Análise Exploratória da Reprodução Parcial do CodeHelp com e sem Guardrails

Davi Laerte Nunes Sabino Nascimento

2025-06-21

1. Introdução e Contextualização dos Dados

Este estudo exploratório faz parte de uma de reprodução parcial do sistema CodeHelp, uma ferramenta desenvolvida para apoiar estudantes de programação 1 com uso de LLMs, usando como estratégia socrática mecanismos de contenção pedagógica, chamados de *guardrails*. Esses *guardrails* têm como objetivo:

- Evitar que o modelo forneça diretamente a solução completa e direta aos estudantes
- Estimular a reflexão e a compreensão dos erros cometidos pelos estudantes

Para avaliar a eficácia desses mecanismos, foi submetido um conjunto de perguntas simuladas, em nível introdutório, ao CodeHelp em dois modos operacionais: - Com *guardrails* habilitados - Com *guardrails* desabilitados (sem restrição explícita)

Cada pergunta foi respondida pelo CodeHelp em ambas as condições, ou seja, cada pergunta gerou duas respostas um com guardrails e outra sem, e as respostas foram posteriormente avaliadas por uma LLM (ChaptGPT) que atuou como avaliador pedagógico (Julgou as respostas). As avaliações foram armazenadas no arquivo CSV `responses_python_questions_evaluations.csv`.

Cada linha do arquivo representa uma resposta gerada pelo CodeHelp em uma das duas condições, juntamente com as métricas atribuídas pela LLM avaliadora, que são as seguintes:

- `avoids_giving_solution_directly`
- `promotes_reflection`
- `introductory_level_explanation`
- `consistency_with_question`
- `overall_score`

2. Descrição dos Campos do Dataset

O arquivo `responses_python_questions_evaluations.csv` contém os dados de avaliação das respostas geradas pelo CodeHelp, com e sem *guardrails*. A seguir estão os campos presentes no dataset:

- **original_idx**: identificador da pergunta original, usado para parear as respostas geradas nas duas condições.
- **guardrails**: valor booleano (`True` ou `False`) que indica se os *guardrails* estavam ativados na geração da resposta.
- **query_id**: identificador único da requisição feita ao CodeHelp. É sempre diferente, mesmo para a mesma pergunta, já que cada execução gera um novo ID.
- **issue**: descrição da dúvida enviada pelo aluno (a pergunta).

- **code**: trecho de código enviado pelo aluno junto com a dúvida (campo opcional).
- **error**: mensagem de erro, se fornecida pelo aluno (campo opcional).
- **response**: resposta gerada pelo CodeHelp.
- **avoids_giving_solution_directly**: nota de 0 a 10 que avalia se a resposta evita fornecer diretamente a solução.
- **promotes_reflection**: nota de 0 a 10 que avalia o quanto a resposta estimula a reflexão do aluno.
- **introductory_level_explanation**: nota de 0 a 10 que avalia a adequação da explicação ao nível introdutório.
- **consistency_with_question**: nota de 0 a 10 que avalia a consistência da resposta com a pergunta enviada.
- **overall_score**: nota de 0 a 10 representando a avaliação geral da resposta.
- **includes_full_code**: valor booleano (`True` ou `False`) que indica se a resposta inclui um código completo como solução.
- **comments**: comentário textual gerado pela LLM avaliadora com uma justificativa para as notas atribuídas, esse campo foi usado para validar o julgamento da LLM (feito por uma pessoa com conhecimento técnico).

3. Análise Inicial do Dataset

O dataset analisado contém um total de **200 respostas**, correspondentes a **100 perguntas originais** que foram processadas em dois cenários distintos: com e sem *guardrails*. Cada pergunta foi submetida ao CodeHelp nas duas condições, gerando duas respostas diferentes que foram posteriormente avaliadas por uma LLM.

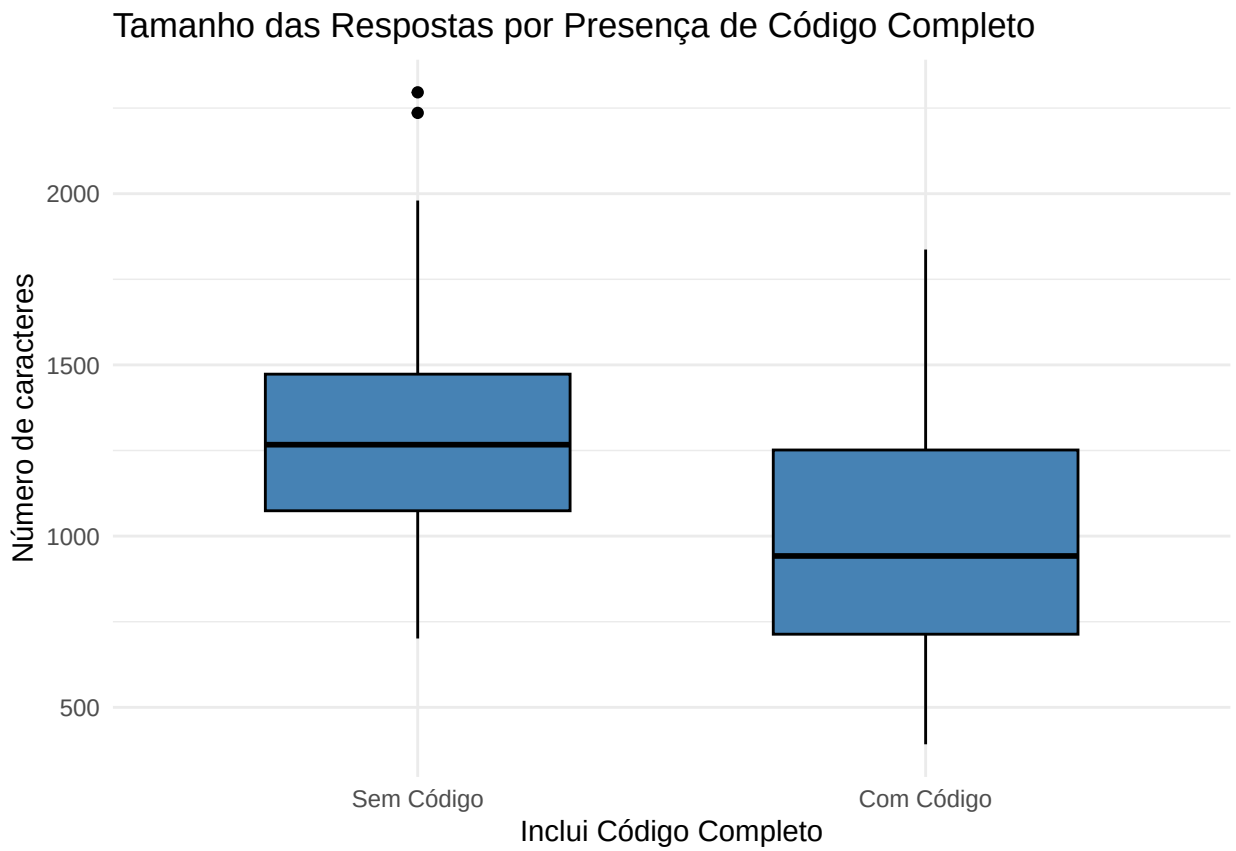
Nesta seção, observamos algumas características gerais do conjunto de dados, incluindo o tamanho médio das respostas geradas e a proporção de respostas que incluem um código completo como solução.

```
## Total responses: 200
```

```
## Average response length: 1157.94 characters
```

```
## Proportion of responses with full code: 45.5 %
```

Segue também a distribuição do número de caracteres na resposta pela presença de código completo na solução.



Olhando os dados gerados, podemos perceber que o número de caracteres tende a ser maior quando não há inclusão de código, muito provavelmente pelo fato de a resposta necessitar de mais palavras para a explicação, ou seja, a LLM precisa compensar com uma explicação mais detalhada em linguagem natural. Diferente de uma resposta com inclusão de código, que precisaria de menos palavras, já que o próprio código funciona como parte da explicação, sendo mais autoexplicativo.

4. Análise Geral das Métricas de Avaliação

Nesta seção, exploramos a distribuição geral das métricas numéricas atribuídas pela LLM às respostas geradas pelo CodeHelp. O objetivo é entender o comportamento agregado dessas avaliações, independentemente do uso de *guardrails*. Para isso, apresentamos estatísticas descritivas e gráficos de distribuição para cada uma das cinco métricas.

As métricas analisadas são:

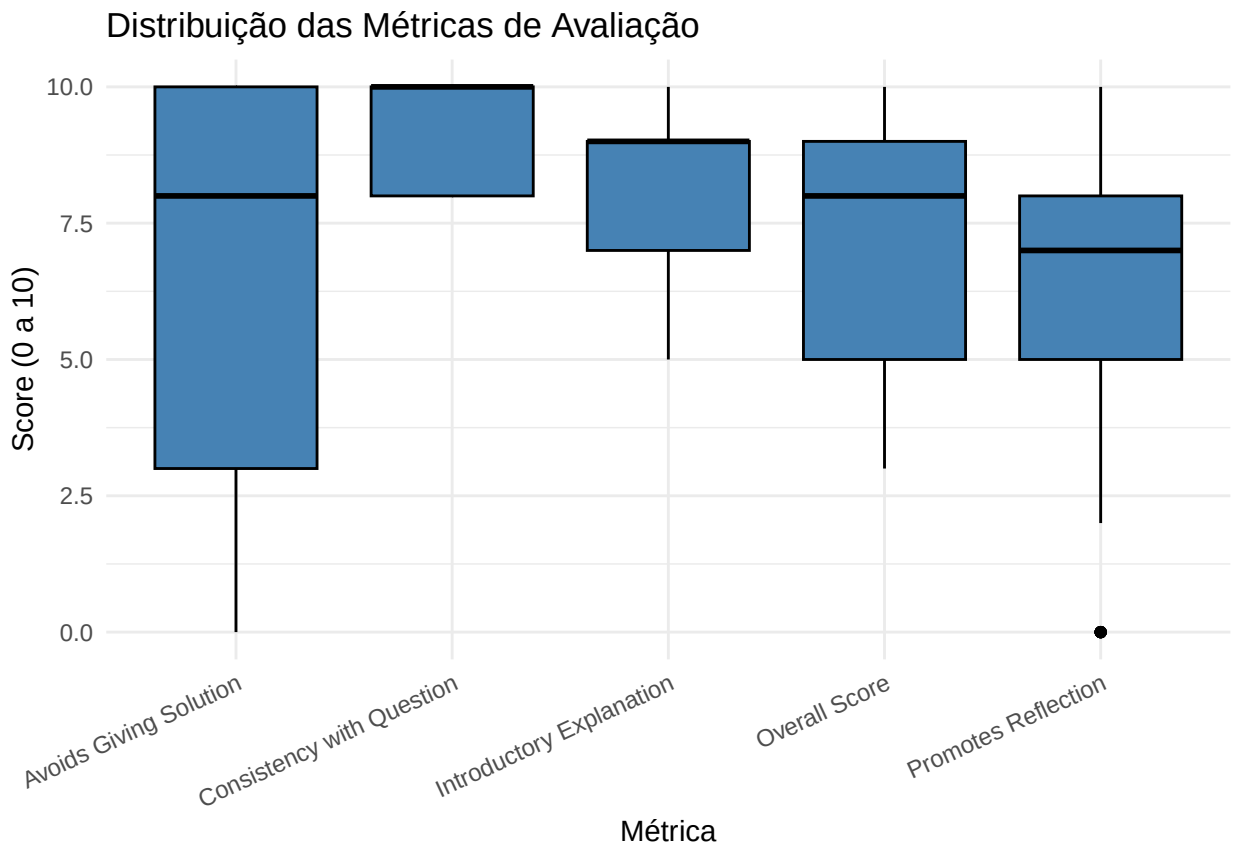
- `avoids_giving_solution_directly`
- `promotes_reflection`
- `introductory_level_explanation`
- `consistency_with_question`
- `overall_score`

Segue um resumo estatistico das métricas.

Table 1: Resumo estatístico das métricas de avaliação (0 a 10)

Métrica	Média	Desvio Padrão	Mediana	Q1	Q3	Mínimo	Máximo
Avoids Giving Solution	6.46	3.65	8	3	10	0	10
Promotes Reflection	6.54	2.52	7	5	8	0	10
Introductory Explanation	8.02	1.49	9	7	9	5	10
Consistency with Question	9.29	0.90	10	8	10	8	10
Overall Score	7.14	2.21	8	5	9	3	10

Também segue um boxplot com a distribuição das métricas.



Observando a distribuição das métricas, vemos que algumas apresentam valores majoritariamente mais altos, como é o caso de *consistency_with_question* e *introductory_explanation*, enquanto outras variam mais, em especial avoids_giving_solution, que apresenta maior desvio padrão e também maior diferença entre os quartis.

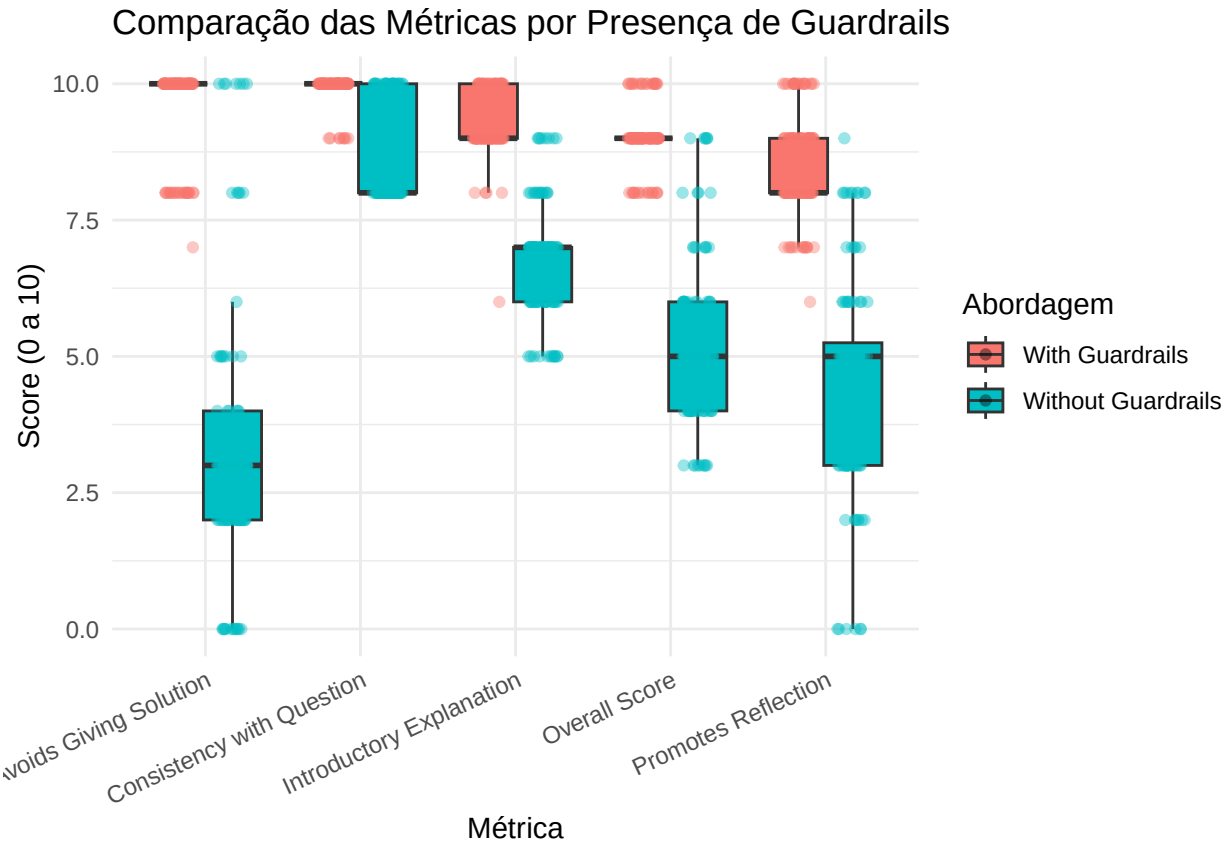
Uma possível explicação é que as métricas com valores mais altos avaliam aspectos como a consistência da resposta em relação à pergunta e sua adequação ao nível introdutório esperado. Essas características podem não ser afetadas diretamente pela presença ou ausência dos guardrails, já que a LLM tende a manter certo nível de coerência e adequação em ambos os cenários.

Já a métrica *avoids_giving_solution* é claramente mais sensível à presença dos guardrails, pois reflete diretamente a política de não fornecer respostas completas. Ela também pode variar por outros fatores, dependendo do tipo de pergunta feita.

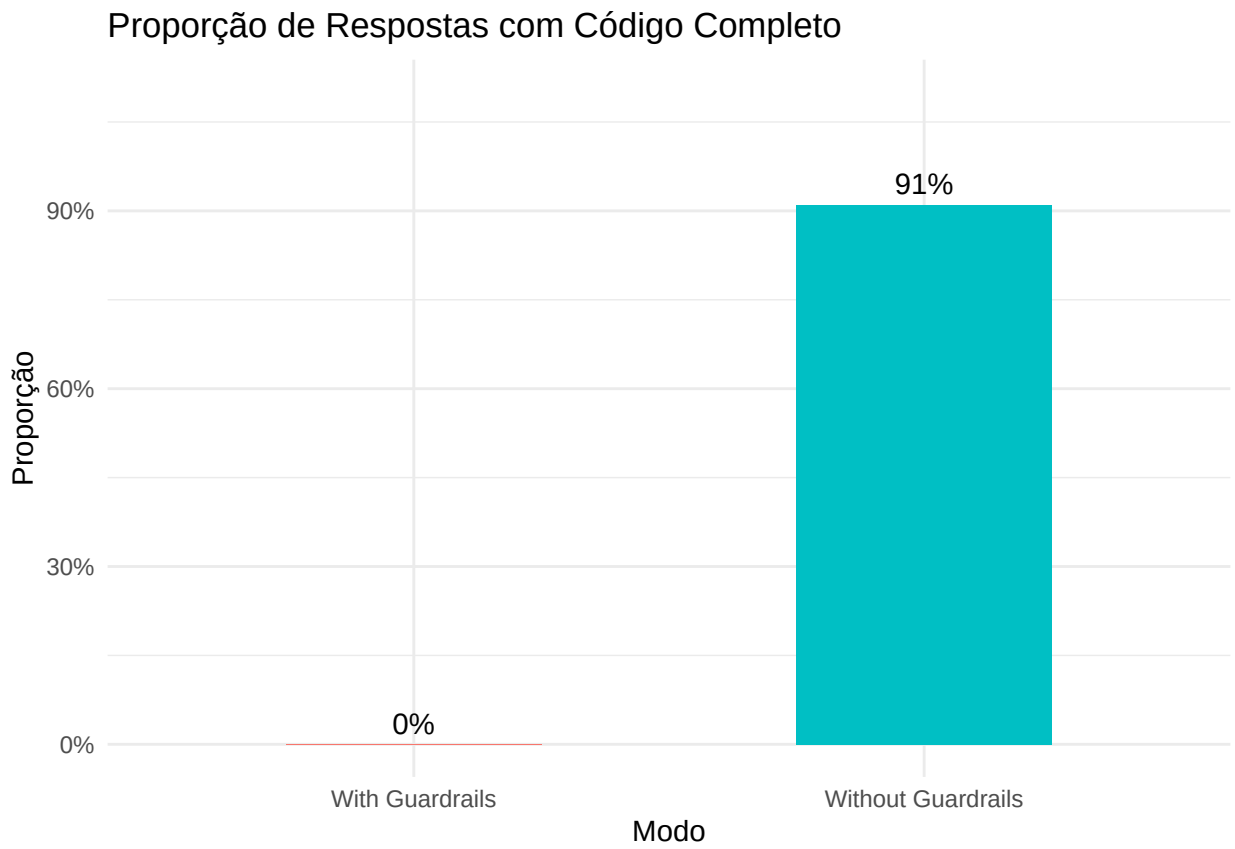
5. Comparação entre Respostas com e sem Guardrails

Nesta seção, comparamos as respostas geradas pelo CodeHelp com e sem o uso de *guardrails*. O objetivo é investigar se há diferenças perceptíveis nas avaliações feitas pela LLM e nas características gerais das respostas.

As comparações incluem: - Métricas avaliativas (ex: `overall_score`, `promotes_reflection`) - Tamanho das respostas - Frequência de inclusão de código completo



Aqui vamos analisar a proporção das métricas em relação a inclusão de código.



Também vamos gerar um resumo estatistico das métricas com e sem guardrails.

Table 2: Resumo estatístico das métricas de avaliação por tipo de execução

Abordagem	Métrica	Média	Desvio Padrão	Mediana	Q1	Q3	Mínimo	Máximo
With Guardrails	Avoids Giving Solution	9.57	0.84	10	10	10.00	7	10
Without Guardrails	Avoids Giving Solution	3.35	2.55	3	2	4.00	0	10
With Guardrails	Consistency with Question	9.92	0.27	10	10	10.00	9	10
Without Guardrails	Consistency with Question	8.66	0.87	8	8	10.00	8	10
With Guardrails	Introductory Explanation	9.22	0.61	9	9	10.00	6	10
Without Guardrails	Introductory Explanation	6.81	1.07	7	6	7.00	5	9
With Guardrails	Overall Score	9.03	0.58	9	9	9.00	8	10
Without Guardrails	Overall Score	5.25	1.51	5	4	6.00	3	9
With Guardrails	Promotes Reflection	8.51	0.96	8	8	9.00	6	10
Without Guardrails	Promotes Reflection	4.58	2.01	5	3	5.25	0	9

Analisando os dados gerados, percebemos que as métricas apresentam diferenças perceptíveis entre as duas abordagens — com e sem guardrails. As questões respondidas sem guardrails apresentam médias e medianas menores em todas as métricas, embora a magnitude dessa diferença varie: por exemplo, a métrica *consistency_with_question* apresenta uma diferença pequena, enquanto *avoids_giving_solution* mostra uma diferença bastante expressiva.

Esses resultados corroboram a hipótese do artigo do CodeHelp, indicando que as respostas geradas com guardrails são mais eficazes em promover reflexão e em evitar que a solução seja entregue diretamente ao estudante.

Outro ponto interessante é que *100% das respostas com guardrails não incluíram código completo*, enquanto cerca de *91% das respostas sem guardrails incluiram código completo*. Isso também reforça a hipótese de que as LLMs tendem a fornecer código por padrão, e que os guardrails atuam como um mecanismo de contenção para evitar esse comportamento.

Por fim, também observamos que as respostas sem guardrails apresentam maior variação nas métricas, tanto em termos de desvio padrão quanto nos gráficos de boxplot. Em contraste, as respostas com guardrails mostram uma variação menor, indicando uma maior consistência em seguir o padrão esperado. Isso sugere que impor limitações ao comportamento da LLM contribui para a geração de respostas mais uniformes e alinhadas com os objetivos pedagógicos.

6. Teste de Hipótese com Dados Pareados

Para avaliar se o uso de *guardrails* influencia significativamente a qualidade das respostas geradas, realizamos um teste de hipótese pareado utilizando o **Wilcoxon**. Este teste não paramétrico é apropriado para comparar distribuições quando os dados não seguem uma distribuição normal, especialmente em pares relacionados, que é o caso aqui, em que cada pergunta (`original_idx`) foi avaliada nas duas condições.

O teste foi aplicado à métrica `overall_score`, que representa a avaliação geral da LLM para cada resposta, essa métrica tem como base as outras quatro métricas:

- `avoids_giving_solution_directly`
- `promotes_reflection`
- `introductory_level_explanation`
- `consistency_with_question`

Table 3: Resumo dos Testes Estatísticos Aplicados

Test	p-value	Alternative Hypothesis
Shapiro-Wilk Normality Test	2.96e-04	Data is not normally distributed
Wilcoxon	2.47e-17	true location shift is not equal to 0

Analisando o resultado do teste de *Shapiro-Wilk*, entendemos que os dados não seguem uma distribuição normal, pois rejeitamos a hipótese de normalidade com base no p-valor inferior a 0,05. Sendo assim, o uso do teste de *Wilcoxon* foi apropriado, já que os dados não são normalmente distribuídos e tratam-se de dados pareado,ou seja, uma mesma pergunta gerando duas respostas distintas, dependendo da abordagem (com ou sem guardrails).

Observando o resultado do teste de *Wilcoxon* aplicado à métrica *overall_score*, podemos rejeitar a hipótese nula de que a diferença entre os dois grupos é igual a zero, já que o p-valor é significativamente menor que 0,05.

Com isso, concluímos que, com base nos dados analisados, existe evidência estatística de que a abordagem com guardrails produz respostas diferentes, e superiores, em relação às métricas pedagógicas utilizadas.

7. Considerações Finais

A análise realizada demonstrou diferenças consistentes nas respostas geradas pelo CodeHelp com e sem o uso de *guardrails*. De forma geral, as respostas com *guardrails* apresentaram scores mais elevados nas métricas avaliativas, especialmente em critérios como “evitar fornecer a solução completa” e “promover reflexão”.

A distribuição dos scores revelou que as respostas com *guardrails* tendem a ser mais homogêneas, com valores concentrados em faixas superiores (9 e 10), enquanto as respostas sem *guardrails* apresentaram maior variabilidade.

O teste de hipótese pareado aplicado à métrica `overall_score`, que indica o score geral das métricas pedagógicas utilizadas, indicou que a diferença entre as abordagens é estatisticamente significativa ($p < 0.05$), reforçando a hipótese de que os *guardrails* influenciam positivamente na qualidade pedagógica percebida das respostas.

Esses resultados vão de encontro ao que foi proposto no artigo original do CodeHelp, reforçando a ideia de que o uso de guardrails pode melhorar a qualidade pedagógica das respostas.A forma como essa análise foi conduzida pode servir como ponto de partida para testes com outros modelos ou até outros contextos de uso, ajudando a entender melhor como diferentes estratégias afetam o tipo de resposta gerada.