RECA
REVISTA ELETRÔNICA DE COMPUTAÇÃO APLICADA
Vol. 6 Nro. 2 2025

Departamento de Computação
Uni-FACEF

Avaliação Comparativa de Modelos de Linguagem através de Benchmark Multidimensional e Julgamento Automático por LLM-as-a-Judge

Guilherme Maranha Silva
Graduando em Ciência da Computação – Uni-FACEF
gmaranha4@gmail.com

Prof. Dr. Daniel Facciolo Pires
Docente do Departamento de Computação – Uni-FACEF
daniel@facef.br

RESUMO

Modelos de Linguagem de Grande Escala (LLMs) têm se tornado fundamentais em aplicações de inteligência artificial voltadas para tarefas textuais, como geração criativa, tradução, resumo e resolução de problemas. Contudo, comparar objetivamente diferentes modelos requer métricas padronizadas de desempenho e qualidade. Este trabalho apresenta um estudo quantitativo e qualitativo envolvendo quatro modelos amplamente utilizados — Llama 3 70B, Mixtral 8x7B, Llama 3 8B e Mistral 7B — avaliados em 50 *prompts* distribuídos em cinco categorias: Criatividade, Resumo, Tradução, Q&A factual (Perguntas e Respostas Factuais) e Matemática. O desempenho técnico foi mensurado por tempo médio de resposta, enquanto a qualidade subjetiva foi avaliada por meio de um método automatizado conhecido como *LLM-as-a-Judge*, utilizando o modelo Llama 3 70B como avaliador gratuito via *OpenRouter*. Foram gerados 200 *outputs* analisados em múltiplos critérios. Os resultados indicam diferenças relevantes entre os modelos, tanto em velocidade quanto em qualidade textual. A combinação de *benchmarking* e avaliação automática demonstrou ser eficaz, reprodutível e escalável. Conclui-se que métodos baseados em *LLM-as-a-Judge* são promissores como alternativa para avaliação humana em experimentos com LLMs.

Palavras-chave: inteligência artificial, modelos de linguagem, benchmark, avaliação automática, LLM-as-a-Judge.

ABSTRACT

Large Language Models (LLMs) have become essential in artificial intelligence applications designed for textual tasks, including creative generation, translation, summarization, and problem solving. However, objectively comparing different models requires standardized metrics for performance and quality. This study presents a quantitative and qualitative evaluation involving four widely used models - Llama 3 70B, Mixtral 8x7B, Llama 3 8B, and Mistral 7B - tested across 50 *prompts* distributed into five categories: Creativity, Summarization, Translation, Factual Q&A, and Mathematics. Technical performance was measured through average response time, while subjective quality was assessed

using an automated method known as *LLM-as-a-Judge*, employing Llama 3 70B as the evaluator via OpenRouter. A total of 200 outputs were generated and analyzed using multiple criteria. The results highlight relevant differences among the models in both speed and textual quality. The combination of benchmarking and automated evaluation proved effective, reproducible, and scalable. The study concludes that LLM-as-a-Judge is a promising alternative to human evaluation in LLM experiments.

Keywords: artificial intelligence, language models, benchmarking, automated evaluation, LLM-as-a-Judge.

1. INTRODUÇÃO

Large Language Models (LLMs) tornaram-se essenciais em sistemas inteligentes de diversas áreas, incluindo educação, saúde, atendimento automatizado e desenvolvimento de software. Com o crescimento acelerado de modelos de código aberto — como a família Llama 3 e os modelos da linha Mistral — surge a necessidade de comparar essas arquiteturas em diferentes dimensões: desempenho computacional, qualidade linguística, precisão factual e capacidade de generalização.

A popularização de APIs de baixo custo e de modelos gratuitos, como os disponibilizados via *OpenRouter*, permite que pesquisadores independentes realizem *benchmarks* abrangentes. No entanto, a avaliação humana é lenta, subjetiva e inviável quando se tratam de centenas ou milhares de *outputs*. Assim, nasce o interesse em empregar LLMs como avaliadores automáticos, em um método chamado *LLM-as-a-Judge*.

Diante desse cenário, o problema investigado é: “Como comparar, de maneira objetiva, replicável e de baixo custo, o desempenho e a qualidade textual de diferentes LLMs em múltiplas categorias de tarefas?”

O objetivo deste trabalho é avaliar e comparar o desempenho e a qualidade de diferentes modelos de linguagem por meio de um *benchmark* padronizado com 50 *prompts* e avaliação automática baseada em *LLMs-as-a-Judge*.

Como objetivos específicos:

- Mensurar o tempo de resposta de quatro LLMs em cinco categorias de tarefas.
- Avaliar a qualidade gerada pelos modelos usando *LLM-as-a-Judge*.
- Gerar métricas comparativas globais entre os modelos.
- Discutir vantagens e limitações do método de avaliação automatizada.

O estudo utilizou um conjunto de 50 *prompts* divididos em cinco categorias. Cada prompt foi enviado a quatro modelos, resultando em 200 chamadas de API. A qualidade textual nas categorias subjetivas foi avaliada automaticamente por um LLM-juiz (Llama 3 70B). Os dados foram tabulados, analisados e sintetizados em tabelas comparativas e métricas agregadas.

2. REVISÃO TEÓRICA

A revisão teórica deste trabalho aborda os principais conceitos relacionados aos modelos de linguagem de grande porte *Large Language Models* (LLMs), com ênfase em suas arquiteturas, capacidades e limitações. São discutidas as abordagens de avaliação utilizadas na literatura, incluindo métodos baseados em benchmarks tradicionais e estratégias de análise qualitativa, destacando os desafios inerentes à mensuração de desempenho em tarefas subjetivas. Além disso, são apresentados fundamentos do uso de modelos de linguagem como avaliadores automáticos, conhecidos como LLM-as-a-Judge, ressaltando seu papel na avaliação comparativa e multidimensional de modelos, bem como suas implicações em termos de confiabilidade, consistência e alinhamento com julgamentos humanos.

2.1 Modelos de Linguagem e Arquiteturas de Inteligência Artificial

A evolução das técnicas de Inteligência Artificial (IA) ao longo das últimas décadas foi marcada por avanços significativos tanto na capacidade computacional quanto nos métodos algorítmicos. Tradicionalmente, sistemas inteligentes utilizavam abordagens baseadas em regras explícitas, lógicas formais e representações simbólicas. No Brasil, autores como Rezende (2003) destacam que, até o início dos anos 2000, a IA aplicada seguia predominantemente paradigmas simbólicos, centrados na programação da lógica do sistema, com pouca autonomia adaptativa.

Com o advento de técnicas conexionistas e redes neurais profundas (*deep learning*), tornou-se possível treinar modelos capazes de aprender padrões complexos diretamente a partir de grandes volumes de dados. Essa mudança promoveu um salto tecnológico que possibilitou o desenvolvimento dos Modelos de Linguagem de Grande Escala (LLMs, do inglês *Large Language Models*). Esses modelos são treinados com bilhões ou trilhões de tokens, permitindo-lhes gerar textos coerentes, realizar inferências linguísticas, responder perguntas e produzir conteúdo em múltiplos estilos e domínios.

A base estrutural desses modelos é a arquitetura *Transformer*, que, embora criada no contexto internacional, rapidamente foi incorporada e analisada por pesquisadores brasileiros. Alguns deles destacam como mecanismos de atenção - essenciais no Transformer - substituem a dependência estrita de sequências temporais, permitindo que o modelo avalie relações de longo alcance dentro de um texto. Isso melhora significativamente tarefas como tradução, resumo e interpretação contextual, ampliando a capacidade dos LLMs de lidar com ambiguidades e características linguísticas complexas, como ocorre no português brasileiro.

Outro fator relevante é a escalabilidade. Conforme Souza e Torres (2021) observam, modelos maiores não apenas armazenam mais parâmetros, mas também apresentam maior capacidade de generalização e refinamento semântico. No entanto, isso implica custos computacionais maiores, exigindo

otimizações de hardware, estruturas de paralelismo e técnicas como *mixture-of-experts* (MoE), hoje presentes em modelos como o Mixtral 8x7B.

Assim, os LLMs representam um marco evolutivo dentro da IA moderna: combinam densidade de parâmetros, arquiteturas otimizadas, aprendizagem massiva e capacidades emergentes, mantendo-se como elementos centrais de sistemas inteligentes contemporâneos — o que torna sua avaliação sistemática fundamental.

2.2 Processamento de Linguagem Natural no Brasil

A pesquisa em Processamento de Linguagem Natural (PLN) tem tradição consolidada no Brasil desde os anos 1990, com grupos como o NILC/USP, TeMA/UFPR e o grupo de Linguística Computacional da UNICAMP, que desenvolveram importantes ferramentas para o português brasileiro. Aluísio e Pardo (2010), por exemplo, discutem as particularidades do português no contexto computacional, ressaltando desafios relacionados à flexão verbal, concordância nominal, ordem sintática variável e riqueza morfológica — fatores que tornam a língua mais desafiadora para modelos estatísticos.

No contexto mais recente, trabalhos nacionais (Avila, Real, Fonseca, 2020; Santos & Souza, 2021) demonstram que modelos baseados em *Transformers* tiveram um impacto expressivo no desempenho de tarefas linguísticas. Estudos relatam reduções significativas em erros de tradução, melhorias na fluidez de resumos automáticos e avanços na detecção de entidades nomeadas. Esses avanços são fundamentais porque consolidam a ideia de que LLMs generalistas podem, mesmo sem treinamento específico na língua portuguesa, desempenhar atividades linguísticas com qualidade comparável à de modelos especializados.

Outro aspecto relevante no cenário brasileiro é o crescimento de bases textuais nacionais, como corpus NILC, corpus Tycho Brahe e bancos de dados que sustentam iniciativas como o PLN-BR e o BERTimbau, que representam marcos importantes de adaptação da tecnologia para a língua portuguesa. Conforme Fernandes e Vieira (2022) apontam, a existência de corpora anotados em português é essencial para pesquisas baseadas em avaliação sistemática, pois permite testar modelos em condições reais e culturalmente adequadas.

Com a popularização de modelos multilíngues e APIs abertas, o Brasil vivencia a democratização do acesso à IA. Pequenos grupos de pesquisa, centros universitários e até mesmo alunos de graduação conseguem, hoje, testar e comparar modelos de ponta — o que dialoga diretamente com o tipo de *benchmark* apresentado neste estudo.

2.3 Avaliação de Modelos e Benchmarking em IA e PLN

A avaliação de modelos de linguagem é um tema recorrente em pesquisas acadêmicas, especialmente porque os LLMs apresentam comportamentos variados conforme a tarefa, o tamanho do modelo e o tipo de arquitetura. Rezende (2003) enfatiza que, no campo dos sistemas inteligentes, métricas objetivas — como tempo de execução, acurácia e precisão — são fundamentais para medir eficiência. Entretanto, para tarefas de linguagem, essas

métricas quantitativas são insuficientes, pois não capturam nuances como coesão, correção semântica ou criatividade.

Nesse contexto, *benchmarks* são definidos como conjuntos padronizados de tarefas, dados e métricas, projetados para avaliar de forma sistemática e comparável o desempenho de modelos em cenários controlados. Em Inteligência Artificial e Processamento de Linguagem Natural, os benchmarks desempenham um papel central ao fornecer uma base comum de comparação, permitindo analisar a capacidade de generalização, robustez e consistência dos modelos frente a diferentes tipos de tarefas linguísticas. A adoção de *benchmarks* bem definidos contribui para reduzir vieses experimentais e facilita a reprodutibilidade dos resultados, aspecto essencial em pesquisas científicas.

Além disso, benchmarks permitem a avaliação multidimensional dos modelos, ao abranger tanto tarefas objetivas — como classificação textual e tradução automática — quanto tarefas mais abertas, como sumarização e geração de texto. No Brasil, eventos como o STIL (*Symposium in Information and Language Technology*) frequentemente apresentam estudos comparativos baseados em *benchmarks*, refletindo o interesse da comunidade científica nacional no desenvolvimento de métodos de avaliação robustos e padronizados.

Fernandes e Vieira (2022), em sua revisão sistemática sobre avaliação em PLN, pontuam que métodos tradicionais dependem fortemente da anotação humana, o que torna a avaliação lenta, subjetiva e pouco escalável. Além disso, como apontam Barbosa; Fernandes; Castro (2022), a velocidade de desenvolvimento dos LLMs torna impraticável manter grandes equipes humanas dedicadas à avaliação contínua de modelos.

Diante disso, a literatura brasileira começa a discutir métodos alternativos capazes de permitir avaliação ágil, replicável e precisa — sendo a abordagem *LLM-as-a-Judge* uma das mais promissoras.

2.4 Avaliação Automática de Texto: LLM-as-a-Judge

O método *LLM-as-a-Judge* consiste em utilizar um modelo de linguagem para avaliar as respostas de outros modelos. Pesquisadores brasileiros têm investigado esse método como alternativa à avaliação humana. Silva e Bermejo (2023) destacam que o uso de LLMs avaliadores apresenta alta correlação com julgamentos humanos em tarefas de criatividade e qualidade textual. Além disso, os autores observam que modelos maiores produzem avaliações mais estáveis, especialmente quando orientados com *system prompts* estruturados e saídas restritas ao formato JSON.

O método oferece benefícios expressivos:

1. Custo extremamente baixo – elimina a necessidade de avaliadores humanos;
2. Escalabilidade – permite avaliar centenas ou milhares de respostas rapidamente;
3. Consistência – reduz subjetividade individual;

4. Reprodutibilidade – a mesma entrada sempre resulta em padrão similar de avaliação.

No contexto acadêmico brasileiro, esse tipo de abordagem tem ganhado espaço em pesquisas sobre ensino de IA, automação de correções textuais e desenvolvimento de sistemas de avaliação assistida por computador. A aplicação prática também se expande para setores como educação, atendimento ao cliente e análise automática de qualidade de textos gerados por chatbots corporativos.

Pesquisas nacionais (Barbosa; Fernandes; Castro, 2022) apontam ainda que, embora o método seja poderoso, ele não substitui completamente a avaliação humana em tarefas altamente subjetivas, como literatura, humor e tons emocionais sutis. Contudo, para tarefas estruturadas — como as deste estudo — o uso de LLM-as-a-Judge demonstra enorme viabilidade.

2.5 LLMs Abertos e Democratização da IA no Brasil

A disseminação de tecnologias abertas de IA tem impacto direto no ecossistema brasileiro. Souza e Torres (2021) analisam que modelos *open-source*, aliados a plataformas de acesso unificado como *OpenRouter*, oferecem oportunidades inéditas para instituições com infraestrutura limitada. Isso é especialmente importante no contexto da pesquisa universitária brasileira, onde muitas vezes os recursos computacionais são restritos.

A possibilidade de acessar modelos complexos por meio de APIs públicas, sem necessidade de servidores de grande porte, cria condições ideais para projetos independentes de teste, comparação e avaliação, como este *benchmark* apresentado. Tal estrutura democratiza a IA e incentiva experimentação, inovação e formação de estudantes nas áreas de computação e ciência de dados.

Além disso, a crescente participação brasileira em eventos como o Congresso Brasileiro de Inteligência Artificial demonstra que o país está alinhado às tendências globais de avaliação, governança e aplicação ética de IA. Os *benchmarks* nacionais ajudam a consolidar boas práticas e ampliam o repertório metodológico disponível para pesquisadores, docentes e profissionais.

3. PROCEDIMENTOS METODOLÓGICOS

A seção descreve detalhadamente todas as etapas metodológicas adotadas na pesquisa, desde a seleção dos modelos de linguagem avaliados até a definição dos *prompts*, procedimentos de coleta de dados, estratégias de avaliação automática e formas de tabulação e análise dos resultados. A metodologia foi construída para garantir reprodutibilidade, comparabilidade, controle experimental e rastreamento completo dos *outputs* gerados, características fundamentais em estudos de *benchmarking* de Modelos de Linguagem de Grande Escala (LLMs).

3.1. Modelos Avaliados

A pesquisa avaliou quatro Modelos de Linguagem amplamente utilizados na comunidade de inteligência artificial, todos disponibilizados por meio do serviço *OpenRouter*. A seleção dos modelos de linguagem para este estudo foram Llama 3 70B, Mixtral 8x7B, Llama 3 8B e Mistral 7B, escolhidas por critérios de acessibilidade, representatividade técnica e viabilidade econômica.

A escolha priorizou modelos de código aberto ou amplamente disponíveis que pudessem ser acessados de forma gratuita, permitindo que a pesquisa fosse conduzida sem a necessidade de investimentos em infraestrutura de hardware de alto custo ou assinaturas de APIs proprietárias.

Os modelos avaliados foram:

- Llama 3 70B Instruct – Modelo de grande porte, com aproximadamente 70 bilhões de parâmetros. Devido ao seu tamanho, é reconhecido por gerar respostas mais detalhadas, coerentes e com maior profundidade semântica, sendo considerado o modelo de mais alto desempenho entre os avaliados.
- Mixtral 8x7B Instruct – Arquitetura baseada em *Mixture-of-Experts* (MoE), capaz de combinar eficiência e desempenho por meio da ativação parcial dos especialistas internos. Apresenta custo computacional reduzido em comparação com modelos densos de grande porte, mantendo alta qualidade textual.
- Llama 3 8B Instruct – Versão compacta da família Llama 3, destinada a cenários de baixo custo computacional. Possui aproximadamente 8 bilhões de parâmetros e é frequentemente usado em aplicações com restrições de latência.
- Mistral 7B Instruct v0.2 – Modelo também compacto, com 7 bilhões de parâmetros, conhecido por equilíbrio entre velocidade e qualidade, frequentemente utilizado como baseline em estudos de PLN.

A inclusão desses modelos permite comparar diferentes escalas (7B, 8B, 70B), diferentes arquiteturas (densas e MoE) e diferentes propósitos (velocidade vs. qualidade), possibilitando uma análise abrangente do desempenho em tarefas textuais.

As nomenclaturas dos modelos referem-se, primordialmente, à sua escala e arquitetura interna, fatores que influenciam diretamente o equilíbrio entre velocidade e qualidade textual. Os números acompanhados da letra "B" (como *70B, 8B e 7B*) indicam a quantidade de bilhões de parâmetros que compõem o modelo; em teoria, modelos com maior número de parâmetros, como o Llama 3 70B, possuem maior capacidade de representação semântica e complexidade linguística. Já o modelo Mixtral 8x7B utiliza uma arquitetura de *Mixture-of-Experts* (MoE), que, embora possua um total elevado de parâmetros, ativa apenas uma fração deles para cada tarefa específica, visando otimizar a eficiência computacional sem sacrificar o desempenho.

3.2. Conjunto de Prompts

O conjunto experimental foi composto por 50 *prompts*, distribuídos igualmente em cinco categorias, definidas com o objetivo de avaliar diferentes dimensões da competência linguística dos modelos analisados. A **Tabela 3.2** apresenta a organização dessas categorias, bem como a quantidade de *prompts* atribuída a cada uma e as respectivas habilidades avaliadas.

Tabela 3.2 – Categorias de prompts e habilidades avaliadas

Categoria	Quantidade	Habilidade Avaliada
Criatividade	10	Capacidade de geração textual original, narrativa e aderência a formatos criativos
Q&A Factual	10	Recuperação de conhecimento factual, precisão objetiva e respostas diretas
Matemática	10	Raciocínio lógico, aritmética básica e manipulação de expressões numéricas
Resumo	10	Capacidade de condensação, preservação de ideias centrais e organização textual
Tradução	10	Competência bilíngue, preservação semântica e fluidez no idioma-alvo

Fonte: Elaborado pelo autor (2025).

O total de 200 *outputs* gerados (50 *prompts* × 4 modelos) permitiu uma análise estatística consistente, contemplando variações por categoria e por modelo.

3.3. Procedimento de Benchmarking

A etapa de *benchmarking* consistiu na execução sistemática de todos os *prompts* em todos os modelos selecionados, utilizando uma estrutura automatizada desenvolvida em Python. Para garantir padronização, reprodutibilidade e controle experimental, todo o código-fonte, bem como os scripts utilizados na condução dos experimentos, foram disponibilizados em um repositório público no Github[1]. Para garantir padronização e controle, o experimento seguiu os passos:

Para a execução dos testes e coleta de dados, utilizou-se o *OpenRouter*, uma plataforma que atua como um agregador unificado de APIs para diversos modelos de linguagem de grande escala. O *OpenRouter* facilita o acesso a uma vasta gama de LLMs, tanto proprietários quanto de código aberto, através de uma interface padronizada, o que simplifica a implementação técnica do *benchmark* ao permitir que diferentes arquiteturas sejam consultadas com o mesmo protocolo de integração. Além da facilidade de uso, a plataforma foi fundamental por disponibilizar versões gratuitas dos modelos selecionados, garantindo a reprodutibilidade dos experimentos sob uma estrutura de baixo custo.

[1] Repositório do projeto disponível em: https://github.com/GuilhermeMaranha/TCC_guilherme

(1) Envio dos *prompts* aos modelos

Cada prompt foi enviado por meio da biblioteca LiteLLM, responsável por padronizar chamadas à API *OpenRouter*, mantendo consistência independentemente do modelo utilizado.

(2) Registro automático do tempo de resposta

A latência foi medida com time.time(), registrando o instante da chamada, o instante de recebimento da resposta, e a diferença entre ambos (latência total).

Essa métrica é fundamental para comparar eficiência computacional e reatividade.

(3) Captura integral do conteúdo textual

O texto retornado pelo modelo foi armazenado sem qualquer modificação, preservando: conteúdo literal, quebras de linha, variações de estilo e eventuais erros de formatação.

Isso assegura fidelidade na avaliação posterior.

(4) Registro de erros

Erros foram classificados como: APIError – falha do provedor, APIConnectionError – instabilidade de rede, e Exception – falhas internas imprevistas.

Cada erro foi armazenado no DataFrame para análise futura e atribuição de nota zero.

(5) Armazenamento estruturado

Os dados coletados formaram um DataFrame pandas contendo: Categoria, Prompt, Modelo, Latência, Resposta e Status (OK/ERRO).

(6) Exportação dos resultados

Os 200 resultados foram exportados para arquivos CSV:

- comparacao_llm_50_prompts.csv – sem notas
- comparacao_llm_50_prompts_com_notas_gratuito.csv – com avaliação

Esse repositório estruturado permitiu a realização das análises de desempenho e qualidade.

3.4. Avaliação de Qualidade (LLM-as-a-Judge)

As categorias subjetivas — Criatividade, Resumo e Tradução — demandam avaliação qualitativa, o que torna inadequado o uso exclusivo de métricas tradicionais de Processamento de Linguagem Natural. Para contornar essa limitação, foi adotado o método *LLM-as-a-Judge*, no qual um modelo de linguagem é utilizado como avaliador automático das respostas geradas.

O modelo selecionado como juiz foi o Llama 3 70B Instruct. A implementação completa do processo de avaliação, incluindo os critérios utilizados, os *prompts* de julgamento e a estrutura de consolidação das notas, encontra-se documentada e disponibilizada no *openrouter*[2], assegurando transparência e reprodutibilidade experimental.

Modelo avaliador

O modelo selecionado como juiz foi o:

- Llama 3 70B Instruct.

A escolha levou em consideração: maior estabilidade avaliativa, maior capacidade semântica, forte correlação com avaliações humanas, e acesso gratuito via OpenRouter.

Procedimento de avaliação

Para cada resposta gerada pelos modelos, o avaliador recebeu: Categoria, Prompt original, Resposta dada pelo modelo, e Critérios específicos.

O *system prompt* exigia explicitamente o formato de saída:

```
{
  "nota": 1-5,
  "justificativa": "texto explicativo"
}
```

Critérios de avaliação por categoria

- Criatividade
  - originalidade;
  - estilo narrativo;
  - aderência ao número de frases;
  - coerência interna.
- Resumo
  - fidelidade ao conteúdo central;
  - ausência de distorções;
  - clareza e coesão;
  - concisão.
- Tradução
  - preservação do sentido;
  - naturalidade do texto;
  - correção gramatical;

[2] https://openrouter.ai/meta-llama/llama-3-70b-instruct

- fluidez no idioma-alvo.

Integração ao DataFrame

Os campos adicionados foram:

- Nota_Qualidade;
- Justificativa.

Erros de execução recebiam nota zero automaticamente.

Controle e padronização

Para garantir consistência, a temperatura do juiz foi fixada em 0.1 e um *delay* de um segundo foi aplicado entre chamadas para evitar *rate-limit* da API. Em modelos de linguagem, a temperatura controla a aleatoriedade da geração de texto: valores próximos a zero tornam a saída mais determinística, focada na escolha das palavras mais prováveis de acordo com o treinamento do modelo. A escolha de uma temperatura baixa para o juiz é fundamental para garantir a **consistência** e a **reprodutibilidade** do julgamento; isso assegura que o modelo avalie os critérios de forma objetiva e estável, reduzindo variações criativas que poderiam comprometer a padronização das notas atribuídas aos diferentes *outputs*.

Uma nuance metodológica relevante deste estudo reside no fato de o modelo Llama 3 70B ter sido utilizado simultaneamente como um dos modelos avaliados e como o juiz central do experimento. Essa configuração pode introduzir o que a literatura define como viés arquitetural, uma vez que um modelo tende a favorecer respostas que apresentem padrões estilísticos, de estruturação e de raciocínio similares aos seus próprios. Embora o uso de um modelo de maior porte como avaliador seja justificado por sua maior estabilidade semântica e correlação com o julgamento humano, a similaridade de família entre o juiz e o avaliado (Llama) deve ser considerada na interpretação dos resultados, uma vez que o avaliador pode demonstrar uma predisposição inerente a validar positivamente a lógica de sua própria arquitetura. Para mitigar distorções, a aplicação de *system prompts* rigorosos e a limitação da saída ao formato JSON foram adotadas, buscando garantir que a avaliação se mantivesse estritamente pautada nos critérios objetivos de cada categoria.

3.5. Exemplos de pergunta em cada categoria

- Criatividade: "Escreva uma fábula (6–8 frases) sobre a amizade entre um rio e uma ponte."
- Q&A Factual: "Quem foi o primeiro presidente do Brasil?"
- Matemática: "Calcule 25 + 37."
- Resumo: "Resuma em 3–4 frases: reciclagem reduz extração, economiza energia e diminui poluição; coleta seletiva eficaz reduz aterros e gera empregos verdes; reduzir e reutilizar têm impacto maior que reciclar."
- Tradução: "Traduza para inglês: 'O conhecimento é poder.'"

4. RESULTADOS E DISCUSSÃO

Esta seção apresenta e discute os resultados obtidos no experimento envolvendo quatro modelos de linguagem avaliados em 50 *prompts* distribuídos em cinco categorias.

A **Figura 1** apresenta o tempo médio de resposta, por categoria de tarefa, para cada um dos quatro modelos avaliados. Essa métrica é fundamental para compreender a eficiência operacional dos modelos, especialmente em cenários de uso real onde a latência influencia diretamente a experiência do usuário, como *chatbots*, sistemas de suporte ao cliente e aplicações em dispositivos móveis.

FIGURA 4.1. Resumo de desempenho médio

Modelo_Nome	Criatividade	Matemática	Q&A Factual	Resumo	Tradução	Média Total
Llama-3-8B	3.16s	0.75s	2.11s	2.07s	0.73s	1.76s
Llama-3-70B	4.15s	1.21s	1.30s	3.16s	0.77s	2.12s
Mixtral-8x7B	5.45s	2.05s	1.54s	3.40s	1.69s	2.83s
Mistral-7B	17.11s	1.42s	1.06s	1.43s	0.64s	4.33s

Fonte: Elaborado pelo autor (2025).

4.1 Interpretação Geral dos Resultados

A análise inicial dos dados revela que nenhum dos modelos avaliados apresenta uma dominância absoluta em todas as categorias testadas. Esse cenário indica que o desempenho de cada inteligência artificial é profundamente influenciado por fatores estruturais e contextuais, tais como a arquitetura do modelo, o volume total de parâmetros, o nível de otimização interna e o grau de complexidade inerente a cada *prompt* enviado.

Nesse contexto, observou-se um padrão comportamental distinto entre as diferentes escalas de modelos: enquanto as arquiteturas menores se destacam pela maior agilidade no tempo de resposta, os modelos de grande porte tendem a apresentar uma latência superior, compensada, no entanto, por uma maior consistência e estabilidade nos resultados gerados.

4.2 Análise Comparativa entre Modelos e Categorias

A comparação integrada entre os quatro modelos avaliados revela diferenças claras tanto no desempenho técnico quanto no comportamento por categoria. O Llama-3-8B destaca-se como o modelo mais rápido, com média geral de 1,76 segundos, especialmente eficiente em tarefas objetivas como Matemática e Tradução, nas quais a geração textual é curta. Entretanto, seu desempenho se eleva nas tarefas mais abertas, como Criatividade, indicando limitações esperadas em modelos menores.

O Llama-3-70B, apesar de ser o maior modelo da análise, apresentou tempos surpreendentemente competitivos, alcançando média de 2,12 segundos. Seu comportamento é mais estável e ele tende a gerar textos mais elaborados, o que explica tempos maiores em categorias como Criatividade e Resumo. Ainda assim, seu desempenho em Q&A Factual foi um dos melhores, refletindo sua maior capacidade de compreensão contextual e precisão factual.

O Mixtral-8x7B ocupa uma posição intermediária, apresentando média de 2,83 segundos. Seu desempenho equilibrado decorre de sua arquitetura baseada em *Mixture-of-Experts*, que ativa diferentes especialistas conforme a tarefa. Isso torna o Mixtral mais lento em tarefas longas — como Criatividade e Resumo — mas bastante competitivo em Q&A e Tradução.

O Mistral-7B é o modelo com maior instabilidade: embora rápido em categorias objetivas como Q&A e Tradução — sendo o mais veloz desta última — apresenta latência muito elevada em Criatividade, chegando a 17,11 segundos. Isso demonstra que, apesar de eficiente para respostas curtas e diretas, modelos menores têm dificuldades em produzir textos complexos, narrativos ou que exigem grande manutenção de contexto.

Comparando-se as categorias, observa-se que Matemática e Tradução são as tarefas mais rápidas para todos os modelos, refletindo sua baixa complexidade linguística. Q&A Factual também apresenta tempos reduzidos e pouca variação entre modelos, indicando ser a categoria menos sensível ao tamanho da arquitetura. Já Resumo e, principalmente, Criatividade, são as categorias mais desafiadoras, exigindo mais tempo de processamento e expondo as limitações de modelos menores. A maior discrepância ocorre justamente em Criatividade, cuja a diferença entre modelos pequenos e grandes torna-se mais evidente.

De modo geral, os resultados mostram que tarefas curtas favorecem modelos compactos, enquanto tarefas abertas beneficiam modelos maiores, que possuem mais capacidade de representação linguística. O Mixtral, por sua vez, se destaca como opção equilibrada, combinando boa qualidade textual com velocidade moderada. Esses achados reforçam padrões já previstos pela literatura e ajudam a orientar a escolha do modelo mais adequado dependendo da natureza da aplicação.

4.3 Análise da Qualidade Textual

A avaliação de qualidade textual, realizada por meio do método LLM-as-a-Judge utilizando o modelo Llama-3-70B como avaliador, permitiu atribuir notas de 1 a 5 às respostas nas categorias Criatividade, Resumo e Tradução. Os resultados médios estão apresentados na **Figura 4.3** e evidenciam diferenças claras entre as arquiteturas.

FIGURA 4.2. Nota média (Criatividade, Resumo e Tradução)

	Modelo	Média da Nota
0	Llama-3-70B	4.77
1	Mixtral-8x7B	4.60
2	Mistral-7B	4.37
3	Llama-3-8B	3.90

Fonte: Elaborado pelo autor (2025).

O Llama-3-70B apresentou a maior nota média (4,77), confirmando seu desempenho superior na produção de textos mais coesos, criativos e semanticamente ricos. O Mixtral-8x7B aparece logo em seguida (4,60), demonstrando excelente custo-benefício ao entregar qualidade próxima ao modelo de 70B parâmetros, mesmo com menor custo computacional. O Mistral-7B, com média de 4,37, superou o Llama-3-8B em qualidade, apesar de menor porte, indicando boa competência em tarefas objetivas e estruturadas. O Llama-3-8B, embora o mais rápido tecnicamente, obteve a menor média (3,90), refletindo limitações de profundidade linguística e criatividade.

Em síntese, os resultados revelam que modelos maiores tendem a produzir textos qualitativamente superiores, enquanto modelos intermediários — especialmente o Mixtral — oferecem equilíbrio entre qualidade e eficiência. Já os modelos menores demonstram boa precisão básica, mas menor capacidade em tarefas que exigem maior elaboração textual.

5. CONCLUSÃO

Este trabalho apresentou uma análise comparativa abrangente entre quatro modelos de linguagem — Llama-3-70B, Mixtral-8x7B, Llama-3-8B e Mistral-7B — utilizando um conjunto estruturado de 50 *prompts* distribuídos em cinco categorias e avaliados por meio de métricas objetivas de latência e métricas subjetivas de qualidade textual. A metodologia adotada, baseada em testes automatizados via LiteLLM e no uso da técnica LLM-as-a-Judge, demonstrou ser eficaz, escalável e de baixo custo, permitindo a execução de 200 gerações e 150 avaliações qualitativas de forma padronizada e reprodutível. Esse caráter automatizado reforça seu potencial para apoiar pesquisas acadêmicas em processamento de linguagem natural, especialmente em ambientes com recursos computacionais limitados.

Os resultados evidenciaram diferenças importantes entre as arquiteturas analisadas. O Llama-3-70B alcançou o melhor desempenho em termos de qualidade textual, confirmando a relação entre o número de parâmetros e a profundidade linguística. Já o Mixtral-8x7B destacou-se por oferecer o melhor equilíbrio entre velocidade, consistência e qualidade, representando uma alternativa altamente vantajosa para aplicações que exigem robustez com menor custo computacional. Os modelos menores, como Mistral-7B e Llama-3-8B, apresentaram tempos de resposta mais baixos em tarefas objetivas, mas limitações perceptíveis em tarefas abertas e criativas. Esse conjunto de achados reforça observações da literatura brasileira sobre PLN, que destacam a necessidade de alinhar as características do modelo às demandas específicas da aplicação.

Apesar da robustez metodológica, este estudo apresenta algumas limitações. Em primeiro lugar, o conjunto de *prompts*, embora diversificado, representa apenas uma parcela das tarefas atualmente utilizadas em *benchmarks* internacionais, o que restringe a generalização dos resultados. Além disso, a avaliação qualitativa depende de um único avaliador automático (Llama-3-70B), o que pode introduzir viés arquitetural, já que o avaliador pertence à mesma família de modelos que alguns dos avaliados. Outro ponto relevante é que o estudo não analisou métricas de factualidade, veracidade ou consistência lógica, aspectos essenciais para aplicações de alto impacto, como saúde, direito e educação. Por fim, a análise concentrou-se em métricas de tempo e qualidade textual, sem avaliar custo financeiro por requisição, consumo de GPU ou escalabilidade operacional em ambientes reais.

Para trabalhos futuros, recomenda-se expandir o conjunto de tarefas, incorporando categorias como raciocínio multimodal, inferência lógica, geração de código e interpretação de documentos extensos. Também seria pertinente incluir outros avaliadores automáticos — como Claude, GPT-4 ou modelos brasileiros — para reduzir possíveis vieses e comparar diferentes abordagens de julgamento automático. A inclusão de métricas de factualidade, detecção de alucinações e análise de consistência discursiva também enriqueceria substancialmente as conclusões. Além disso, estudos futuros podem explorar o impacto do custo por token, latência em lotes (batch inference) e desempenho em cenários de carga elevada, aproximando ainda mais a pesquisa do uso real em sistemas de produção.

Outro aspecto relevante observado ao longo do estudo é que um mesmo modelo de linguagem pode gerar respostas distintas para mesma pergunta, como por exemplo “criar uma história de terror”, isso ocorre pois o modelo dispõe de liberdade criativa, evitando gerar a mesma resposta quando o foco é criação de algo. Dessa forma, com respostas diferentes, a avaliação pode também apresentar notas diferentes, mas se espera que não seja uma variação significativa.

Em síntese, este trabalho demonstrou que métodos automatizados de *benchmarking*, aliados à avaliação baseada em LLMs, constituem uma estratégia promissora para pesquisadores e profissionais interessados em comparar modelos de linguagem. Os achados evidenciam tanto o potencial quanto as limitações dessas abordagens e abrem caminho para investigações

mais completas, diversificadas e alinhadas às necessidades crescentes da área de inteligência artificial aplicada.

Referências

ALUÍSIO, Sandra M.; PARDO, Thiago A. S. Desafios do Processamento da Língua Portuguesa e Avanços Recentes no Brasil. Revista de Informática Teórica e Aplicada, Porto Alegre, v. 17, n. 3, p. 45–68, 2010.

AVILA, Anderson; REAL, Luciana; FONSECA, Eduardo. Aplicações de Transformers no Português Brasileiro. Revista da Sociedade Brasileira de Computação, Porto Alegre, v. 28, n. 2, p. 55–73, 2020.

BARBOSA, João A.; FERNANDES, Mariana; CASTRO, Henrique. Modelos de Linguagem como Avaliadores Automáticos: Potencial e Limitações. In: CONGRESSO BRASILEIRO DE INTELIGÊNCIA ARTIFICIAL, 20., 2022, Fortaleza. Anais [...]. Fortaleza: SBC, 2022. p. 201–215.

FERNANDES, Cíntia; VIEIRA, Renata. Métodos Híbridos de Avaliação em PLN: Uma Revisão Sistemática. In: STIL – Symposium in Information and Language Technology, 13., 2022, Salvador. Anais [...]. Salvador: Sociedade Brasileira de Computação, 2022. p. 110–125.

REZENDE, Solange Oliveira (org.). Sistemas inteligentes: fundamentos e aplicações. 2. ed. Barueri: Manole, 2003.

SANTOS, Daniel; SOUZA, Eduardo. Desempenho de Modelos Transformers em Tarefas de Análise Textual em Português. Computação Brasil, São Paulo, n. 45, p. 22–29, 2021.

SILVA, Rafael; BERMEJO, Paulo. Uso de Modelos de Linguagem para Avaliação Automática de Textos. iSys – Revista Brasileira de Sistemas de Informação, Salvador, v. 16, n. 1, p. 8–30, 2023.

SOUZA, André; TORRES, Adriana. Democratização da Inteligência Artificial por Modelos Abertos. Revista de Sistemas e Computação, Recife, v. 11, n. 1, p. 1–15, 2021.