

From Calculation to Adjudication: Examining LLM Judges on Mathematical Reasoning Tasks

Andreas Stephan, Dawei Zhu, Matthias Aßenmacher,
Xiaoyu Shen, Benjamin Roth

NLP @ University of Vienna



ACL 2025

The No. 1 scientific NLP event is in Vienna next year

Background & Motivation

LLMs: Where are we coming??

1. Until ~2017, we trained models from scratch, so we had training and evaluation data for a specific problem (using Random Forest Classifiers, RNNs, CNNs, LSTMs, ...).
2. Until ~2021, models are pre-trained on a vast amount of data, and then fine-tuned on a smaller subset (Transformers, BERT, ...)
3. GPT-3 (2021), ChatGPT (2022): Models follow general instructions (*Zero-Shot Learning*).

Why is this a big deal?

- Models are able to perform more complex tasks
 - Obtaining data annotations is
 - Expensive: Human experts, annotation infrastructure, etc.
 - Complicated: Data Protection Laws, Define Proper annotation scheme
- **BUT!!!!** In order to evaluate models, annotation is still crucial.

The big question in the room:

Can we use LLMs to automatically evaluate candidate models?

LLM-as-a-judge

- Zheng et al. intrude the LLM-as-a-judge paradigm.
- A LLM judge evaluates the performance of other LLMs.
- Example: GPT-4 judging a 2-turn conversation

Question: If the FED buys bonds in the secondary market

(A) the money supply will increase. (B) the money supply will decrease. (C) the money supply will not be affected. (D) the discount rate would be affected. Output the answer only.

Assistant A:

The Federal Reserve buys bonds in the secondary market to increase the money supply.

Assistant B:

(A) the money supply will increase.

Follow-up Question: How does it affect my daily life? Give 3 examples.

Assistant A:

The Federal Reserve buys bonds in the secondary market to increase the money supply. This affects your daily life by:

1. Increasing the money supply.
2. Increasing the money supply and lowering the discount rate.
3. Increasing the money supply and lowering the discount rate and increasing the money supply.

Assistant B:

When the Federal Reserve buys bonds in the secondary market, it can have several effects on the economy that can impact your daily life. Here are three examples:

1. Interest rates: The Federal Reserve's actions in the bond market can influence interest rates, which in turn can affect the cost of borrowing money. For example, if the Federal Reserve buys bonds and drives up demand for them, interest rates on things like mortgages, car loans, and credit cards may go up, making it more expensive to borrow money.
2. Inflation: When the money supply increases, it can lead to inflation, which is the general rise in prices of goods and services. This can impact your daily life by making it more expensive to buy things like groceries, gas, and other necessities.
3. Employment: The Federal Reserve's actions in the bond market can also have an impact on employment. For example, if the Federal Reserve's actions lead to economic growth and increased demand for goods and services, it can create more job opportunities and boost employment.

GPT-4 Judgment:

Assistant A provided an incorrect response to the user's question about how the Federal Reserve buying bonds in the secondary market affects daily life. The answer given is repetitive and lacks clear examples of how the action impacts daily life.

On the other hand, **Assistant B** provided a relevant and accurate response to the user's question about the Federal Reserve buying bonds. The answer includes three clear examples of how the action impacts daily life, such as interest rates, inflation, and employment.

Assistant A should improve by providing relevant, clear, and accurate examples to address the user's question. **Assistant B's** response is more helpful, relevant, and detailed.

Overview: Evaluation of LLMs

- Academic benchmarks using human-annotated well-defined input / output pairs
 - Sentiment analysis
 - Machine Translation
 - MMLU [1]: Multiple choice questions from abstract algebra to law
 -
- ChatBotArena
 - Compare output of two LLMs and humans decide which answer is better
 - ELO-System to rank quality of LLMs
 - <https://lmarena.ai/>
- Automatic Evaluation
 - Using Large Language models
 - **FOCUS of this talk**

Our Setup

Given a textual mathematical question:

- Two candidate LLMs provide a candidate answer
 - They contain chain-of-thought (CoT) reasoning and
 - A number as final answer
- A Judge LLM reasons using CoT which answer is better, and finally **HAS** to decide which answer is correct

Question

Mr. Ruther sold $3/5$ of his land and had 12.8 hectares left. How much land did he have at first?

LLM M_A

Mr. Ruther was left with $1 - 3/5 = \ll 1 - 3/5 = 0.4 \gg 0.4$ or $2/5$ of his land his land which is equal to 12.8 hectares. So...**###32.**



LLM M_B

Let x be the original land he had. $3/5x = 12.8$, $x = 12.8 / 3/5 = \ll 12.8 / 3/5 = 20.8 \gg 20.8$.**#### 20.8.**



Judge LLM

Answer A is correct. In Answer B, the equation is set up incorrectly. If 12.8 hectares is the amount of land left after selling $3/5$ of the land, then 12.8 hectares represents $2/5$ of... **{"answer": "A"}**



CoT text

Final answer

Why mathematical reasoning tasks?

- Mathematical reasoning tasks are challenging.
 - (By observing task performance of candidate LLMs)
- We have access to correctness of an answer.
 - The final answer (a number) is either correct or incorrect.
 - In machine translation (or other text-to-text tasks) it is much more subjective which answer is better.

Experimental Setup

Notation

D	→	Dataset
$M_A, M_B \in \mathbf{M}$	→	Candidate models
$M_J \in M$	→	Judge model
A, B, J	→	Candidate answers (number) by candidate models or judge

$A = T, A = F, A = B$	→	Model is correct, incorrect, or models agree
$\Delta_J = T, \Delta_J = M_A$	→	Judgement is correct, Judgement chooses model M_A

$P(\Delta_J = T A, B, D)$	→	Performance on a dataset D when we always take the answer chosen by the judge
---------------------------	---	---

Judgement Cases

- The judgement performances that a judge chooses a correct answer can be divided into 4 cases
- Both candidates correct
 $P(A = T, B = T|D) = 1$
- Both candidates correct
 $P(A = F, B = F|D) = 0$
- Exactly one answer correct
⇒ Interesting cases

$$\begin{aligned} P(\Delta_J = T|A, B, D) &= \sum_{(X,Y) \in C} P(\Delta_J = T|A = X, B = Y, D)P(A = X, B = Y|D) \\ &= P(A = T, B = T|D) \\ &\quad + P(\Delta_J = T|A = T, B = F, D)P(A = T, B = F|D) \\ &\quad + P(\Delta_J = T|A = F, B = T, D)P(A = F, B = T|D) \end{aligned}$$

Datasets

- **AQUA-RAT** (Ling et al., 2017): A dataset to test the quantitative reasoning ability of LLMs using multiple choice questions
- **GSM8K** (Cobbe et al., 2021): Grade school math word problems. Answers are free-form numbers.
- **MATH** (Hendrycks et al., 2021): Challenging mathematics competition problems, answers are also free-form numbers.

Models

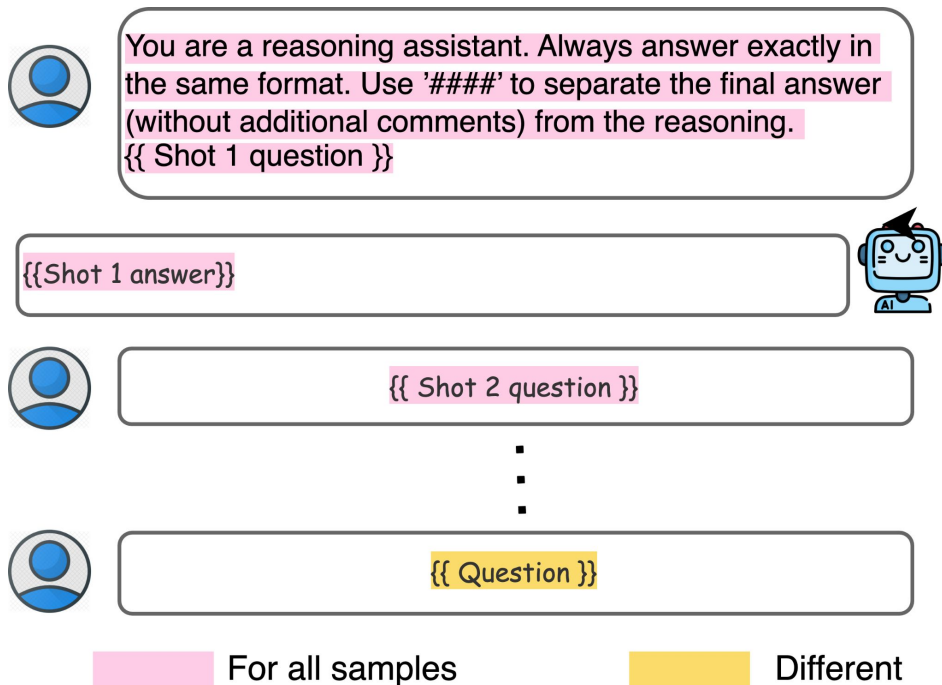
“Large” Open-Weight Models:

- Qwen 2 72B (Yang et al., 2024)
- Llama 3 70B (AI@Meta, 2024)
- Yi 1.5 34B (Young et al., 2024)
- Mixtral 8x7B (Jiang et al., 2024)

“Small” Open-Weight models:

- Llama 3 8B (AI@Meta, 2024)
- Gemma 1.1 7B (Gemma Team et al., 2024)
- Mistral 7B v0.3 (Jiang et al., 2023)
- Mistral 7B v0.1 (Jiang et al., 2023)

Candidate answer generation



We call generated answers by the LLMs **candidate answers**.

For each data point (or question) of all datasets, we generate 2 candidate answers by each model

- using a temperature of 0.9
- as we want to judge answers of the same model

Judgement Generation



Question:

{{question}}

Answer A:

{{answer A}}

Answer B:

{{answer B}}

Compare both answers in detail and choose the answer which correctly answers the question.
Conclude with a JSON in Markdown format indicating your choice between answer A or B:

```
```json
{
 "answer": "B" or "A"
}
```
```

For all 36 model pairs*, and all datasets, we

- Take the first generation of the models, or both candidate generations
- Judges have shown position bias, i.e. favoring a certain ordering of answers, thus we evaluate both orderings

using temperature 0.

*36 =(8+7+...+1) is the amount of unique pairs

Task Performance

| | AQUA-RAT | GSM8K | MATH |
|-----------------|----------|-------|-------|
| Qwen 2 72B | 76.38 | 92.04 | 51.19 |
| Llama 3 70B | 73.62 | 91.05 | 34.37 |
| Yi 1.5 34B | 64.96 | 78.47 | 27.04 |
| Mixtral 8x7B | 47.24 | 61.18 | 13.79 |
| Llama 3 8B | 51.18 | 73.01 | 15.04 |
| Gemma 1.1 7B | 42.91 | 50.72 | 12.60 |
| Mistral 7B v0.3 | 38.19 | 42.76 | 6.13 |
| Mistral 7B v0.1 | 21.65 | 26.08 | 3.10 |

Difficulty:

MATH > AQUA-RAT > GSM8k

Task Performance:

Large difference between
best (*Qwen 2 72B*) and
worst (*Mistral 7B v0.1*) model.

Example from GSM8K

James hires a horse-drawn carriage from 5 PM to 9 PM. He gets 1 hour free. The first paid hour is \$15 and each hour after that is twice the cost. How much did he pay?

Answer A:

He rented it for $9-5=4$ hours

He got $4-1=3$ hours paid

The second hour cost $15 \times 2 = 30$

The third hour cost $30 \times 2 = 60$

So in total, he paid $15 + 30 + 60 = 105$

105

Answer B:

He hired the carriage from 5 PM to 9 PM which is 4 hours ($9-5 = 4$).

The first hour is free, which means he paid for the remaining 3 hours.

The first paid hour costs \$15, and the remaining hours cost twice as much: $2 \times 15 \times 2 = 60$.

Therefore, he paid a total of $0 + 15 + 60 = 75$.

75

Answer A is correct. The first hour is free, so James paid for 3 hours. The second hour costs twice the first paid hour, which is $15 \times 2 = 30$. The third hour costs twice the second hour, which is $30 \times 2 = 60$. Therefore, the total cost is \$15 (first paid hour) + \$30 (second hour) + \$60 (third hour) = \$105.

Answer B is incorrect because it does not correctly calculate the cost of the third hour. It multiplies the cost of the first paid hour by 2 and then by 2 again, which is not how the problem is described.

Experiments

Judge performance per dataset

We average judge performance on all judged model pairs, and analyze performance on three subsets

$P(\Delta_J = T | A, B, D)$ Performance for all pairs (including where both answers are correct / incorrect).

$P(\Delta_J = T | A \neq B, D)$ Performance where both candidate answers give a different result.

$P(\Delta_J = T | \{A, B\} = \{T, F\}, D)$ Performance where exactly one candidate answer is correct, and one incorrect.

Judge performance per dataset

| | | Qwen 2 72B | Llama 3 70B | Yi 1.5 34B | Mixtral 8x7B | Llama 3 8B | Gemma 1.1 7B | Mistral 7B v0.3 | Mistral 7B v0.1 |
|--|----------|--------------|--------------|--------------|--------------|------------|--------------|-----------------|-----------------|
| (1) $P(\Delta_J = T A, B, D)$ | AQUA_RAT | 66.05 | 55.98 | <u>62.2</u> | 57.08 | 51.59 | 53.52 | 54.96 | 51.41 |
| | GSM8K | 77.06 | 72.41 | <u>72.64</u> | 68.81 | 65.88 | 65.47 | 68.57 | 63.28 |
| | MATH | 29.66 | 24.64 | <u>26.69</u> | 23.60 | 22.35 | 21.66 | 22.68 | 19.91 |
| (2) $P(\Delta_J = T A \neq B, D)$ | AQUA_RAT | 53.79 | 45.04 | <u>49.73</u> | 44.47 | 36.56 | 40.48 | 38.26 | 36.43 |
| | GSM8K | 63.16 | <u>59.33</u> | 56.01 | 46.68 | 41.29 | 38.71 | 42.59 | 39.72 |
| | MATH | 25.04 | 22.14 | <u>22.22</u> | 18.29 | 17.04 | 16.00 | 16.88 | 15.22 |
| (3) $P(\Delta_J = T \{A, B\} = \{T, F\}, D)$ | AQUA_RAT | 73.13 | 64.63 | <u>68.66</u> | 63.26 | 52.74 | 58.04 | 54.54 | 52.45 |
| | GSM8K | 85.65 | <u>81.32</u> | 76.84 | 64.97 | 57.87 | 54.16 | 59.58 | 55.48 |
| | MATH | 80.48 | <u>73.01</u> | 71.37 | 61.03 | 58.04 | 54.70 | 56.27 | 50.91 |

- Qwen 2 72B is best performing judge
- Large models outperform smaller models
- Small models hover around random performance, when one correct / incorrect answer is shown

⇒ From now on, we focus on large models as judges

Performance per model pair

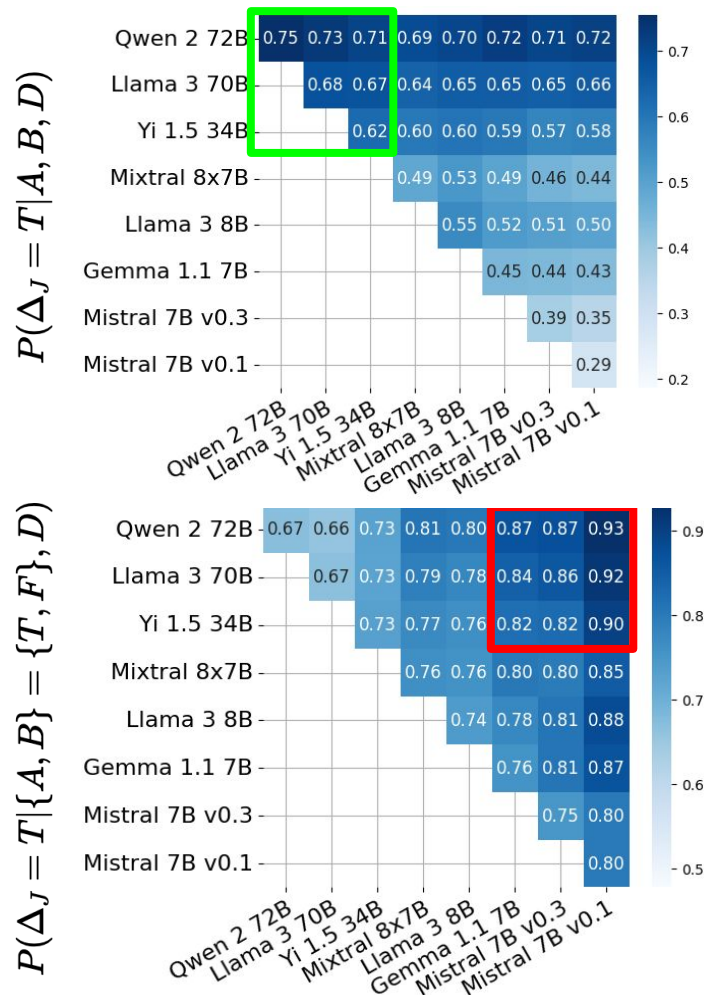
Setup: We average judge performance of Qwen 2 72B over all datasets.

Results (top; arbitrary answer pair)

- Best task performance if both candidate models are best.

Results (bottom; 1 correct, 1 incorrect answer):

- Best performance if there is one good / one bad model.
- For worse models it's easier to judge which answer is correct. (diagonal)



Can we use judges to improve task performance?

Setup

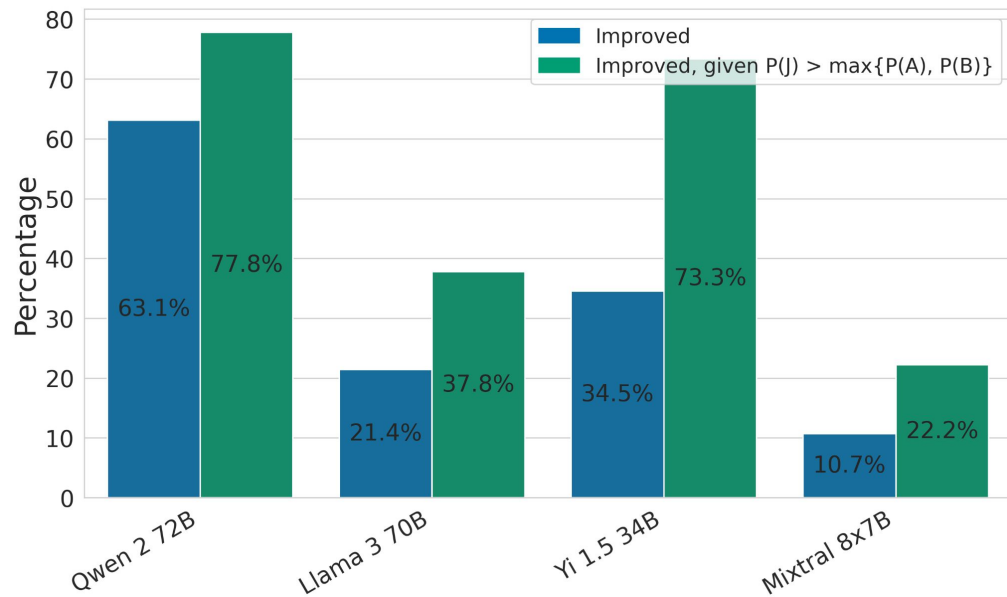
For a dataset, and a combination of two models M_A, M_B

- 1) A judge makes judgement for each instance of the dataset
- 2) We take the chosen answer for performance evaluation

We test

- On which amount (%) of model pairs does a judge improve performance?
- On which amount (%) of model pairs does a judge improve performance where both candidates have worse performance than the judge?

Can we use judges to improve task performance?



- Only Qwen consistently improves performance.
- Qwen 2 and Yi improve performance if judging worse models.

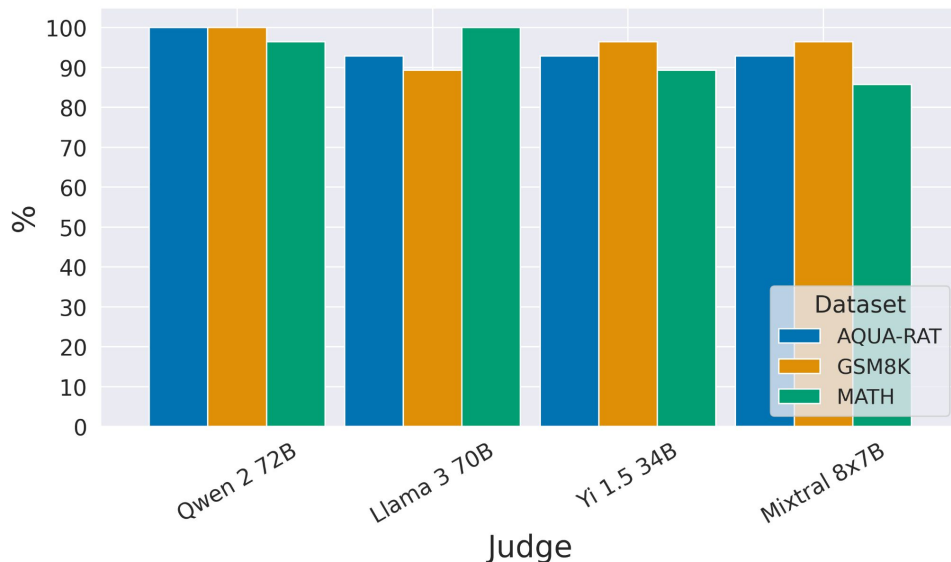
Do judges prefer better models (w.r.t. task performance)?

Setup:

- We analyze for how many pairs does a judge choose the model of higher task performance.
- By counting, on how many samples the judge chooses the stronger model.

Results:

- Almost always chooses correctly.
- LLMs judges seem to be able to decide which model is better.



Relationship between task performance and judgement performance

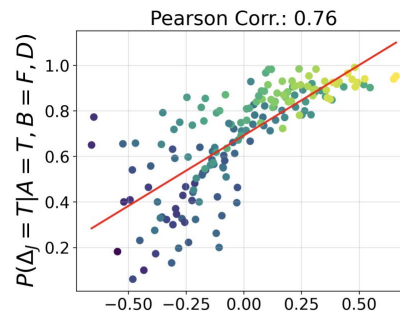
| Judge | model A | model B | dataset | X | Y | P(A D) | P(B D) | $P(\Delta_J = T A = X, B = Y, D)$ | $P(A=X, B=Y D)$ |
|-------------|--------------|-----------------|----------|-------|-------|--------|--------|-------------------------------------|-------------------|
| Qwen 2 72B | Qwen 2 72B | Mistral 7B v0.1 | MATH | True | False | 51.2 | 3.1 | 99.1 | 1217 |
| Qwen 2 72B | Yi 1.5 34B | Mistral 7B v0.1 | MATH | True | False | 27.0 | 3.1 | 98.4 | 633 |
| Qwen 2 72B | Llama 3 70B | Mistral 7B v0.1 | MATH | True | False | 34.4 | 3.1 | 98.3 | 838 |
| Yi 1.5 34B | Mixtral 8x7B | Gemma 1.1 7B | AQUA-RAT | True | False | 47.2 | 42.9 | 76.0 | 96 |
| Yi 1.5 34B | Qwen 2 72B | Mistral 7B v0.1 | MATH | False | True | 51.2 | 3.1 | 6.5 | 31 |
| Llama 3 70B | Owen 2 72B | Mistral 7B v0.1 | MATH | False | True | 51.2 | 3.1 | 6.2 | 32 |
| Qwen 2 72B | Qwen 2 72B | Mistral 7B v0.1 | MATH | False | True | 51.2 | 3.1 | 6.1 | 33 |

- LLMs tend to choose answers of better candidate models even if they are wrong.

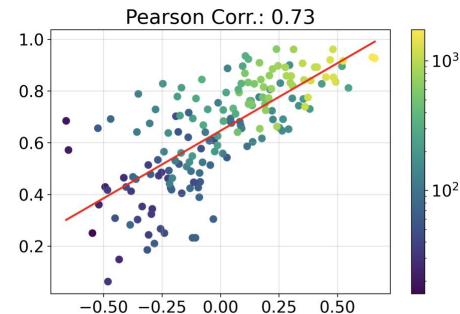
Relationship between task performance and judgement performance

Setup

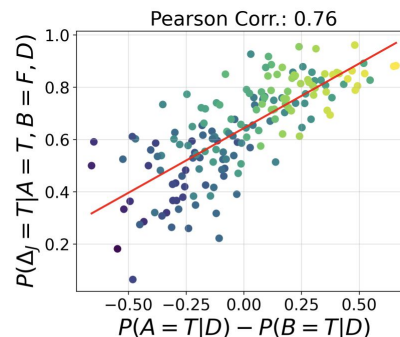
- X-axis: The difference in task performance between the candidate models
- Y-axis: Judgement performance on subset of the dataset where one model is correct, one model incorrect
- Point: represents one subset of a dataset subset
- Color: size of subset (absolute)



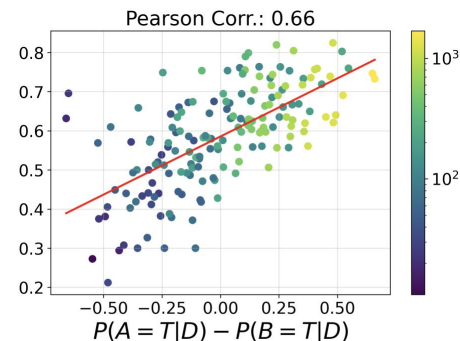
(a) Qwen2 72B



(b) LLama2 70B



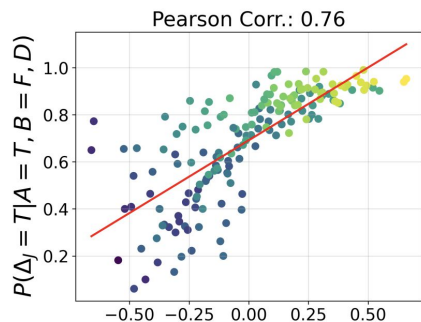
(c) Yi 1.5 43B



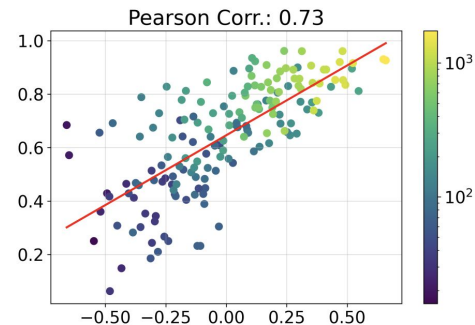
(d) Mixtral 8x7B

Relationship between task performance and judgement performance

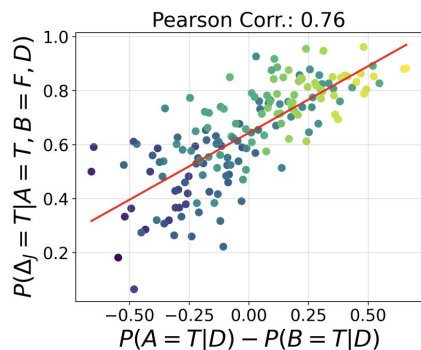
- We observe a strong correlation between performance difference and judgement performance
- LLMs tend to choose answers of better candidate models even if they are wrong.
- Usually not observed because it happens rarely that a better model is incorrect and worse correct.



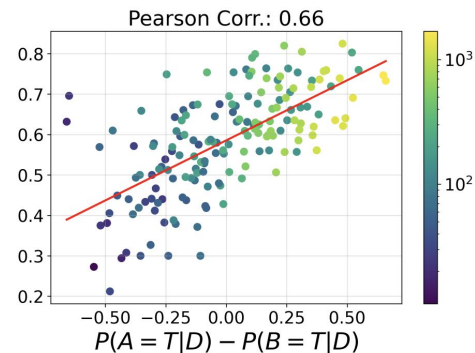
(a) Qwen2 72B



(b) LLama2 70B



(c) Yi 1.5 43B



(d) Mixtral 8x7B

Can summary statistics explain judgement performance?

We observed that there is a relationship between judgement performance and task performance.

→ **Question:** How much is judgement performance impacted by the following summary statistics?

(1) Using model task performances

(2) Using judgement counts

| | | $P(\Delta_J = T \mid \text{ })$ | | |
|--------------------------|---|---------------------------------|------------|-----------------------|
| ↓ Features \ Condition → | | A, B | $A \neq B$ | $\{T, F\} = \{A, B\}$ |
| (1) | $P(J), P(A), P(B)$ | 97.50 | 90.20 | 59.20 |
| (2) | $P(A = B),$
$P(J = A \mid A \neq B)$ | 76.00 | 54.90 | 49.90 |

Can summary statistics explain judgement performance?

Setup

- Calculate Coefficients of Determination (R^2 , higher is better) for linear regression.
- “How much variance in the judgement performance is explained by the features?”

Results

- Final task performance is almost completely explained by individual model performances.
- Case (2) still exhibits rather high R^2 value, indicating that the features explain 50% of the variance.

| | | $P(\Delta_J = T \mid \text{ })$ | | |
|--------------------------|---|---------------------------------|------------|-----------------------|
| ↓ Features \ Condition → | | A, B | $A \neq B$ | $\{T, F\} = \{A, B\}$ |
| (1) | $P(J), P(A), P(B)$ | 97.50 | 90.20 | 59.20 |
| (2) | $P(A = B),$
$P(J = A \mid A \neq B)$ | 76.00 | 54.90 | 49.90 |

Style hypothesis

LLM Judges do not only focus on the pure reasoning or logic in the answers but also base their judgements on the way a candidate LLM writes and presents the solution.

We call this writing style.

Can we predict individual judgements?

Idea: If a model bases its decision on writing styles, can we predict it using less complex models (which do not exhibit strong reasoning capabilities)?

Setup: Predict judgement using

- (1) TF-IDF features (weighted word counts) + Random Forest (RF)
 - (a) Model only has access to which vocabulary is used, and how often
 - (b) Encode both answers using TF+IDF and concatenate features
- (2) RoBERTa (BERT-style model)
 - (a) Model has access to sentence structure etc., but lacks complex reasoning ability (chain of thought prompting) modern LLMs exhibit
 - (b) Show same text as original LLM judge

Can we predict individual judgements?

Setup (2):

Split all judgement (across all candidate pairs, datasets) into train, dev, test split.

| ↓ Model \ Judge → | Qwen 2 72B | Llama 3 70B | Yi 1.5 34B | Mixtral 8x7B |
|-------------------|------------|-------------|------------|--------------|
| (1) TF-IDF + RF | 60.78 | 61.37 | 60.77 | 58.69 |
| (2) RoBERTa | 68.14 | 66.49 | 67.03 | 63.91 |

Results:

(1) Achieves performance around 60% (50% random choice)

=> specific phrases give some influence.

(2) Using contextual information RoBERTa achieves almost 70%

=> Indication that there is influence of reasoning (subset of 30%) and simpler indicators, a model such as RoBERTa can use.

Summary

- 1) We conduct an in-depth performance analysis of LLM judges for mathematical reasoning tasks.
- 2) Our analysis reveals a correlation between the judgment performance and candidate task performance, providing a novel insight into the behaviour of LLM judges, namely that they tend to favor stronger models.
- 3) We show that statistics such as task performance or agreement of candidate models are indicative of judgment performance.

Thank you!!

Questions? Ideas?