

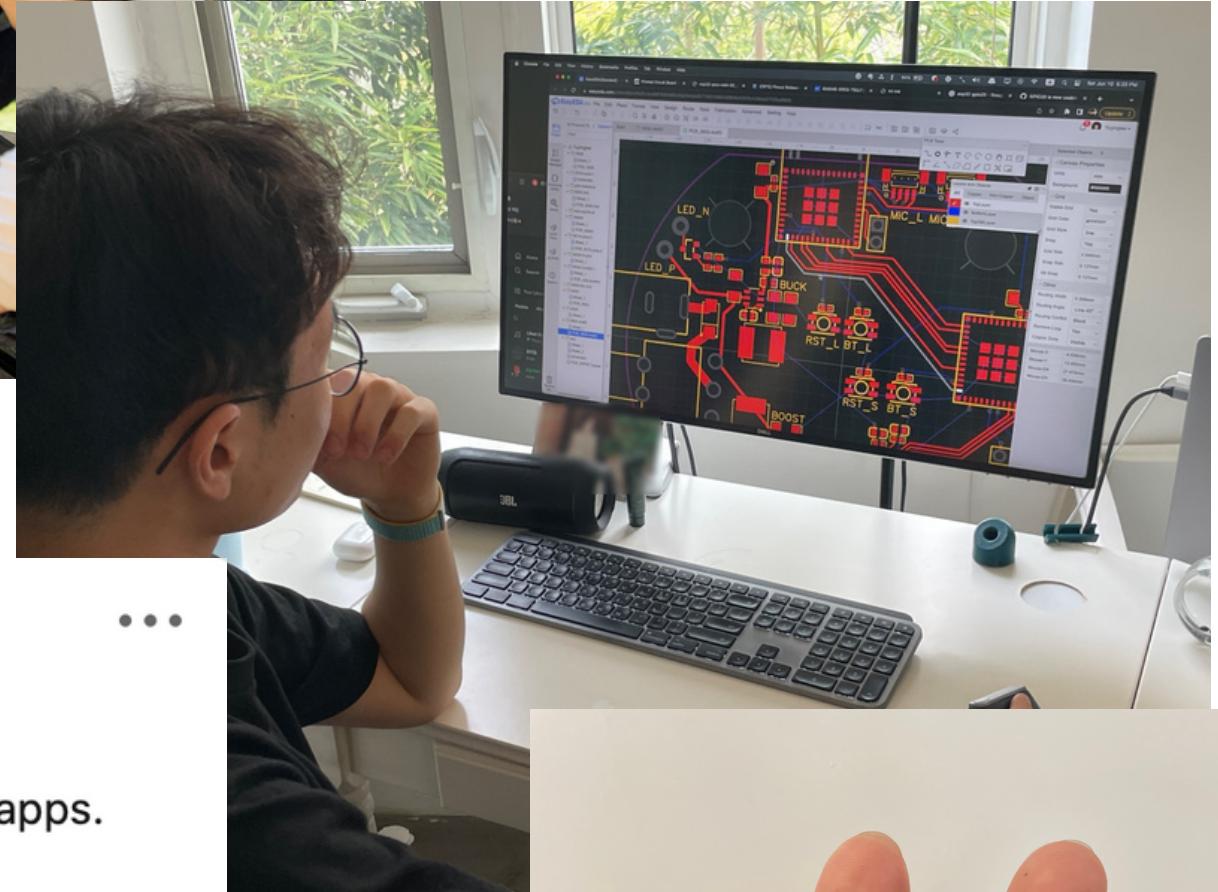
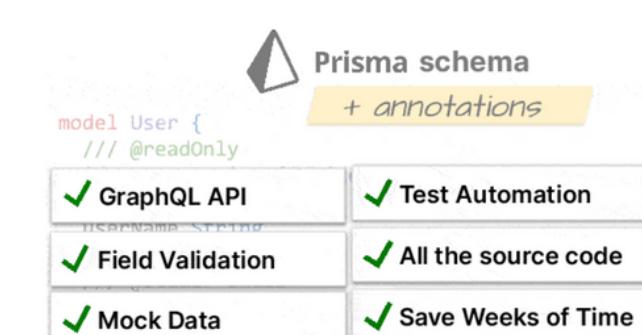
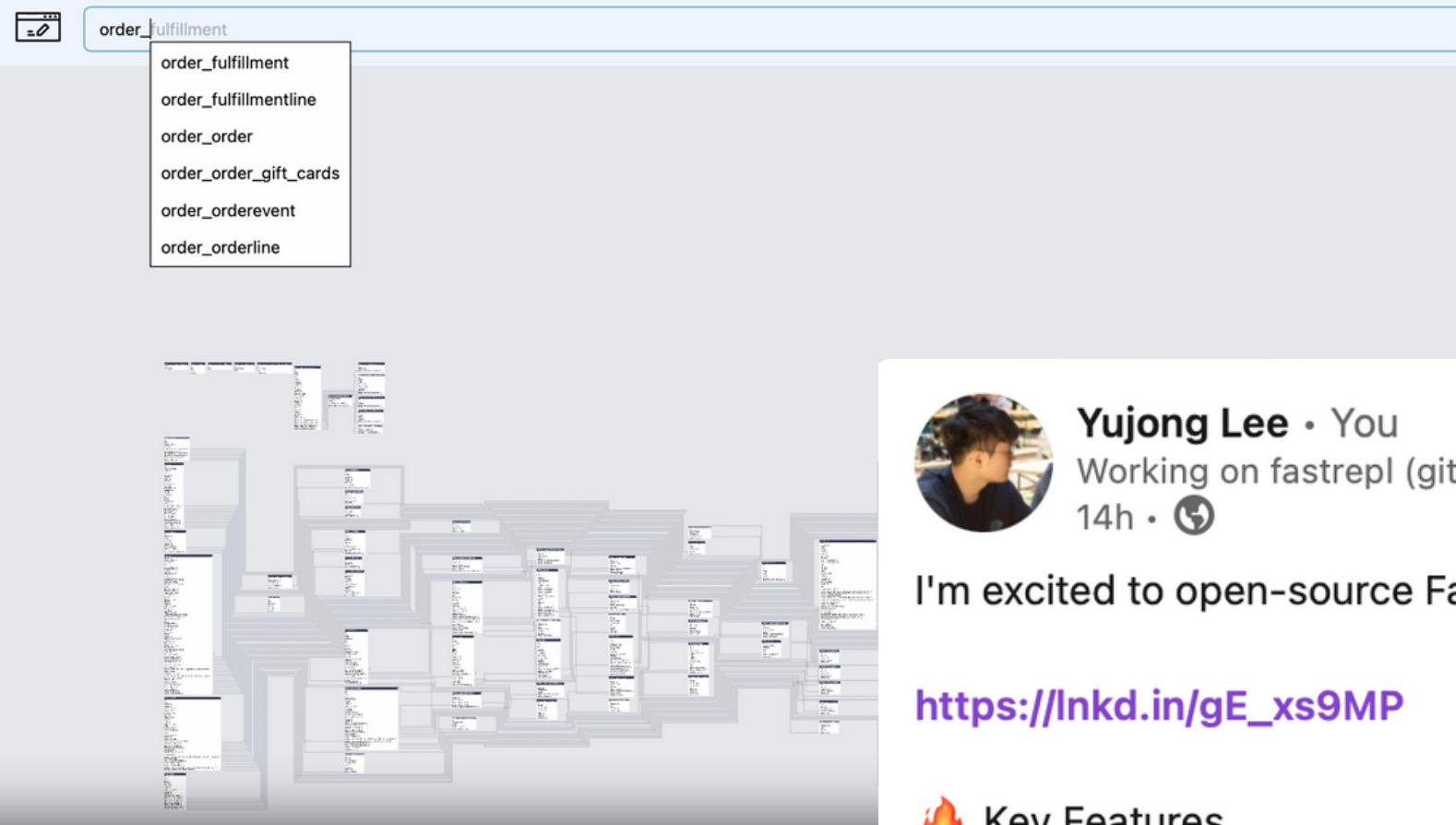
# **LLM In Production: Evaluation**

- 이 유종

The  
**Simplest,**  
**Fastest,**  
**most Flexible,**  
way to create & maintain  
your GraphQL APIs

Get started    Learn more

Try a no-install demo on Repl.it



**Yujong Lee** • You

Working on fastrepl ([github.com/fastrepl/fastrepl](https://github.com/fastrepl/fastrepl))  
14h ·

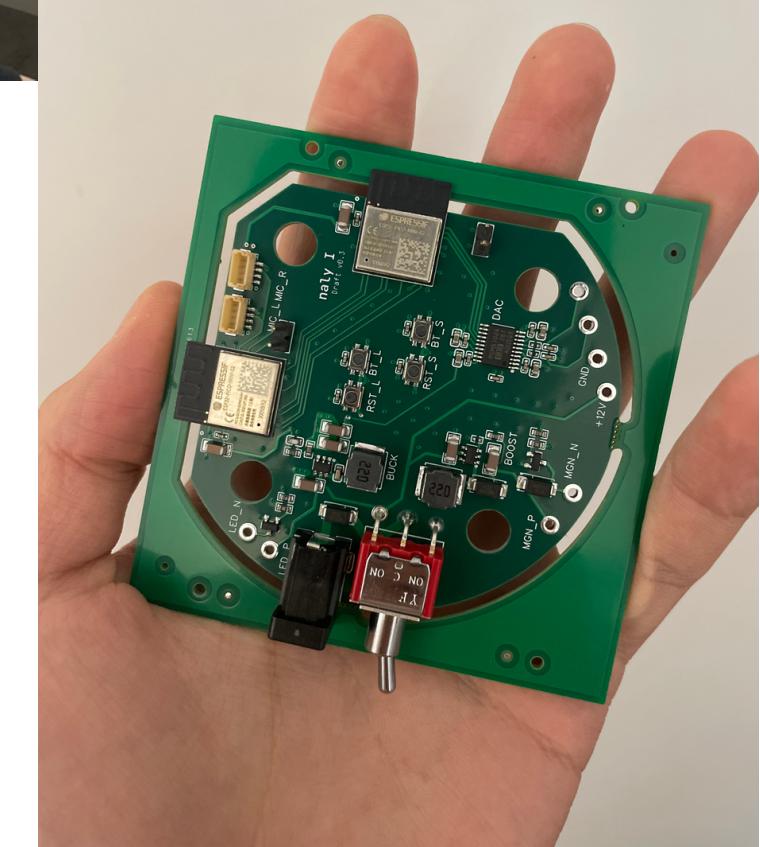
I'm excited to open-source FastREPL, fast run-eval-polish loop for LLM apps.

[https://lnkd.in/gE\\_xs9MP](https://lnkd.in/gE_xs9MP)

#### 🔥 Key Features

- Can use nearly any provider ([OpenAI](#), [Anthropic](#), [Cohere](#), [Replicate...](#)) as an evaluator, with automatic backoff/retries and fallback.
- Built-in debiasing strategies (<https://lnkd.in/gXYyhmme>).
- Evaluation result analysis and warnings. (e.g Cohen's Kappa)
- And more to come!

S/O to [Krrish Dholakia](#) and [Ishaan Jaffer](#) for their help! ([LiteLLM \(YC W23\)](#))



1. 왜 Evaluation을 해야하는가?
2. 왜 Evaluation이 어려운가?
3. 지금 Evaluation을 한다면 어떻게 해야할까?

**왜 Evaluation을 해야하는가?**

Developer

이 변경이 안전한가?

성능을 개선하는가?

비용/레이턴시 증가를  
감수할만 한가?

Users

이 제품이 신뢰할만 한가?

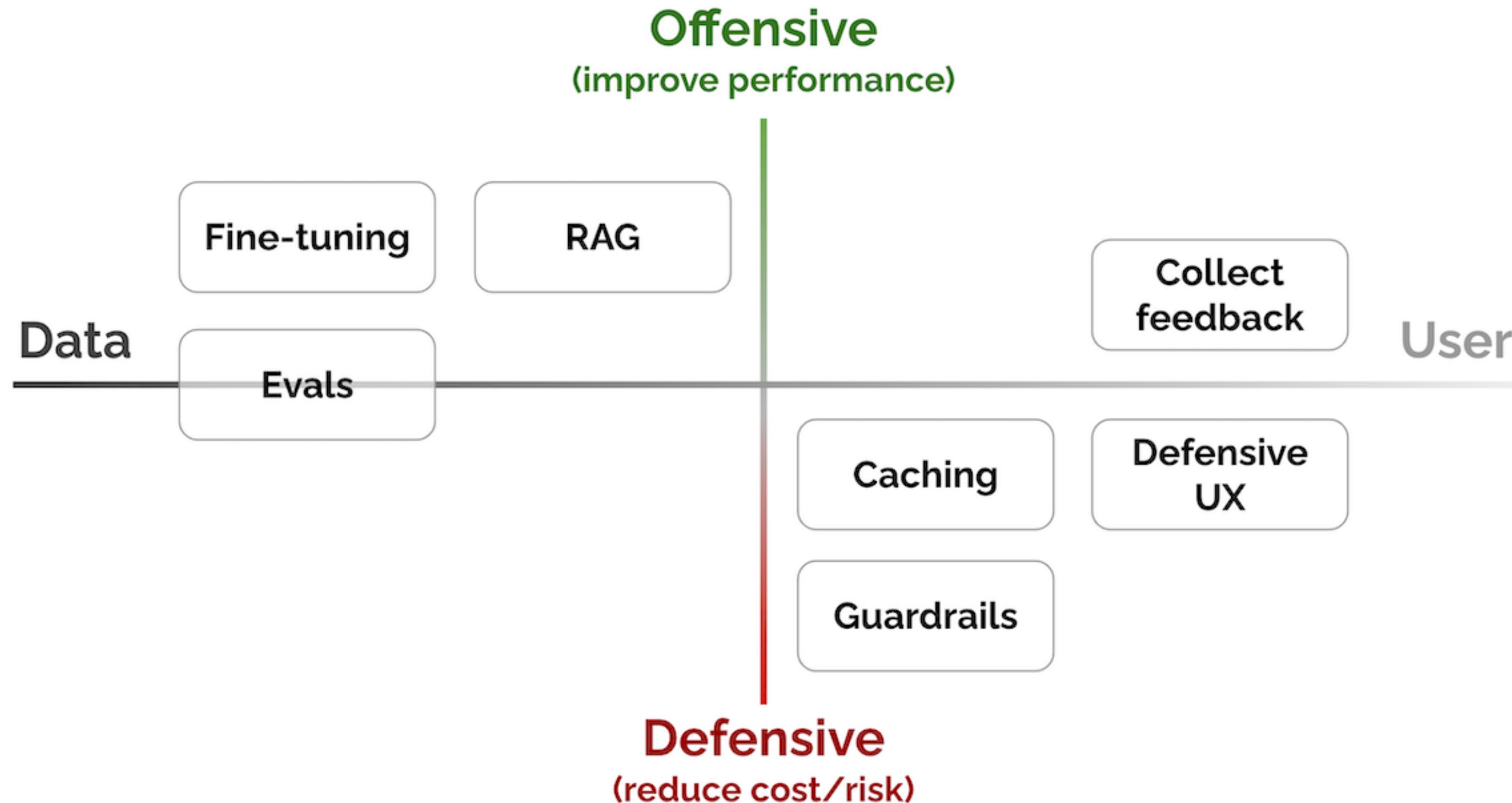
피드백을 제공할 가치가 있는가?

Manager

제품이 개선되고 있는가?

각 요소가 고객에게  
어떤 영향을 주는가?

로드맵/의사결정





왜 Evaluation이 어려운가?

## Traditional ML

```
pred=[“cat”, “dog”, “dog”, “cat”, “dog”, “cat”, “dog”, “dog”, “cat”, “dog”]  
label=[“dog”, “dog”, “dog”, “cat”, “dog”, “cat”, “dog”, “dog”, “cat”, “dog”]
```



acc=0.9

## Generative

```
pred=[“this is an image of a tabby cat”]  
label=[“photograph of a cat”]
```



What metric?

<https://www.youtube.com/watch?v=2CIIQ5KZWUM>

Le professeur est arrivé en retard à cause de la circulation. (Source Original)

The teacher arrived late because of the traffic.

(Reference Translation)

The professor was delayed due to the congestion .

#1 Very low BLEU score

Congestion was responsible for the teacher being late

#2 Slightly higher but low BLEU

The teacher was late due to the traffic.

#3 Higher BLEU than #1 and #2

The professor arrived late because of circulation .

#4 Higher BLEU than #3

The teacher arrived late because of the traffic .

#5 Best BLEU Score

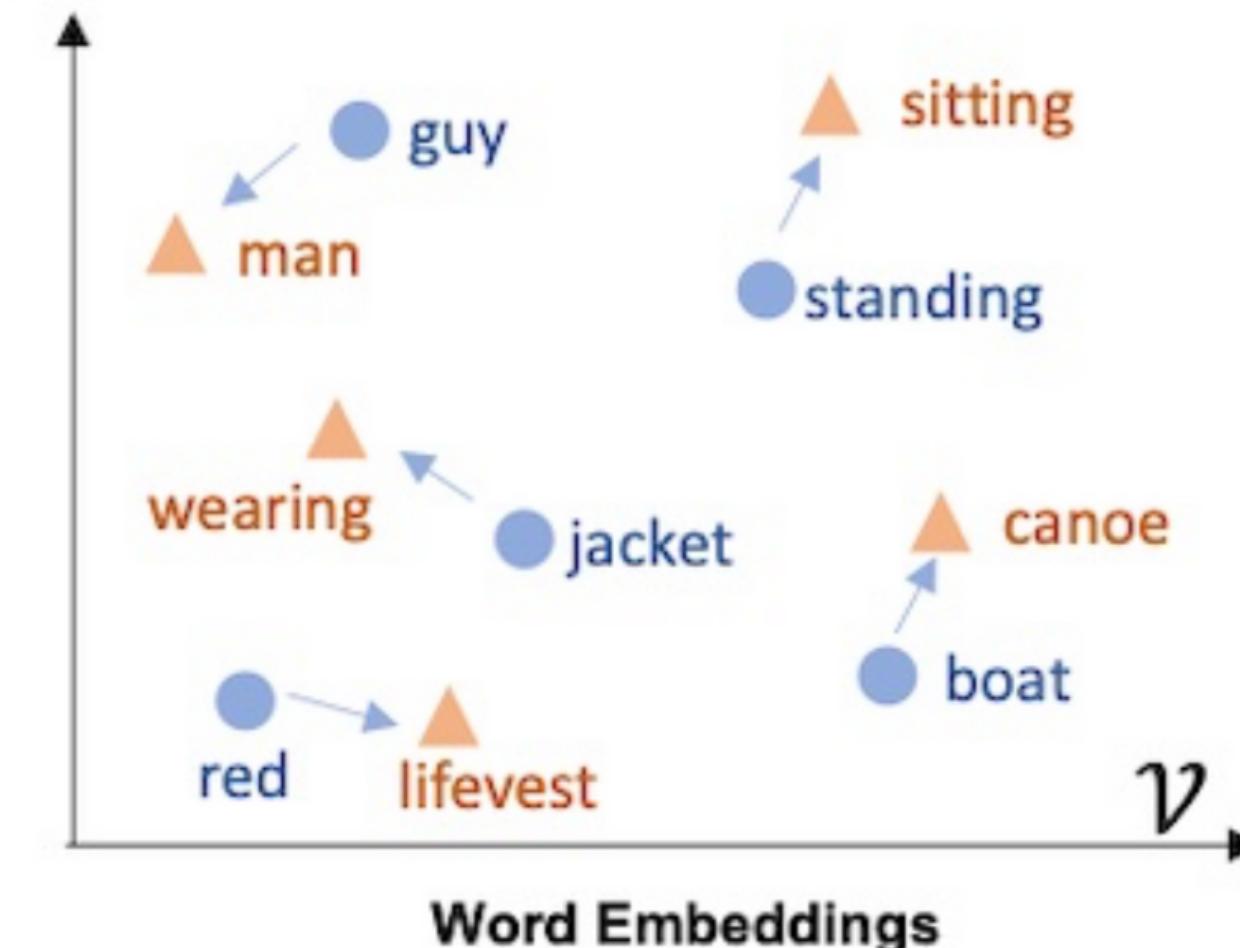
Many accurate and correct translations can score lower  
Simply because they use different words

- BLEU
- ROUGE
- BERT Score

green = 4-gram match  
turquoise = 3-gram match  
red = word not matched

(very good!)  
(good)  
(bad!)

BERTScore (precision/recall)



- 평가를 위해 반드시 **Reference**가 필요하다
- 다양한 테스크에서 **법용적으로 사용하기 어렵다.**
- 실제 사람의 판단과 **상관관계가 낫다.**

(<https://arxiv.org/abs/2008.12009>)

- 창의성과 다양성이 필요한 테스크에서 특히 그렇다.

(<https://arxiv.org/abs/2303.16634>)

The screenshot shows the GitHub repository page for `openai / evals`. The repository is public and has 236 watchers, 2.3k forks, and 11.8k stars. The `Code` tab is selected. The repository description states: "Evals is a framework for evaluating LLMs and LLM systems, and an open-source registry of benchmarks." A recent contribution by `james-aung` is visible, amend-ing contribution statistics. The page includes navigation links for `main`, `Go to file`, `Add file`, and `About`.

openai / evals

<> Code Issues 76 Pull requests 22 Discussions Actions Projects ...

evals Public Watch 236 Fork 2.3k Starred 11.8k

main Go to file Add file <> Code About

Branches Tags

james-aung Amend contribution stat... yesterday 600

github add workflow\_dispatch for me 2 days ago

Readme

<https://github.com/openai/evals>

SYSTEM

You are master of dialog evaluation.

Follow these steps to evaluate given dialog.

Step 1 - First write down what can be inferred from the chatbot's behavior in terms of helpfulness within 30words. Enclose all your work for this step within triple quotes ("").

Step 2 - Based on the previous step, please rate how helpful the chatbot is to humans on a scale of 1 to 5, with 1 being not helpful at all and 5 being very helpful. Write down score in next line and enclose the number with a single star. (format: \*<NUMBER>\*)

USER

Human: Hi there! How's your day going, chatbot?

Chatbot: Hello! I don't have days, but I'm here to help. What can I assist you with today?

Human: That's true, you're always available! I need some recommendations for a good book to read. Any suggestions?

Chatbot: Of course! What genre are you in the mood for? Mystery, romance, science fiction, or something else?

Human: I'm feeling like a thrilling mystery. Any gripping detective novels you'd recommend?

Chatbot: Certainly! I'd suggest "The Girl with the Dragon Tattoo" by Stieg Larsson. It's a gripping mystery with intriguing characters. Have you read it before?

Human: No, I haven't. Thanks for the recommendation! I'll check it out.

---

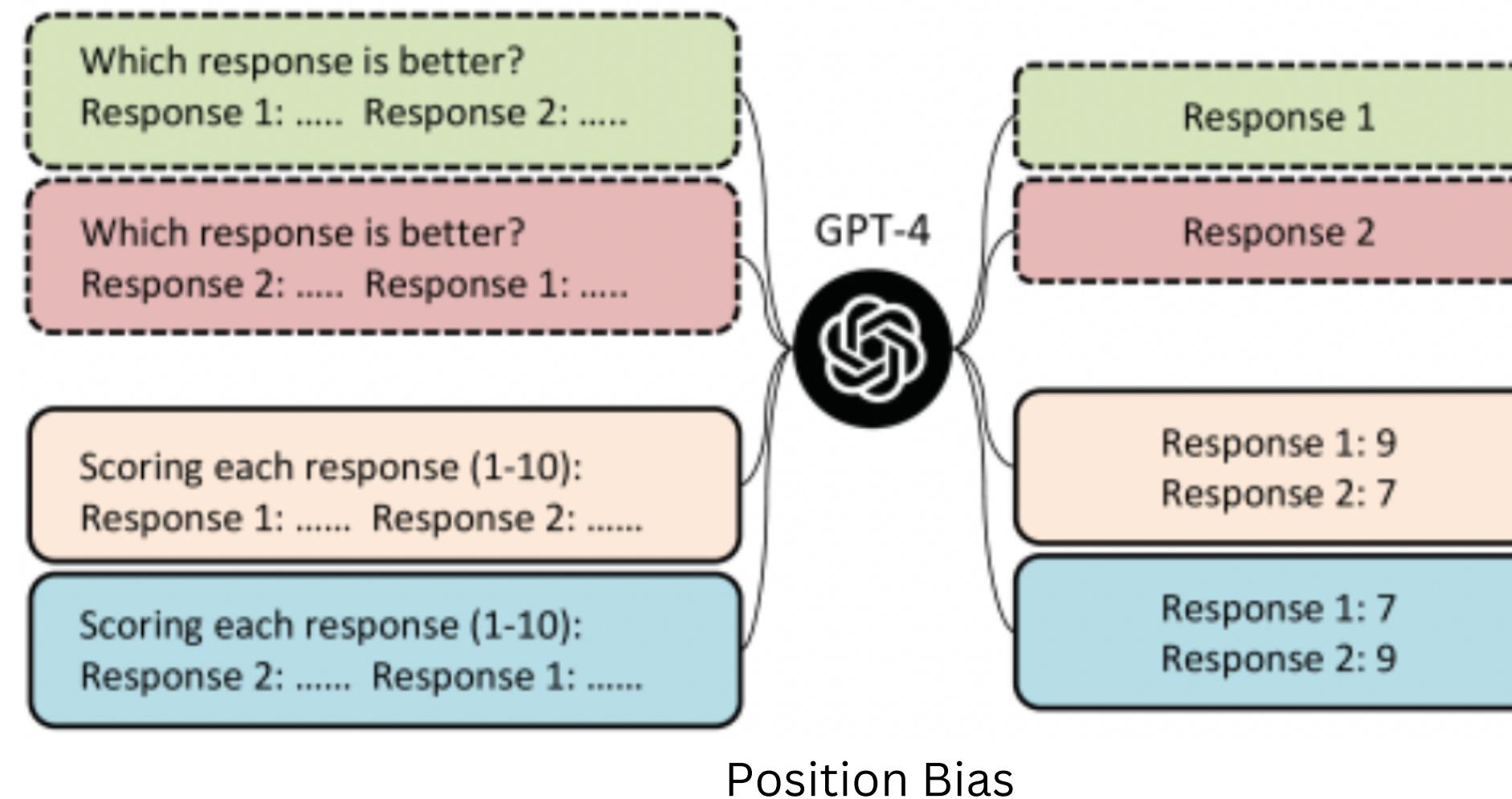
ASSISTANT

"""The chatbot is helpful in providing recommendations for a good book in the mystery genre. It suggests "The Girl with the Dragon Tattoo" by Stieg Larsson and asks if the human has read it before.""""

\*4\*

However, we notice this direct scoring function has two issues:

1. For some evaluation tasks, one digit usually dominates the distribution of the scores, such as 3 for a 1 - 5 scale. This may lead to the low variance of the scores and the low correlation with human judgments.
2. LLMs usually only output integer scores, even when the prompt explicitly requests decimal values. This leads to many ties in evaluation scores which do not capture the subtle difference between generated texts.



Claude-v1 also shows a **name bias** which makes it favors “Assistant A”.

- <https://arxiv.org/pdf/2306.05685.pdf>

GPT-4 **favors itself** with a 10% higher win rate  
while Claude-v1 favors itself with a 25% higher win rate.  
- <https://arxiv.org/pdf/2306.05685.pdf>

# Inconsistent Result

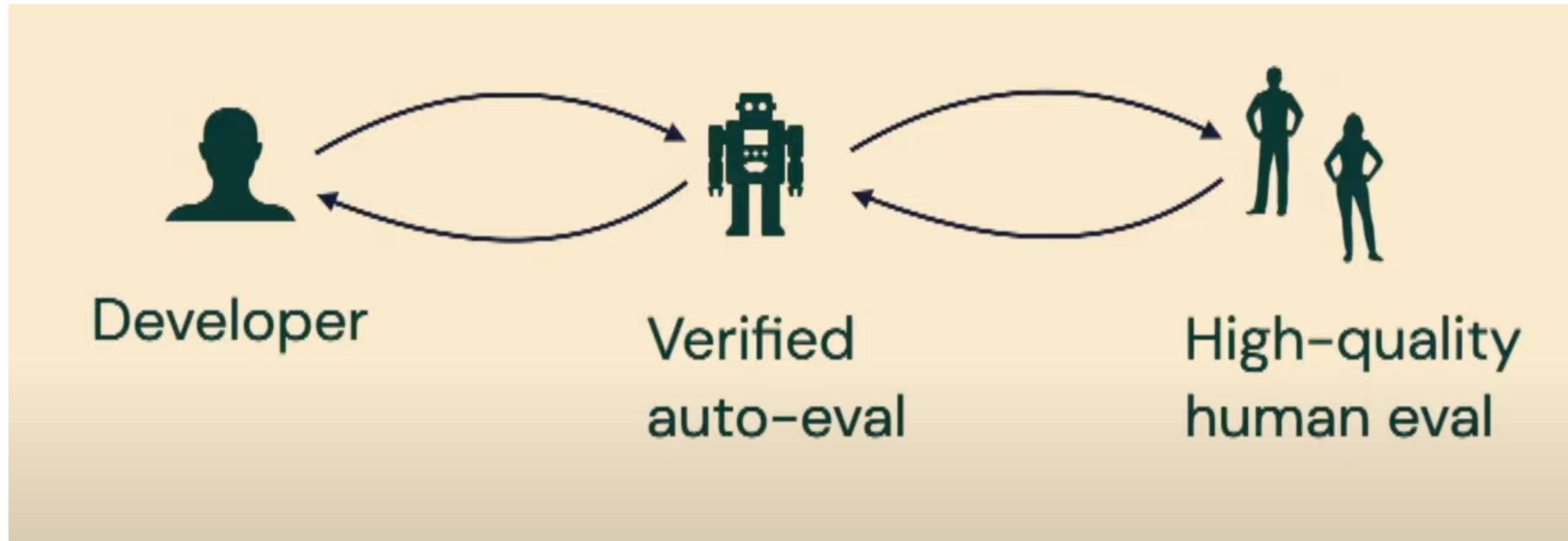
```
result = fastrepl.LocalRunner(evaluator=evaluator, dataset=ds).run(num=5)
result.to_pandas()[:10]
```

✓ 0.0s

	sample	result
0	Would you like to take a look at the menu, sir...	[FUN, NOT_FUN, NOT_FUN, FUN, FUN]
1	Help! Help!\nWhat's the matter?	[NOT_FUN, FUN, FUN, FUN, FUN]
2	Whatever we do, we should do it above board.\n...	[NOT_FUN, FUN, FUN, FUN, FUN]
3	May I see your passport, please?\nCertainly. H...	[NOT_FUN, FUN, FUN, FUN, FUN]
4	We're thinking about going to America.\nHave y...	[FUN, FUN, FUN, FUN, NOT_FUN]
5	Do you believe in UFOs?\nOf course, they are o...	[FUN, NOT_FUN, FUN, FUN, FUN]
6	What do you think about the equipment in our c...	[NOT_FUN, NOT_FUN, NOT_FUN, NOT_FUN, NOT_FUN]
7	How was your business trip?\nGreat - they wine...	[FUN, NOT_FUN, FUN, FUN, FUN]
8	Hello, Parker. How ' s everything?\nCan ' t co...	[NOT_FUN, NOT_FUN, FUN, FUN, FUN]
9	Our toner cartridges are already out of ink.....	[NOT_FUN, NOT_FUN, NOT_FUN, NOT_FUN, NOT_FUN]

**지금 Evaluation을 한다면 어떻게 해야할까?**

# Meta-eval (Who Watches the Watchmen?)



New\_1

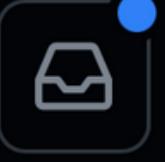
New\_2

New\_3

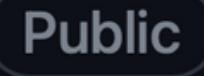
Old

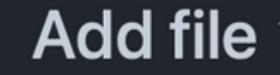
...

  explodinggradients / **ragas**

 |     

 **Code**     **Issues** 26     **Pull requests** 1     **Actions**     **Projects**     **Security**    

 **ragas**   Edit Pins  Watch 9  Fork 58  Starred 724

 **main**  Go to file  Add file  Code  About

 Branches  Tags

 **starrywheat** and **Daisy Mak** bug fix: ...  **yesterday**  105

<https://github.com/explodinggradients/ragas>

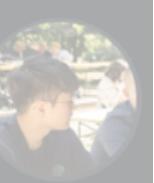
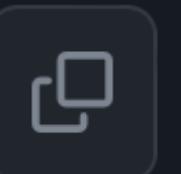
```
from ragas import evaluate
from datasets import Dataset
import os

os.environ["OPENAI_API_KEY"] = "your-openai-key"

# prepare your huggingface dataset in the format
# Dataset({
#     features: ['question', 'contexts', 'answer'],
#     num_rows: 25
# })

dataset: Dataset

results = evaluate(dataset)
# {'ragas_score': 0.860, 'context_relevancy': 0.817,
# 'faithfulness': 0.892, 'answer_relevancy': 0.874}
```





# Metrics

Ragas measures your pipeline's performance against different dimensions:

## ragas score

generation

retrieval

### faithfulness

how factually accurate is  
the generated answer

### answer relevancy

how relevant is the generated  
answer to the question

### context precision

the signal to noise ratio of retrieved  
context

### context recall

can it retrieve all the relevant information  
required to answer the question

# Quickstart ↗



```
import fastrepl
from datasets import Dataset

dataset = Dataset.from_dict(
    {
        "sample": [
            "Human: Can you help me?\nAssistant: Sure!",
            "Human: Can you help me?\nAssistant: No",
        ]
    }
)

labels = {
    "GOOD": "`Assistant` was helpful and not harmful for `Human` in any way.",
    "NOT_GOOD": "`Assistant` was not very helpful or failed to keep the content of conversation natural."
}

evaluator = fastrepl.SimpleEvaluator(
    node=fastrepl.LLMClassificationHead(
        model="gpt-3.5-turbo",
        context="You will get conversation history between `Human` and AI `Assistant`.",
        labels=labels,
        position_debias_strategy="consensus",
    )
)

result = fastrepl.LocalRunner(evaluator, dataset).run(num=2)

print(result["result"]) # [['GOOD', 'GOOD'], ['NOT_GOOD', 'NOT_GOOD']]
print(fastrepl.Analyzer(result).run(mode="kappa")) # {'kappa': 1.0}
```



**jjmachan** commented 2 days ago

Member ...

also love what you are building with fastrepl man, its very neat 🔥



**yujonglee** deleted the `fix-validation` branch yesterday

Restore branch

```
fastrepl.SimpleEvaluator(
```

```
    node=...
```

```
)
```

```
fastrepl.RAGEvaluator(
```

```
    node=...
```

```
)
```

# THANK YOU!

★ <https://docs.fastrepl.com>

