLMs em Sistemas Agênticos: On-Premises vs. Cloud APIs

A divisão entre infraestrutura própria e nuvem gerenciada não reside apenas no aspecto financeiro. O comportamento dinâmico dos sistemas agênticos gera requisitos técnicos específicos que influenciam diretamente a escolha da arquitetura.

3.1 Quando a Infraestrutura Dedicada (On-Premises / Private Cloud) é Necessária

A primeira justificativa para a adoção de infraestrutura local é a latência de loops sequenciais. Em sistemas agênticos, o *prompt* do passo N depende da resposta obtida no passo $N-1$, estabelecendo uma cadeia de dependências em que o tempo de processamento é crítico. Uma latência adicional de 150 a 300ms por chamada a uma API externa de nuvem acumula atrasos de vários segundos em sequências de execução longas, comprometendo a eficiência operacional. A implantação local, utilizando conexões internas de rede de baixa latência, reduz esses gargalos de comunicação externa.

A soberania, a segurança e a conformidade regulatória constituem outro fator determinante. Organizações que processam dados de alta confidencialidade, a exemplo de informações médicas reguladas pelo regime de dados pessoais sensíveis da Lei Geral de Proteção de Dados (LGPD) sob fiscalização da Autoridade Nacional de Proteção de Dados (ANPD) no Brasil, ou pelo *Health Insurance Portability and Accountability Act* (HIPAA) nos Estados Unidos, enfrentam restrições quanto ao envio de dados para servidores de terceiros. O processamento local permite o controle físico e digital direto sobre a custódia das informações.

Adicionalmente, as APIs de nuvem comercial frequentemente impõem limites de requisições (*rate limits*), medidos em requisições por minuto (RPM) ou tokens por minuto (TPM), para evitar a sobrecarga de seus clusters. Sistemas multiagentes operando concorrentemente podem exceder esses limites sob cargas elevadas de trabalho, gerando interrupções no fluxo de execução. O uso de hardware próprio elimina tais restrições artificiais, limitando-se apenas à capacidade computacional do equipamento disponível.

Há também o aspecto do custo de contexto repetitivo. Em fluxos de trabalho agênticos, o histórico da conversação e as definições das ferramentas integradas são transmitidos repetidamente a cada iteração. Nas chamadas de API de nuvem, o custo é cobrado por token em cada chamada efetuada. Na infraestrutura local, o custo marginal do uso contínuo de contextos amplos restringe-se ao consumo elétrico do hardware de inferência, o que representa maior previsibilidade de despesas sob alta demanda de tokens.

A customização e o ajuste fino (*fine-tuning*) também justificam a escolha pelo modelo local. Cenários que demandam técnicas de adaptação nos pesos dos modelos (como LoRA/QLoRA ou treinamento contínuo) ou tokenizadores customizados para domínios específicos (como o vocabulário jurídico ou médico) exigem acesso direto aos parâmetros do modelo de linguagem. Por fim, cenários com necessidade de operabilidade offline, como em ambientes industriais isolados ou locais sem conectividade estável, demandam infraestrutura local para garantir a continuidade operacional.

3.2 Quando o Uso de APIs e Nuvem Gerenciada é Recomendado

Por outro lado, o uso de APIs gerenciadas em nuvem apresenta vantagens relevantes em cenários distintos. Para atividades de prototipagem e desenvolvimento de produtos mínimos viáveis (MVPs), o acesso a modelos prontos elimina a necessidade de aportes de capital iniciais em hardware. Isso permite a realização de testes de conceitos de forma ágil e com menor barreira de entrada.

Além disso, as APIs constituem o canal de acesso a modelos fechados proprietários com elevado desempenho em tarefas de raciocínio complexo, cujas arquiteturas não são disponibilizadas em versões de pesos abertos. Para tarefas de alta complexidade em que a capacidade cognitiva do modelo é o fator crítico para o sucesso da aplicação, a utilização dessas interfaces externas é necessária.

A flexibilidade frente a flutuações e sazonalidade de demanda é outra característica favorável das APIs, que operam sob um regime de custos variáveis (*OpEx*) baseado no consumo efetivo. Em contrapartida, uma infraestrutura física dedicada geraria custos contínuos de depreciação mesmo em períodos de inatividade. Por fim, o consumo de serviços gerenciados reduz a complexidade operacional, transferindo para os provedores de nuvem a responsabilidade por tarefas como manutenção preventiva, refrigeração de data centers, gestão de falhas de hardware e otimização de baixo nível de kernels de processamento gráfico.

4 Requisitos de Hardware e Dimensionamento de LLMs de Pesos Abertos

Para o planejamento de uma infraestrutura local dedicada (on-premises), o principal limitador físico de desempenho é a quantidade de VRAM (Video RAM) disponível nas unidades de processamento gráfico (GPUs). As famílias de modelos de pesos abertos de maior relevância no cenário atual de 2026 compreendem o Llama 3/3.1/3.3 (Meta) [4], o Qwen 2.5/2.5-Coder (Alibaba) [5] e o Gemma 2 (Google) [6].

4.1 A Fórmula de Cálculo de VRAM

A estimativa da quantidade de VRAM necessária para carregar e executar um modelo em regime de inferência pode ser modelada de forma analítica com base na arquitetura de transformadores [8]. Trata-se de uma regra prática (*rule of thumb*) amplamente adotada pela comunidade [9], ajustada para contemplar a reserva destinada ao cache de chaves e valores (*Key-Value Cache* ou KV Cache), além dos buffers operacionais do sistema e do ambiente CUDA:

$$VRAM_{\text{req}}(GB) \approx (P \times B) \times 1.2 \tag{1}$$

Nessa formulação, P representa a quantidade total de parâmetros do modelo (expressa em bilhões) e B o tamanho do dado em bytes de acordo com o formato de representação numérica e a precisão adotados. Em conformidade com a prática atual de inferência, assume-se $B = 2,0$ para meia precisão (FP16/BF16); $B = 1,0$ para precisões reduzidas de 8 bits (FP8/INT8); e $B = 0,5$ para quantizações de 4 bits (INT4/Q4). O fator multiplicativo de $1,2$ denota uma margem de contingência de 20% para acomodar o KV Cache em sequências de contexto típicas, os buffers de ativação do modelo e o consumo de memória do framework de execução, tais como vLLM [10] ou TensorRT-LLM [11].

4.2 Tabela de Requisitos de Hardware por Tamanho de Modelo (2026)

A Tabela 1 apresenta os requisitos mínimos de VRAM e as respectivas recomendações de hardware para diferentes escalas de modelos de pesos abertos.

Para a composição desta tabela, foram considerados os aceleradores mais comuns em ambientes de computação pessoal e corporativa. As recomendações foram divididas entre hardware de consumo (GPUs de arquitetura de desktop e estações de trabalho de memória unificada) e hardware de servidor (GPUs de nível empresarial de alta largura de banda).

A estimativa de custos apresentada reflete os valores médios praticados no mercado internacional em 2026. Para o hardware de consumo, as estimativas são baseadas nos preços recomendados pelo fabricante (MSRP) e preços de varejo globais. Para o hardware corporativo, como as GPUs NVIDIA A100 e H100, os valores correspondem ao custo estimado de aquisição de mercado secundário ou contratos de fornecimento corporativo em larga escala para servidores completos.

Tabela 1 – Requisitos de hardware por tamanho de modelo (2026)

Modelo	Precisão	VRAM Mínima	Hardware (Consumo)	Hardware (Servidor)	Custo Est. (2026)
8B	INT4 (Q4)	∼4.8 GB	1x RTX 4060 Ti (16GB)	1x NVIDIA L4 (24GB)	$450 - $600
	FP16	∼19.2 GB	1x RTX 4090 (24GB)		$2,000
14B	INT4 (Q4)	∼8.4 GB	1x RTX 4060 Ti (16GB)	1x NVIDIA L4 (24GB)	$450 - $800
	FP8	∼16.8 GB	1x RTX 4090 (24GB)	1x NVIDIA L4 (24GB)	$2,000
32B	INT4 (Q4)	∼19.2 GB	1x RTX 4090 (24GB)	1x A100 (40GB/80GB)	$2,000
	FP8	∼38.4 GB	2x RTX 4090 (48GB)	2x NVIDIA L4	$4,000 - $8,000+
70B / 72B	INT4 (Q4)	∼42.0 GB	2x RTX 4090 (48GB)	1x A100 (80GB)	$4,000 - $5,000
	FP8	∼84.0 GB	Mac Studio M3 (96GB)*	2x H100 (160GB)	$5,299 / $70,000
405B	INT4 (Q4)	∼243 GB	8x RTX 3090/4090	4x H100 (320GB VRAM)	$16,000 - $20,000
	FP8	∼486 GB	Mac Studio M5 (pl.)	8x H100 SXM5 (640GB)	$120,000 - $300,000+

*Nota: Arquiteturas de Unified Memory (Apple Silicon) utilizam a RAM do sistema de forma compartilhada como VRAM; devido à grande largura de banda de memória e capacidade, essa arquitetura é comum para a execução de modelos de 70B com custos reduzidos.

5 Sistematização: Parâmetros, Custos, Benchmarks e Eficiência Econômica

A análise comparativa do desempenho relativo e da viabilidade econômica dos modelos de pesos abertos exige o cruzamento de suas métricas de capacidade cognitiva com seus custos operacionais. A Tabela 2 sistematiza essas variáveis, comparando as capacidades de raciocínio, os investimentos em infraestrutura local e os custos de utilização via APIs comerciais no cenário de 2026.

Os índices de desempenho apresentados (representados pelos benchmarks MMLU para conhecimento geral, GPQA para raciocínio lógico e científico de nível de pós-graduação e HumanEval para capacidade de geração de código) foram obtidos diretamente dos relatórios técnicos oficiais publicados pela Meta AI [4] e pela Alibaba Cloud [5]. Os custos associados à infraestrutura física local e as tarifas de entrada (input) e saída (output) das APIs foram levantados junto aos índices do portal Artificial Analysis [12] e dos painéis de preços das respectivas plataformas.

Para avaliar de forma quantitativa o retorno do investimento por token consumido, é definida a Razão de Custo-Eficiência (R_{ce}), expressa pela relação entre o desempenho do modelo no benchmark MMLU e o custo médio ponderado da API ($T_{blended}$):

$$R_{ce} = \frac{MMLU(\%)}{T_{blended}} \tag{2}$$

Onde $T_{blended}$ representa a tarifa mista por milhão de tokens, calculada com base em uma distribuição típica de tráfego de sistemas agênticos, composta por 80% de tokens de entrada e 20% de tokens de saída:

$$T_{blended} = 0.8 \times T_{input} + 0.2 \times T_{output} \tag{3}$$

Essa razão fornece um indicador normalizado que descreve a eficiência cognitiva adquirida por unidade monetária gasta. Os resultados evidenciam que os modelos de menor escala (como o Llama 3.1 8B) apresentam uma razão significativamente mais elevada devido às tarifas de API extremamente reduzidas em 2026, ao passo que modelos de grande porte (como o Llama 3.1 405B) oferecem menor eficiência de custo bruto por token, embora sejam necessários para tarefas de alta complexidade.

Cabe destacar que a inclusão das especificações e custos do hardware local na Tabela 2, muito embora não influencie diretamente o cálculo matemático da razão de custo-eficiência das APIs (R_{ce}), justifica-se por razões metodológicas de tomada de decisão. Primeiramente, essa estrutura permite confrontar de forma direta o investimento inicial necessário para a aquisição da infraestrutura local (CapEx) com o custo operacional recorrente do uso de APIs (OpEx) para cada faixa de tamanho de modelo. Em segundo lugar, evidencia como as barreiras de entrada físicas escalam de forma exponencial à medida que a complexidade do modelo aumenta. Por fim, tais custos locais estabelecem os parâmetros de entrada (custo

fixo de aquisição) que serão empregados nas seções subsequentes para a determinação do ponto de equilíbrio econômico (*break-even*).

Tabela 2 – Benchmarks, Custos de Hardware e Tarifas de API em 2026

Modelo	Parâmetro	MMLU	GPQA	HumanEval	Hardware Local	Custo Hard.	API Ent.	API Saí.	Custo-Eficiência*
Llama 3.1 8B	8B	69.4%	30.4%	72.6%	1x RTX 4060 Ti 16GB	$1,500	$0.02	$0.03	**3.154.5%**
Qwen 2.5 32B	32B	79.0%	41.0%	92.7%	1x RTX 4090 24GB	$3,500	$0.07	$0.14	**940.5%**
Llama 3.3 70B	70B	86.0%	50.5%	89.0%	2x RTX 4090 24GB	$6,000	$0.10	$0.30	**614.3%**
Llama 3.1 405B	405B	88.6%	50.7%	89.0%	Servidor 8x H100 SXM5	$300,000	$3.50	$3.50	**25.3%**

*Nota: A Razão Custo-Eficiência é calculada como MMLU (em %) dividido pelo custo médio misturado da API por milhão de tokens (assumindo uma distribuição típica de 80% tokens de entrada e 20% tokens de saída).

6 Estudo de Caso: Cálculo do Custo de MToks On-Premises

Para mensurar a viabilidade econômica do on-premises frente às APIs, apresentamos dois cenários reais com modelagem de custo total de propriedade (TCO) sob três patamares de utilização mensal, fundamentados na metodologia de análise de custo e eficiência computacional de inferência proposta em trabalhos como FlexGen [13] e FrugalGPT [14]. O TCO (*Total Cost of Ownership*) é um indicador financeiro que compreende a soma de todos os custos diretos e indiretos associados à aquisição, operação e manutenção de uma infraestrutura ao longo de seu ciclo de vida útil. Neste estudo, o TCO mensal é estimado a partir da soma da amortização do capital (depreciação), do suporte administrativo e do consumo de energia elétrica. Ambos os cenários compartilham os mesmos parâmetros financeiros base: depreciação do hardware em 3 anos (36 meses) com amortização linear; custo de eletricidade comercial de $0,15/kWh para o Cenário A; custo industrial ou de data center de $0,10/kWh para o Cenário B; e período mensal de 30 dias, equivalente a 720 horas.

Para o cálculo de consumo elétrico, é considerado o indicador PUE (*Power Usage Effectiveness*). O PUE expressa a razão entre a energia total consumida pela instalação (incluindo sistemas de refrigeração, transformadores e perdas na rede de distribuição) e a energia utilizada estritamente pelos equipamentos de processamento de dados. Dessa forma, um fator PUE de 1,2 indica que 20% da eletricidade total consumida é destinada ao resfriamento e suporte da infraestrutura física.

6.1 Cenário A: Estação de Trabalho Local (Workstation 2x RTX 4090)

O objetivo deste cenário é avaliar o custo de rodar um modelo de 70B parâmetros, especificamente o Llama 3.3 70B em quantização Q4, em uma estação de trabalho de alto desempenho de mercado consumidor.

O investimento inicial (CapEx) é composto por duas GPUs GeForce RTX 4090 de 24GB de VRAM a $2.000 cada, totalizando $4.000 apenas em aceleradores. Somado ao

gabinete, placa-mãe profissional, CPU avançada, 128GB de RAM DDR5, fonte de 1.600W e SSD NVMe de 2TB, estimado em $2.000, o CapEx total chega a $6.000, resultando em uma depreciação mensal de $166,67.

Em termos de consumo energético, o sistema em carga ativa consome 850W. Aplicado um fator PUE de 1,2 para contabilizar as perdas térmicas e o resfriamento local, a carga real sobe para 1,02 kW. Em estado ocioso (*idle*), o consumo cai para 150W, ou 0,18 kW com o mesmo PUE. Isso se traduz em custos horários de $0,153 durante a operação ativa e $0,027 durante a ociosidade. Em termos de *throughput*, utilizando vLLM [10] com paralelismo de tensores (TP=2) e quantização Q4, o sistema atinge uma média de 50 tokens por segundo sob concorrência moderada, equivalentes a 0,18 MToks por hora ativa.

Os custos por milhão de tokens gerados variam conforme o nível de utilização do hardware:

1. Baixa Utilização (10% ativo - 72h carga, 648h idle): Tokens = 12.96 MToks. Custo elétrico = $28.52. TCO Mensal = $195.19. Custo por MTok = $15.06.

2. Média Utilização (50% ativo - 360h carga, 360h idle): Tokens = 64.80 MToks. Custo elétrico = $64.80. TCO Mensal = $231.47. Custo por MTok = $3.57.

3. Alta Utilização (100% ativo - 720h carga constante): Tokens = 129.60 MToks. Custo elétrico = $110.16. TCO Mensal = $276.83. Custo por MTok = $2.14.

6.2 Cenário B: Servidor Corporativo (HGX 8x H100 SXM5)

Este cenário objetiva rodar o Llama 3.1 405B em FP8.

- CapEx (Investimento Inicial): Servidor completo com 8x H100 SXM5 = $300,000. Suporte corporativo e administração (10% ao ano) = $2,500/mês. Depreciação Mensal = $8,333.33. Custo Fixo Total = $10,833.33/mês.

- OpEx (Energia em Data Center):

  - *Consumo em Carga Ativa:* 10 kW. Com PUE de 1.3, consumo = 13.0 kW.
  - *Consumo em Espera (Idle):* 2 kW. Com PUE 1.3, consumo = 2.6 kW.
  - *Custo Energia Ativa (hora):* 13.0 kW × $0.10 = $1.30/hora.
  - *Custo Energia Idle (hora):* 2.6 kW × $0.10 = $0.26/hora.

- Vazão (Throughput): 800 tokens por segundo. Geração por hora ativa = 2.88 MToks.

Os custos por milhão de tokens gerados variam conforme o nível de utilização do hardware:

1. Baixa Utilização (10% ativo - 72h carga, 648h idle): Tokens = 207.36 MToks. Custo elétrico = $262.08. TCO Mensal = $11,095.41. Custo por MTok = $53.51.

2. Média Utilização (50% ativo - 360h carga, 360h idle): Tokens = 1,036.80 MToks. Custo elétrico = $561.60. TCO Mensal = $11,394.93. Custo por MTok = $10.99.

3. Alta Utilização (100% ativo - 720h carga constante): Tokens = 2,073.60 MToks. Custo elétrico = $936.00. TCO Mensal = $11,769.33. Custo por MTok = $5.68.

7 Equações e Análise de Break-Even (Ponto de Equilíbrio)

Para formular o ponto de equilíbrio de custos em termos de volume mensal de tokens gerados (V, expresso em milhões de tokens), definimos as equações de custos de cada opção, conforme os modelos analíticos de otimização de custo de inferência na literatura [14].

7.1 Formulação Matemática

O custo mensal de usar uma API baseada em volume de tokens é diretamente proporcional à demanda:

$$C_{\text{API}}(V) = V \times T_{\text{blended}} \tag{4}$$

onde T_{blended} é o preço médio ponderado por milhão de tokens cobrado pelo provedor, conforme definido na Seção 5.

A estrutura de custos da infraestrutura local é diferente: possui uma componente fixa mensal F, independente do uso, somada a um custo variável incremental proporcional ao volume gerado:

$$C_{\text{On-Prem}}(V) = F + P_{\text{incremental}} \times V \tag{5}$$

O custo fixo F engloba a depreciação do hardware, os custos de manutenção e a eletricidade consumida mesmo em estado ocioso durante todas as horas do mês:

$$F = D_{\text{mensal}} + \text{Overhead}_{\text{maint}} + (720 \times C_{\text{idle}}) \tag{6}$$

O custo elétrico incremental por milhão de tokens gerados representa o acréscimo de consumo energético ao sair do estado ocioso para a operação ativa:

$$P_{\text{incremental}} = \frac{C_{\text{active}} - C_{\text{idle}}}{\text{MToks por hora ativa}} \tag{7}$$

O ponto de equilíbrio econômico ocorre quando as duas curvas de custo se cruzam, ou seja, quando $C_{\text{API}}(V) = C_{\text{On-Prem}}(V)$. Resolvendo para o volume V:

$$V_{\text{break-even}} = \frac{F}{T_{\text{blended}} - P_{\text{incremental}}} \tag{8}$$

Uma observação matemática de grande relevância prática: se $T_{\text{blended}} \leq P_{\text{incremental}}$, o denominador torna-se nulo ou negativo, indicando que o custo marginal da eletricidade do hardware local já supera o preço da API. Nesse caso, um break-even puramente financeiro é inviável do ponto de vista matemático, independentemente do volume gerado.

7.2 Aplicação Prática do Ponto de Equilíbrio

7.2.1 Cenário A: Estação de Trabalho (Workstation 2x RTX 4090)

Para o Cenário A, os parâmetros derivados são: custo fixo $F = \$166{,}67 + (720 \times \$0{,}027) = \$186{,}11$ e custo incremental $P_{\text{incremental}} = (\$0{,}153 - \$0{,}027)/0{,}18 = \$0{,}70/\text{MTok}$. A curva de custo local é, portanto, $C_{\text{On-Prem}}(V) = 186{,}11 + 0{,}70 \times V$.

Confrontando essa curva com diferentes APIs de mercado em 2026, encontram-se resultados qualitativamente distintos. Contra o Claude Sonnet 4.6 (Anthropic), com preço blended de \$5,40/MTok, o break-even é atingido em:

$$V = \frac{186{,}11}{5{,}40 - 0{,}70} \approx 39{,}6 \text{ MToks/mês} \tag{9}$$

Isso corresponde a pouco mais de 7 horas ativas por dia: um volume realista para uma equipe com uso intenso, tornando a estação local economicamente vantajosa nesse regime. Contra o GPT-4o (OpenAI), com blended de \$4,00/MTok, o break-even sobe para:

$$V = \frac{186{,}11}{4{,}00 - 0{,}70} \approx 56{,}4 \text{ MToks/mês} \tag{10}$$

ainda factível em ambientes de alta demanda (10,4 horas ativas por dia).

O cenário muda radicalmente ao comparar com APIs de código aberto comoditizadas. O Llama 3.3 70B hospedado no Groq [15], a \$0,63/MTok blended, já torna o denominador negativo ($0{,}63 - 0{,}70 = -0{,}07$): o custo marginal da eletricidade local por token é maior do que o preço integral da API de nuvem. A situação é ainda mais desfavorável contra o OpenRouter/DeepInfra [16], que oferece o mesmo modelo a apenas \$0,14/MTok blended, o que resulta em um denominador de $-0{,}56$. As tarifas reduzidas oferecidas por essas plataformas são baixas a ponto de a tarifa elétrica local torna a máquina própria financeiramente inviável para fins de economia de tokens. A única justificativa para rodar a máquina local, nesses casos, reside em requisitos de segurança, privacidade ou latência interna.

7.2.2 Cenário B: Servidor 8x H100 (Modelo Llama 3.1 405B)

Para o Cenário B, os parâmetros são: $F = \$11.020{,}53$ e $P_{\text{incremental}} \approx \$0{,}361/\text{MTok}$. Confrontando com a API do Llama 3.1 405B na Together AI [17] (\$3,50/MTok flat), o cálculo indica que seria necessário gerar:

$$V = \frac{11.020{,}53}{3{,}50 - 0{,}361} \approx 3.510{,}8 \text{ MToks/mês} \tag{11}$$

Contudo, a capacidade máxima teórica de um servidor de 8×H100 em carga plena é de apenas 2.073,6 MToks/mês. Como o teto de produção física é inferior ao volume de break-even, um único servidor de 8×H100 é incapaz de atingir o ponto de equilíbrio econômico frente à API gerenciada, independentemente do nível de utilização. A depreciação substancial de $300 mil em 36 meses é o principal responsável por essa inviabilidade.

A análise contra o Claude Sonnet 4.6 ($5,40/MTok blended) aponta um break-even de $V \approx 2.187$ MToks/mês, ligeiramente abaixo do teto de produção de 2.073,6 MToks. Essa proximidade sugere que, a partir de múltiplos nós corporativos (nos quais o custo administrativo e operacional é diluído sobre uma frota maior) ou em organizações com dezenas de agentes rodando concorrentemente sobre documentos complexos, o break-even contra APIs fechadas de ponta começa a se justificar do ponto de vista financeiro.

8 Discussão: Arquitetura Híbrida e Roteamento de Modelos em 2026

A constatação econômica mais importante do cenário de 2026 reside na elevada eficiência das APIs de modelos abertos comoditizados. A concorrência entre provedores equipados com hardware customizado (TPUs e LPUs da Groq [15]) e centros de processamento de dados integrados a matrizes energéticas sustentáveis de baixo custo reduziu significativamente os preços por token. A análise financeira demonstra que o custo operacional correspondente à eletricidade necessária para processar cargas de trabalho localmente em uma GPU de consumo RTX 4090, em regiões com tarifas elétricas de padrão comercial, supera o valor de aquisição de tokens nas APIs dos mesmos modelos sob a forma de serviços gerenciados. Esse processo de deflação de preços foi acentuado pela entrada de provedores globais altamente competitivos, como a DeepSeek [18], que viabilizaram a oferta de tokens a preços historicamente baixos por meio de infraestruturas compartilhadas e otimizadas em larga escala.

Nesse contexto, consolida-se o emprego de arquiteturas híbridas estruturadas em torno de um gateway de inteligência artificial habilitado com capacidades de roteamento de modelos (*model routing*) [14]. O roteamento de modelos consiste em uma técnica de gestão dinâmica de requisições, na qual cada prompt de entrada é avaliado por um componente intermediário (classificador de complexidade ou mecanismo de regras) e direcionado ao modelo mais adequado. Essa abordagem visa balancear custo, latência e capacidade de processamento de acordo com as especificidades da tarefa. As estratégias de otimização operacional incluem:

1. Roteamento por Complexidade (Regra 85/15): Esta abordagem consiste no direcionamento automático de aproximadamente 85% das tarefas rotineiras de agentes — como formatação de dados, extração de entidades e respostas a consultas simples —

para modelos locais de menor escala (como o Llama 3.1 8B, caracterizado por baixo consumo energético) ou para APIs comoditizadas de baixo custo [16]. Reservam-se as tarefas de alta complexidade cognitiva, representadas pelos 15% restantes, para modelos de fronteira acessados via APIs pagas (como o Claude 3.5 Sonnet).

2. Compartilhamento e Cache de Contexto (*Context Caching*) [19]: Esta técnica viabiliza o agrupamento de requisições que compartilham contextos idênticos (como manuais de instrução de agentes ou bases de conhecimento estáticas) no gateway, permitindo reduções tarifárias de até 90% em provedores de nuvem que suportam cache de tokens, ou otimizando a latência por meio do gerenciamento local do cache de chaves e valores (KV Cache) na memória do sistema.

9 Especificidades de Implementação em Setores Críticos (Público, Judiciário e Bancário)

A análise de viabilidade para a implantação de Modelos de Linguagem de Grande Porte não se restringe às variáveis financeiras de CapEx e OpEx. Em setores altamente regulados, as restrições jurídicas e de governança frequentemente se sobrepõem às métricas de ponto de equilíbrio econômico, tornando a infraestrutura local não apenas vantajosa, mas obrigatória. Esta seção discute os fatores regulatórios e estratégicos que moldam a escolha arquitetural no setor público, no poder judiciário e no setor bancário.

9.1 Setor Público e Soberania Digital Nacional

No setor público, a questão da soberania digital transcende qualquer cálculo de custo por token. A dependência de provedores de computação em nuvem estrangeiros expõe dados nacionais às leis de jurisdições externas, como ocorre com as implicações extraterritoriais do *CLOUD Act* norte-americano [20], que autoriza autoridades dos Estados Unidos a solicitar dados sob o controle de empresas sediadas em seu território, independentemente da localização física dos servidores. Esse risco jurídico é incompatível com o tratamento de informações classificadas ou estratégicas do Estado.

No contexto brasileiro, a Lei Geral de Proteção de Dados (LGPD) [21] impõe diretrizes estritas sobre o tratamento de dados pessoais pela administração pública. Normas técnicas do Gabinete de Segurança Institucional (GSI/PR) e da Secretaria de Governo Digital exigem que informações sensíveis ou classificadas sejam processadas em *data centers* localizados em território nacional. Embora a administração pública priorize a computação em nuvem para serviços gerais de menor criticidade, em que os ganhos de escala e agilidade são expressivos, a adoção de LLMs para dados estratégicos governamentais aponta invariavelmente para implantações estritamente locais, em infraestruturas controladas por empresas públicas de TI como o SERPRO e a DATAPREV, ou para parcerias com nuvens

soberanas nacionais que garantam soberania operacional, jurídica e tecnológica absoluta sobre a custódia das informações do Estado.

9.2 Poder Judiciário, Sigilo Processual e Explicabilidade Algorítmica

O Poder Judiciário opera com fluxos documentais que concentram um volume extraordinário de dados sensíveis: segredo de justiça, inquéritos policiais sigilosos, dados de menores, registros financeiros e relatórios de inteligência de segurança pública. O simples tráfego dessas informações por APIs de nuvens corporativas comerciais de terceiros representa um risco grave de vazamento e quebra de sigilo processual, com potencial de comprometer a integridade de investigações e ferir direitos fundamentais.

Do ponto de vista regulatório, o Conselho Nacional de Justiça (CNJ), por meio da Resolução nº 332/2020 [22], exige que sistemas baseados em inteligência artificial no Judiciário observem critérios rígidos de explicabilidade algorítmica e segurança. A explicabilidade importa porque magistrados e partes do processo devem ter condições de auditar o funcionamento dos modelos para contestar eventuais viéses em decisões assistidas por IA. A segurança, por sua vez, demanda que o Judiciário detenha o controle integral do pipeline de inferência, o que inclui a capacidade de inspecionar os pesos do modelo e controlar a base de dados de treinamento para mitigar a incorporação de vieses sistemáticos.

Essa combinação de exigências faz com que o Poder Judiciário opte prioritariamente pelo desenvolvimento e hospedagem local de modelos de pesos abertos. A infraestrutura dedicada, no âmbito da Plataforma Digital do Poder Judiciário (PDPJ-Br), atua como barreira técnica e institucional contra o compartilhamento de metadados jurisdicionais com corporações privadas globais, preservando a autonomia e a soberania do sistema de justiça.

9.3 Setor Bancário: Resiliência Operacional, Risco de Lock-in e Regulamentação

No setor bancário e financeiro, a proteção ao cliente e o Sigilo Bancário, resguardado pela Lei Complementar nº 105/2001 [23], exigem proteção absoluta contra o compartilhamento não autorizado de transações e dados cadastrais. No Brasil, o Banco Central (BACEN) regula as diretrizes de segurança cibernética e o uso de serviços de nuvem por meio da Resolução CMN nº 4.893/2021 [24], que impõe obrigações severas às instituições financeiras quanto à rastreabilidade, auditabilidade e continuidade operacional dos serviços tecnológicos críticos.

Um dos vetores de risco mais significativos para os bancos é a dependência de ponto único de falha. Ao delegar a inferência de LLMs inteiramente a uma API comercial de nuvem, a instituição subordina seus serviços de atendimento inteligente, análise de crédito

e detecção de fraude à disponibilidade de um provedor externo. Uma interrupção global do provedor pode paralisar operações críticas, o que é incompatível com as exigências do BACEN de planos robustos de continuidade de negócios. A implantação local ou híbrida confere ao banco o controle total sobre o *uptime* de seus sistemas de IA.

O risco de *vendor lock-in* é igualmente estratégico. A dependência tecnológica e comercial de um único fornecedor de API restringe a flexibilidade dos bancos de renegociar preços, adotar modelos mais avançados de outros fornecedores ou repatriar workloads para hardware próprio diante de oscilações tarifárias. Ao optar por modelos de pesos abertos hospedados em clusters próprios ou em arquiteturas *multi-cloud* privadas, as instituições financeiras preservam a portabilidade e a autonomia estratégica de longo prazo.

Há ainda o imperativo de auditabilidade contínua. O regulador exige que bancos tenham condições de auditar seus prestadores de serviços tecnológicos críticos, o que provedores de nuvem pública global limitam severamente em termos de acesso físico e inspeção de sistemas. Modelos hospedados localmente ou em nuvens privadas controladas eliminam essa barreira, garantindo conformidade plena com as exigências do BACEN. Soma-se a isso o fato de que grandes bancos, dado o elevado volume de transações diárias, atingem escalas de demanda de tokens em que o custo de APIs comerciais se torna proibitivo, tornando a amortização de supercomputadores próprios um investimento de payback relativamente curto e com custo marginal de operação extremamente baixo.

10 Conclusão

A investigação comparativa realizada neste estudo demonstra que a viabilidade de implantar Modelos de Linguagem de Grande Porte em infraestrutura local para sistemas agênticos em 2026 é condicionada por fatores econômicos, computacionais e regulatórios concorrentes.

Sob a perspectiva estritamente econômica, a consolidação das APIs de modelos abertos comoditizados alterou a dinâmica clássica de custos entre processamento local e em nuvem. A eletricidade necessária para a execução local de inferências em hardware de consumo supera, em regiões sob tarifas comerciais padrão, o custo de aquisição de tokens via APIs comerciais. Para modelos de grande escala, como o Llama 3.1 405B, a rápida obsolescência tecnológica aliada ao custo de depreciação do hardware de servidor limita a atratividade do investimento físico. O cálculo do ponto de equilíbrio econômico indica que o limite físico mensal de geração de um único nó de processamento corporativo é inferior ao volume demandado para justificar a transição financeira em relação a APIs de terceiros.

Por outro lado, em cenários nos quais a decisão de projeto é orientada por requisitos técnicos não financeiros (como latência estrita, operação em redes isoladas ou necessidade de conformidade com marcos legais como a LGPD [21]), a infraestrutura dedicada é a alternativa adequada. A análise setorial confirmou que o tratamento de dados estratégicos

governamentais, o segredo de justiça no Poder Judiciário [22] e a custódia de transações bancárias protegidas por sigilo operacional demandam controle absoluto dos pesos dos modelos e da segurança do tráfego de dados.

Nesse panorama, a abordagem otimizada para as organizações consiste na adoção de arquiteturas híbridas com roteamento inteligente de modelos. Essa estratégia integra a eficiência de custos das APIs comoditizadas para demandas de baixa complexidade, a capacidade de modelos de fronteira para inferências críticas e a soberania e baixa latência da computação local para dados confidenciais, maximizando o desempenho do sistema agêntico sob conformidade regulatória.

Declarações e Divulgações

Declaração de Disponibilidade de Dados

Os dados que apoiam as descobertas deste estudo, incluindo os parâmetros de hardware, estimativas de consumo de energia e valores tarifários de APIs, estão contidos no próprio artigo (Tabela 1 e Tabela 2) e são derivados de fontes públicas devidamente citadas. Os códigos de simulação e equações de ponto de equilíbrio estão integralmente descritos no manuscrito.

Contribuição dos Autores (CRediT)

- **João Pedro Hallack Sansão:** Conceitualização, Metodologia, Análise Formal, Investigação, Escrita — esboço original, Escrita — revisão e edição.

Declaração de Conflito de Interesses

O autor declara que não há conflitos de interesses financeiros, profissionais ou pessoais que possam ter influenciado a realização ou os resultados deste estudo.

Financiamento

O presente trabalho não recebeu financiamento ou auxílio financeiro de agências de fomento ou de entidades privadas, sendo realizado com recursos próprios do autor.

Referências

[1] YAO, S. et al. **ReAct: Synergizing Reasoning and Acting in Language Models**. In: International Conference on Learning Representations (ICLR), 2023.

[2] WEI, J. et al. **Chain-of-Thought Prompting Elicits Reasoning in Large Language Models**. In: Advances in Neural Information Processing Systems (NeurIPS), 2022.

[3] WU, Q. et al. **AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversation**. arXiv preprint arXiv:2308.08155, 2023.

[4] META AI. **The Llama 3.1 and 3.3 Series Technical Report**. Meta AI Research, 2024.

[5] ALIBABA CLOUD. **Qwen 2.5 Technical Report: Enhancing Coding and Reasoning capabilities of Open Models**. Alibaba AI Group, 2024.

[6] GOOGLE DEEPMIND. **Gemma 2: Improving Performance-to-Size Efficiency in Open Models**. Google Research, 2024.

[7] WANG, L. et al. **A Survey on Large Language Model based Autonomous Agents**. Frontiers of Computer Science, v. 18, n. 6, p. 186345, 2024.

[8] POPE, R. et al. **Efficiently Scaling Transformer Inference**. In: Proceedings of Machine Learning and Systems (MLSys), 2023.

[9] LOCAL AI MASTER. **VRAM Requirements for LLMs in 2026**. Local AI Master Blog, 2026. Disponível em: <https://localaimaster.com/blog/vram-requirements-2026>. Acesso em: 30 jun. 2026.

[10] VLLM PROJECT. **vLLM: Easy, Fast, and Cheap LLM Serving with PagedAttention and FP8 Engine**. Disponível em: <https://github.com/vllm-project/vllm>. Acesso em: 30 jun. 2026.

[11] NVIDIA CORPORATION. **Inference Performance Guide: TensorRT-LLM and FP8 Execution on Hopper and Blackwell Architectures**. NVIDIA Technical Docs, 2025.

[12] ARTIFICIAL ANALYSIS. **Inference Provider Benchmarks, Latency and Pricing Index**. Disponível em: <https://artificialanalysis.ai>. Acesso em: 30 jun. 2026.

[13] SHENG, Y. et al. **FlexGen: High-Throughput Generative Inference of Large Language Models with a Single GPU**. In: International Conference on Machine Learning (ICML), 2023.

[14] CHEN, L.; ZAHARIA, M.; ZOU, J. **FrugalGPT: How to Use Large Language Models While Reducing Cost and Improving Performance**. arXiv preprint arXiv:2305.05176, 2023.

[15] GROQ INC. **Real-time LPU Inference Benchmarks for Open-Weights Models**. Disponível em: <https://groq.com>. Acesso em: 30 jun. 2026.

[16] OPENROUTER. **API Models and Live Pricing Dashboard**. Disponível em: <https://openrouter.ai/models>. Acesso em: 30 jun. 2026.

[17] TOGETHER AI. **Together AI Developer API Documentation and Pricing Models**. Disponível em: <https://www.together.ai/pricing>. Acesso em: 30 jun. 2026.

[18] DEEPSEEK RESEARCH. **DeepSeek V4 Framework and Pricing Schedule for Edge-to-Cloud AI Infrastructure**. DeepSeek Inc. Technical Report, 2026.

[19] GIM, I. et al. **Prompt Cache: Modular Attention Reuse for Low-Latency Inference**. In: Proceedings of Machine Learning and Systems (MLSys), 2024.

[20] UNITED STATES. **Clarifying Lawful Overseas Use of Data Act (CLOUD Act)**. H.R. 4943, 115th Congress, 2018.

[21] BRASIL. **Lei nº 13.709, de 14 de agosto de 2018**. Lei Geral de Proteção de Dados Pessoais (LGPD). Diário Oficial da União, Brasília, DF, 15 ago. 2018.

[22] CONSELHO NACIONAL DE JUSTIÇA (CNJ). **Resolução nº 332, de 21 de agosto de 2020**. Dispõe sobre a ética, a transparência e a governança na produção e no uso de Inteligência Artificial no Poder Judiciário. CNJ, Brasília, DF, 2020.

[23] BRASIL. **Lei Complementar nº 105, de 10 de janeiro de 2001**. Dispõe sobre o sigilo das operações de instituições financeiras e dá outras providências. Diário Oficial da União, Brasília, DF, 11 jan. 2001.

[24] BANCO CENTRAL DO BRASIL. **Resolução CMN nº 4.893, de 26 de fevereiro de 2021**. Dispõe sobre a política de segurança cibernética e sobre os requisitos para a contratação de serviços de processamento e armazenamento de dados e de computação em nuvem. Diário Oficial da União, Brasília, DF, 1 mar. 2021.

Este preprint foi submetido sob as seguintes condições:

- Os autores declaram que os necessários Termos de Consentimento Livre e Esclarecido de participantes ou pacientes na pesquisa foram obtidos e estão descritos no manuscrito, quando aplicável.
- Os autores declaram que a elaboração do manuscrito seguiu as normas éticas de comunicação científica.
- Os autores declaram que estão cientes que são os únicos responsáveis pelo conteúdo do preprint e que o depósito no SciELO Preprints não significa nenhum compromisso de parte do SciELO, exceto sua preservação e disseminação.
- Os autores declaram que os dados, aplicativos e outros conteúdos subjacentes ao manuscrito estão referenciados.
- O manuscrito depositado está no formato PDF.
- Os autores declaram que a pesquisa que deu origem ao manuscrito seguiu as boas práticas éticas e que as necessárias aprovações de comitês de ética de pesquisa, quando aplicável, estão descritas no manuscrito.
- Os autores declaram que uma vez que um manuscrito é postado no servidor SciELO Preprints, o mesmo só poderá ser retirado mediante pedido à Secretaria Editorial do SciELO Preprints, que afixará um aviso de retratação no seu lugar.
- Os autores concordam que o manuscrito aprovado será disponibilizado sob licença Creative Commons CC-BY.
- O autor submissor declara que as contribuições de todos os autores e declaração de conflito de interesses estão incluídas de maneira explícita e em seções específicas do manuscrito.
- Os autores declaram que o manuscrito não foi depositado e/ou disponibilizado previamente em outro servidor de preprints ou publicado em um periódico.
- Caso o manuscrito esteja em processo de avaliação ou sendo preparado para publicação mas ainda não publicado por um periódico, os autores declaram que receberam autorização do periódico para realizar este depósito.
- O autor submissor declara que todos os autores do manuscrito concordam com a submissão ao SciELO Preprints.