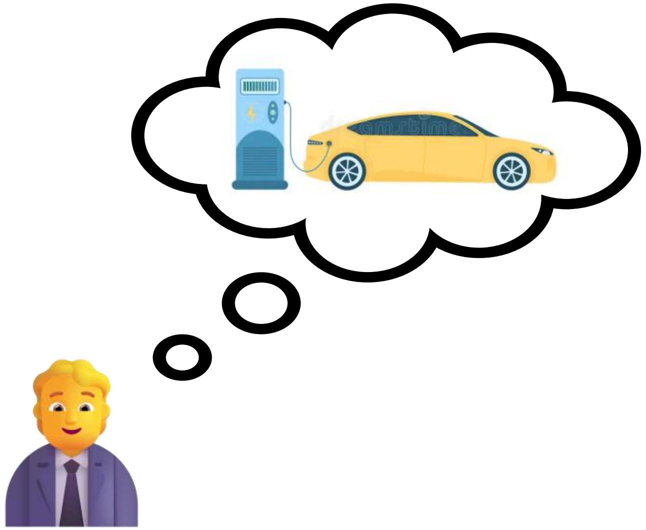
Information Elicitation Meets Large Language Models

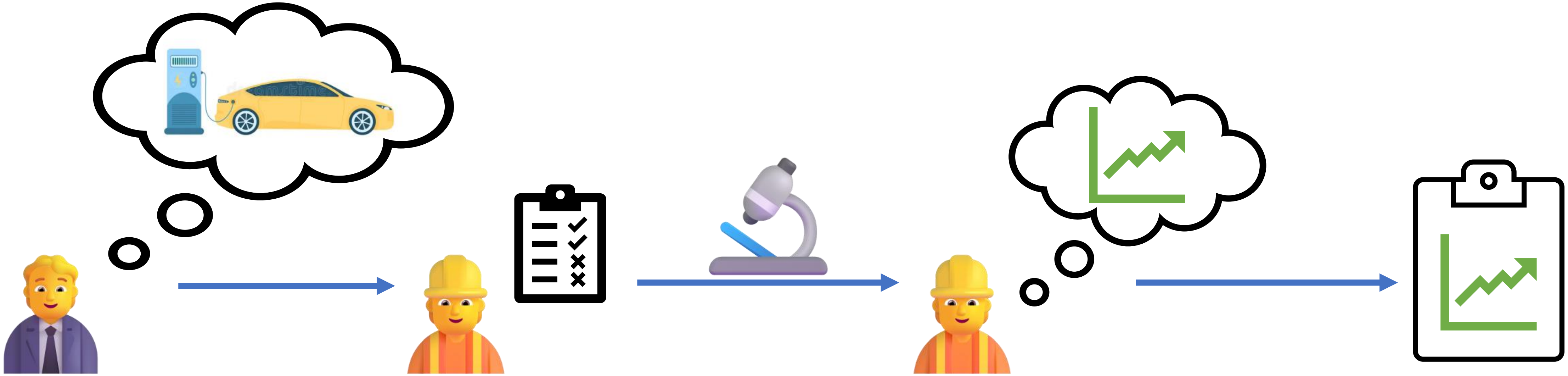
Yuxuan Lu
Peking University

Shengwei Xu
University of Michigan

Information Elicitation

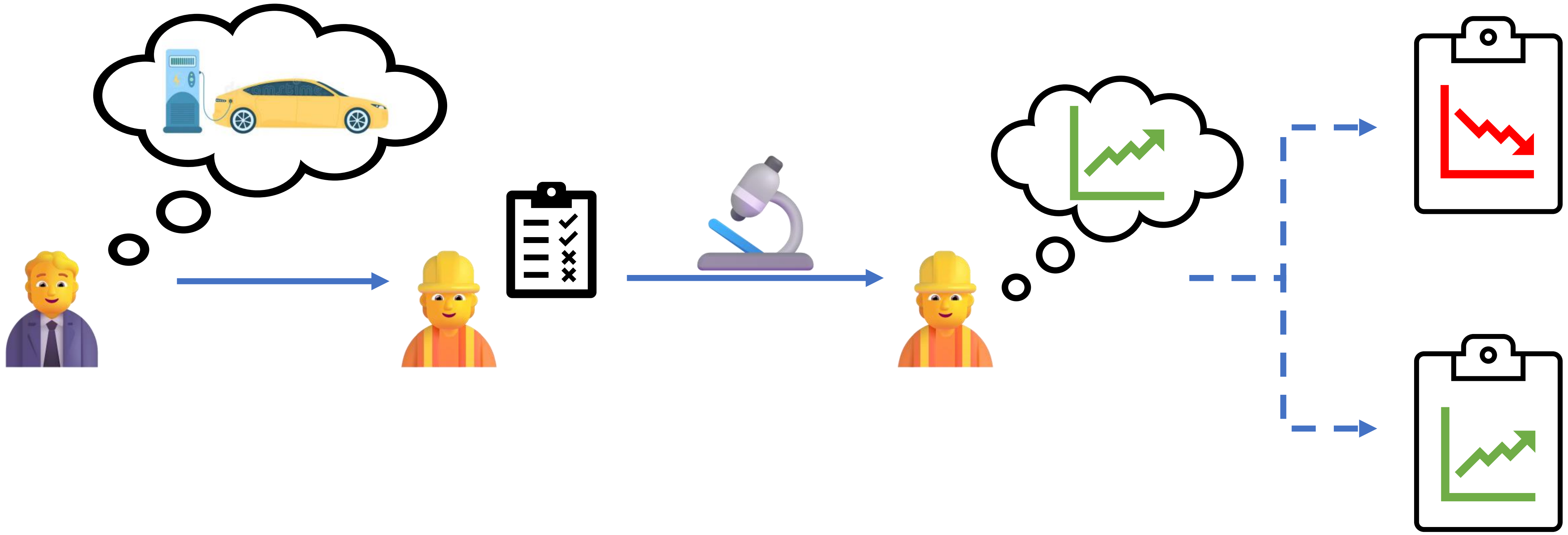


Information Elicitation

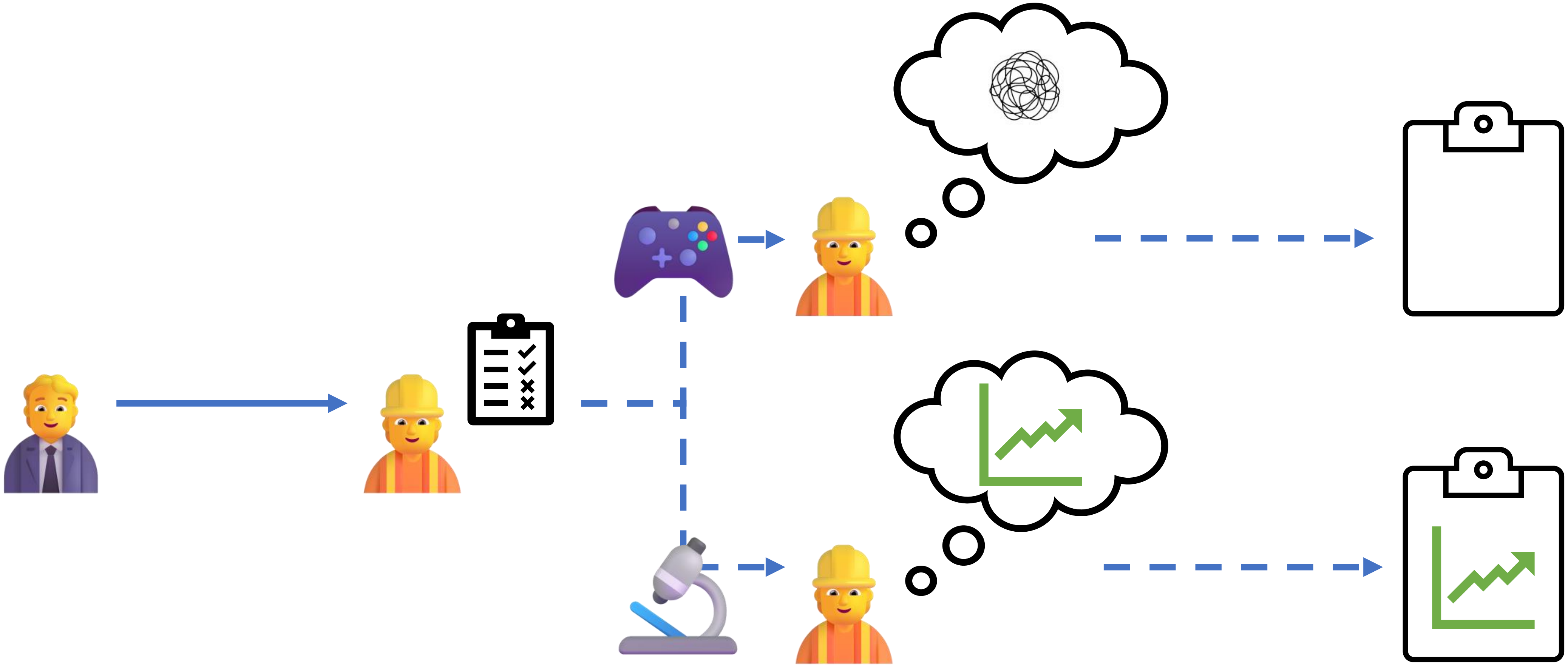


Principal-Agent Problem [Ali and Silvey, 1966]

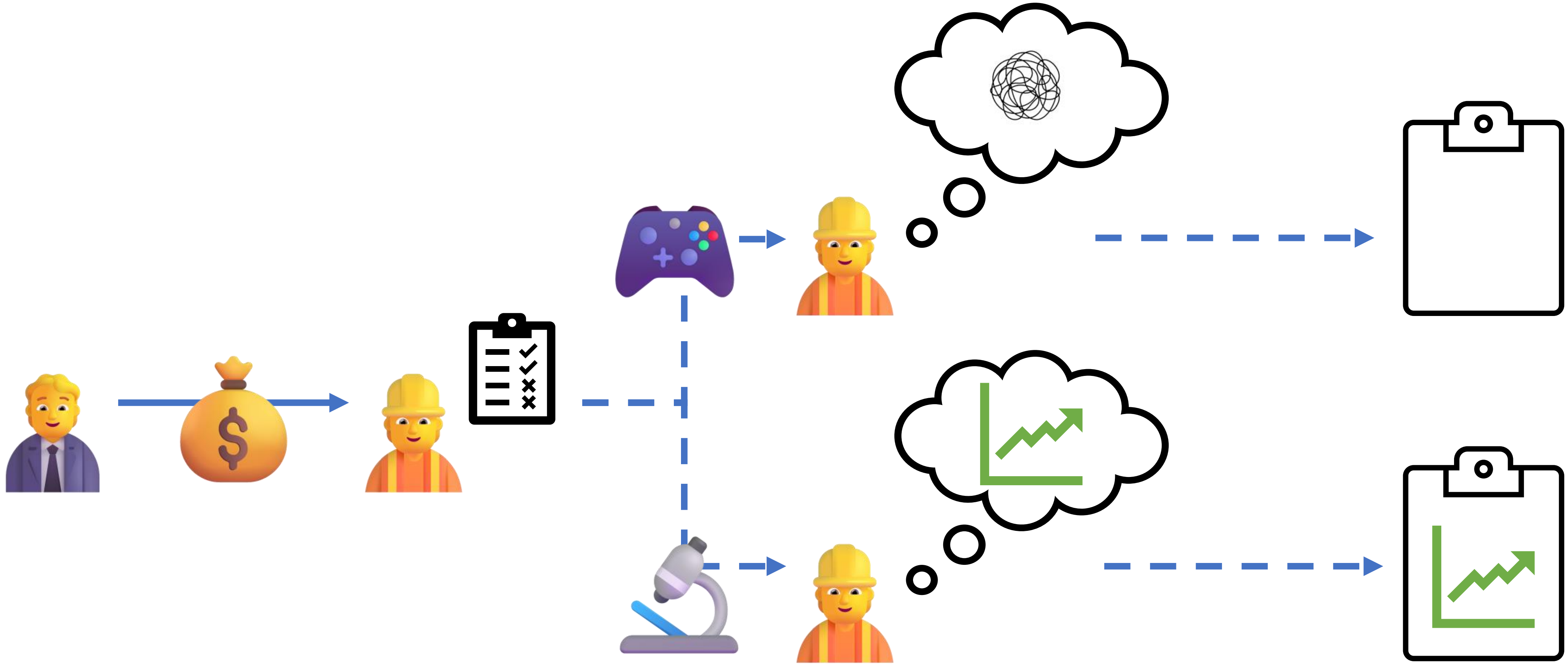
Strategic in Report



Strategic in Effort

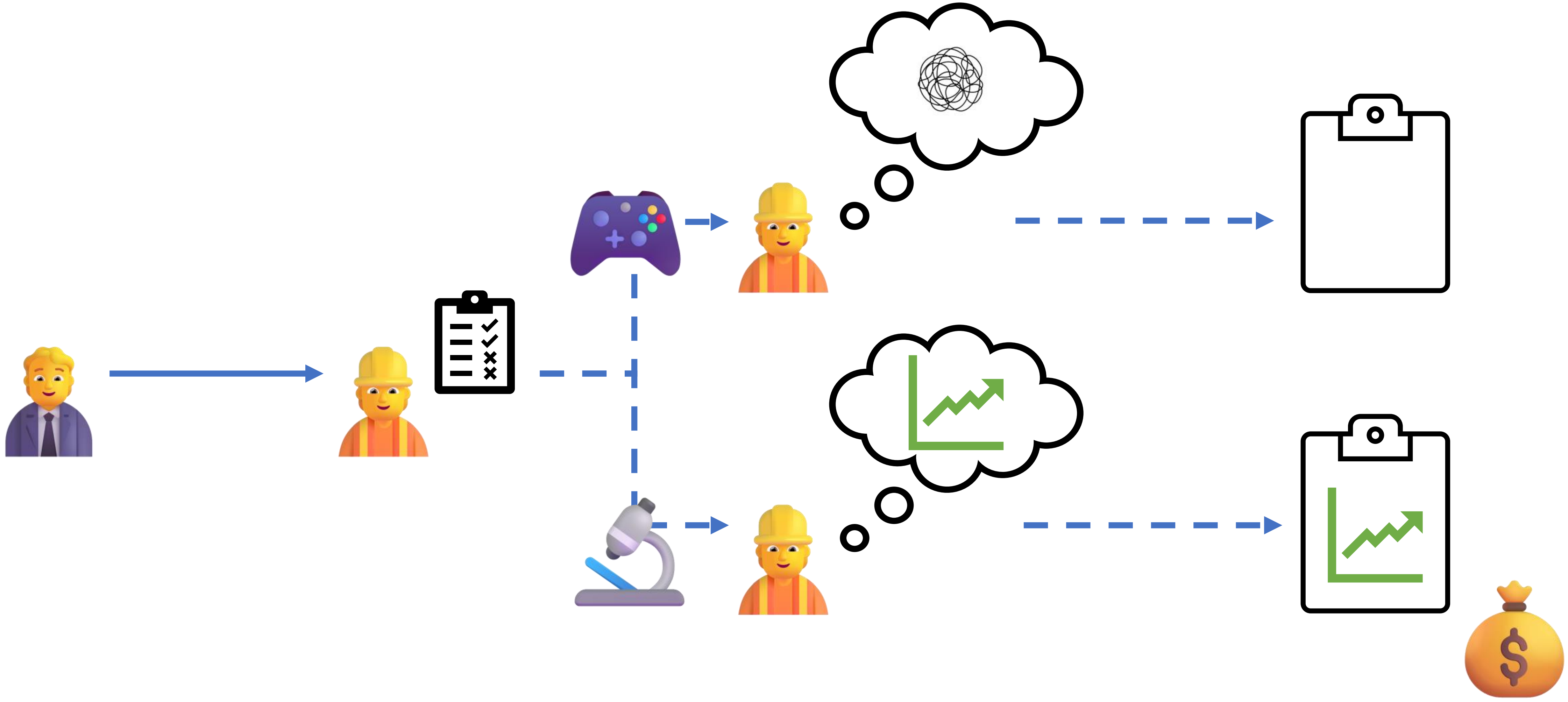


Incentive Mechanism

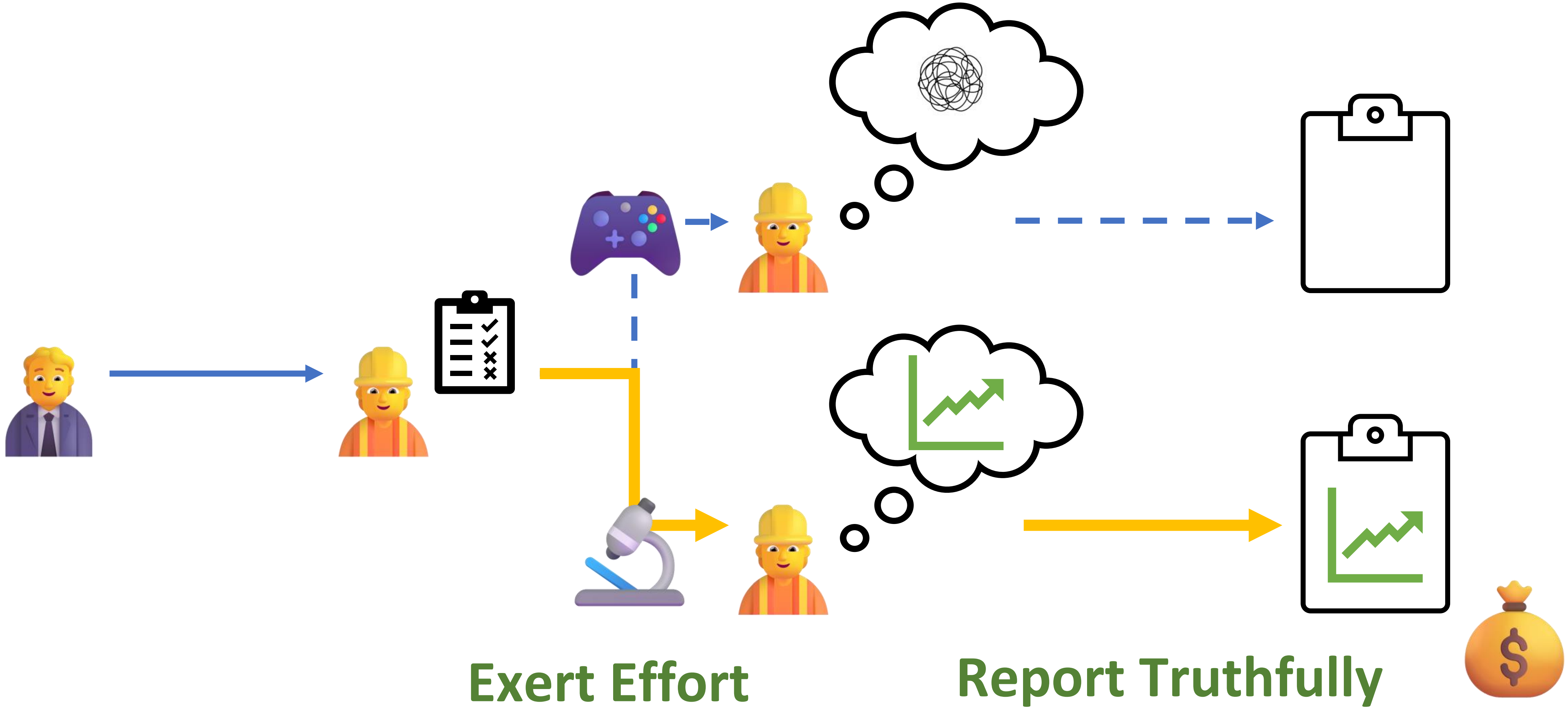


Assumption: self-interest & rational agent

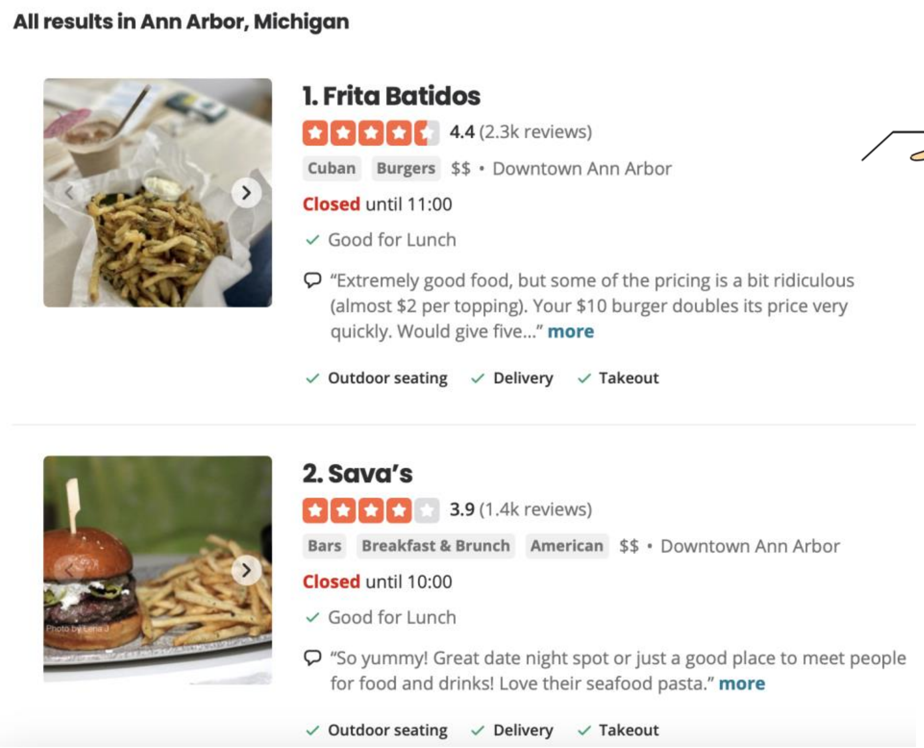
Incentive Mechanism



Information Elicitation Goal



Applications



Business Review

Is the problem well-specified, interesting, and ambitious?

4. Top 25% (Excellent): The problem is very well-specified, captivating, and shows a high level of ambition. It is interesting, relevant, and presents a significant challenge.

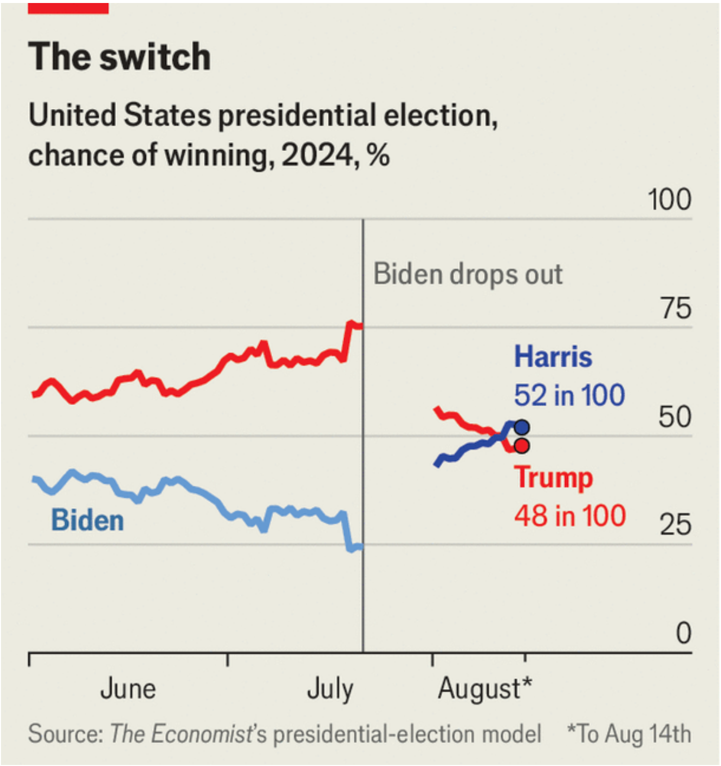
Does the report sufficiently cover existing work?

3. Top 60% (Good): The report covers existing work adequately.

Are Methods, Datasets, and Evaluation well-specified and appropriate?

4. Top 25% (Excellent): The methods, datasets, and evaluation are very well-specified and appropriate, with only minor areas for improvement.

Peer Grading / Peer Review



Opinion Polling

Applications in AI

CONTENT

CONTENT

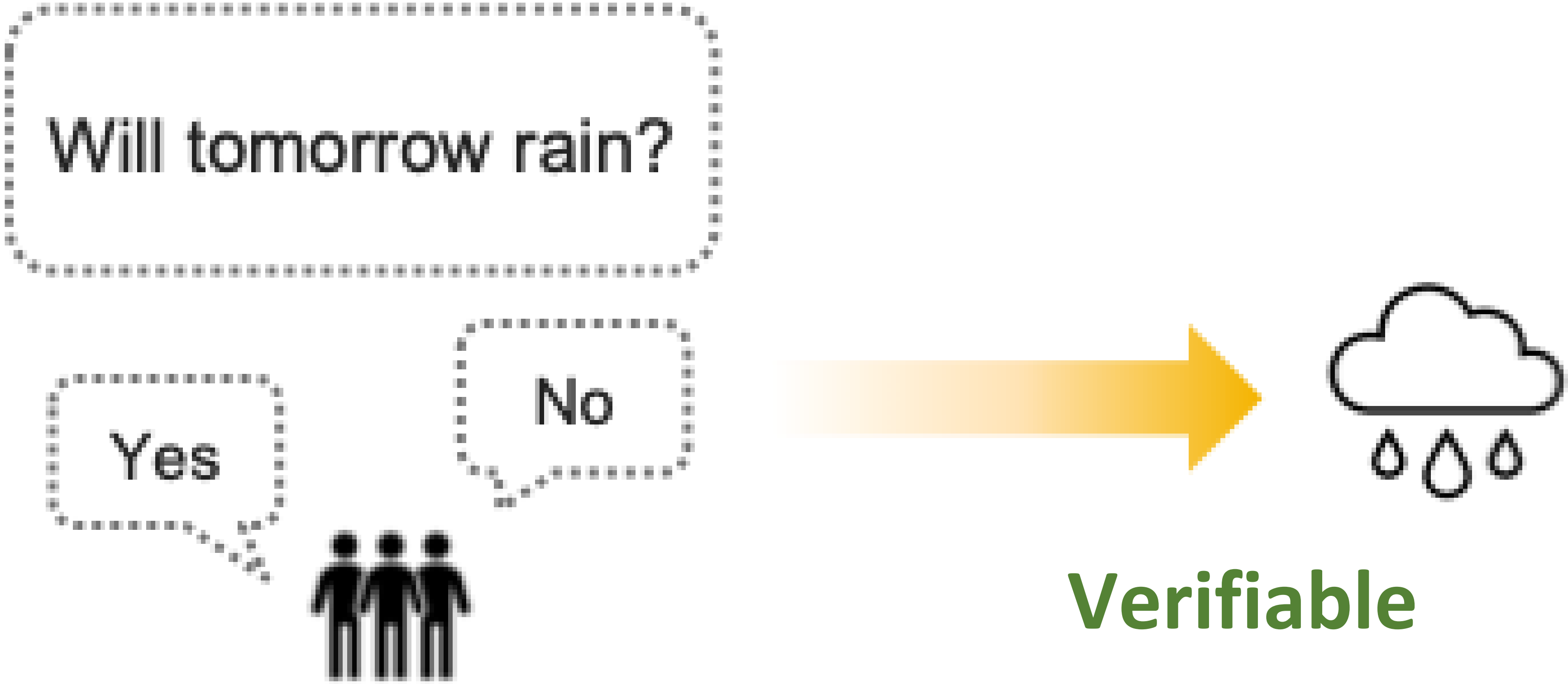
01

Information Elicitation

An Overview: Progresses and Boundaries

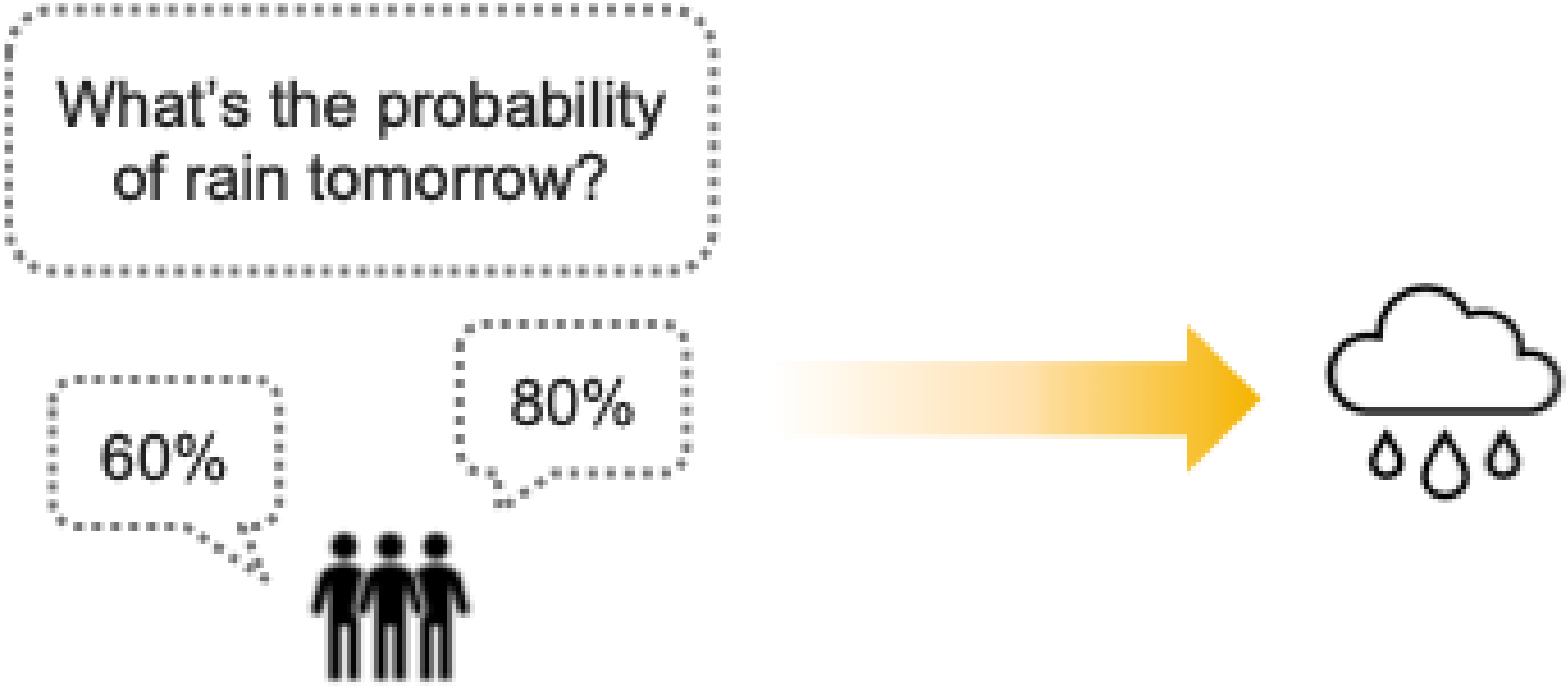
Example

Example

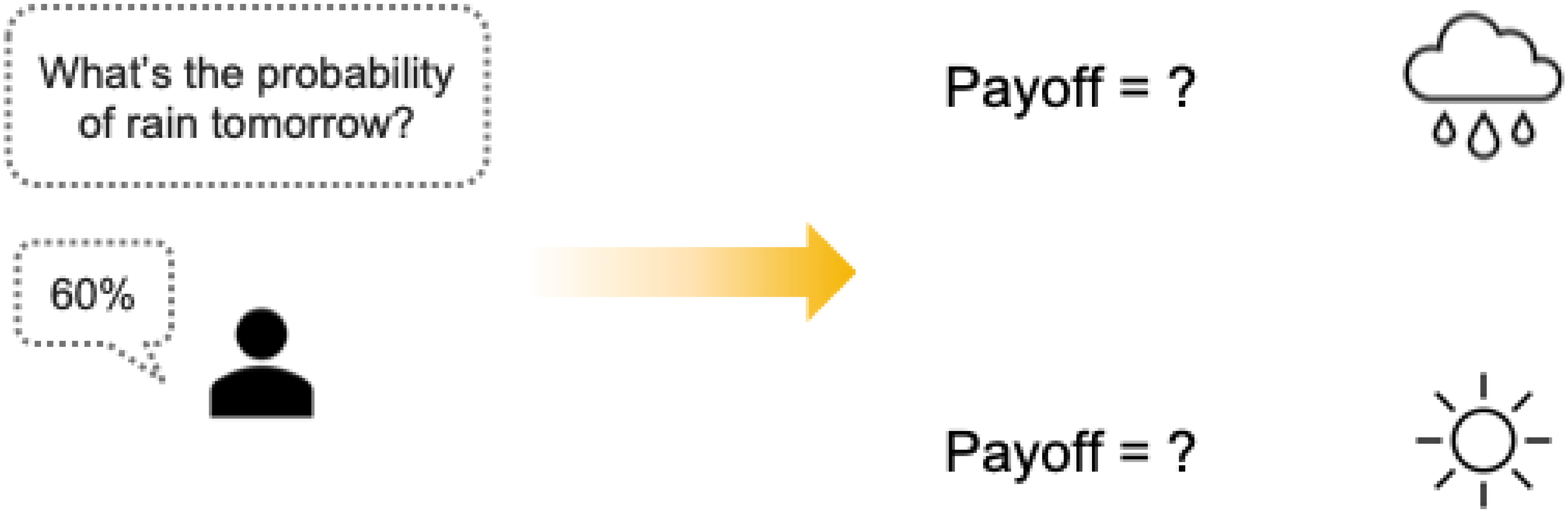


Verifiable

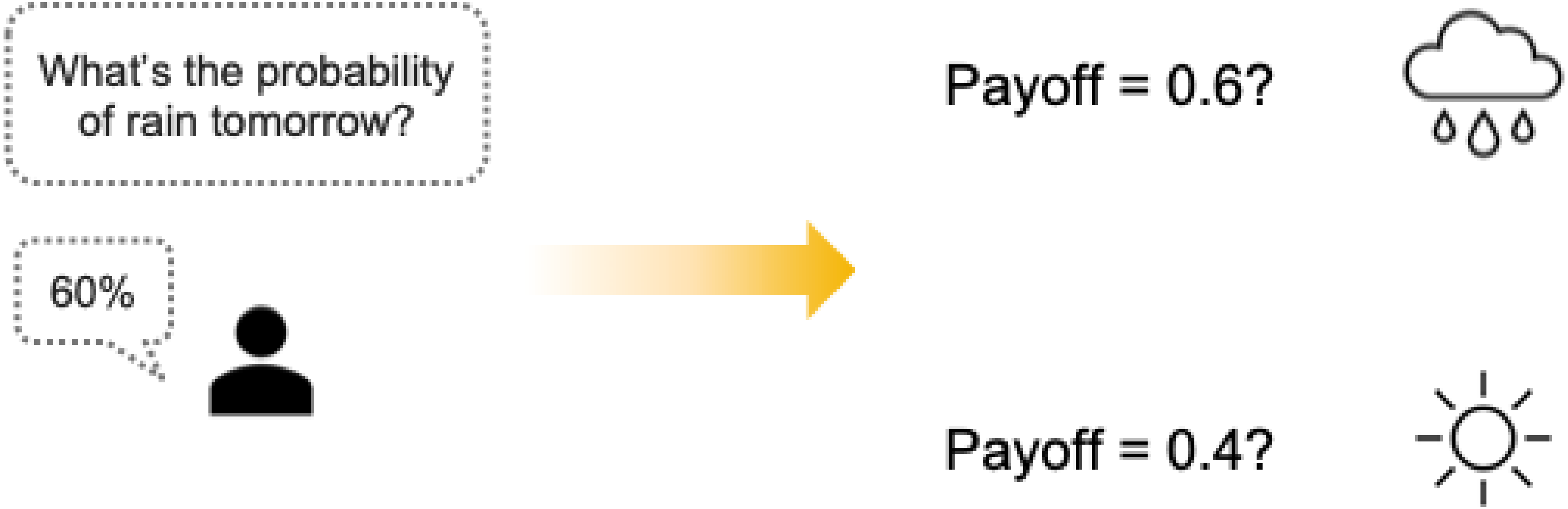
Example: Probability Forecast



Example: Probability Forecast



Example: Probability Forecast



Model

- n states of the world $\omega \in \Omega$, with state space Ω
  - E.g.: Ω = {Rain, No rain}
  - Δ_Ω : Set of probability distributions on Ω
- Agent has a belief $\mathrm{q} \in \Delta_\Omega$
  - E.g.: $\mathrm{q} = (0.6, 0.4)$ representing rain with prob 0.6
  - In binary case, for simplicity, $\mathrm{q} \coloneqq \mathrm{q}_1$

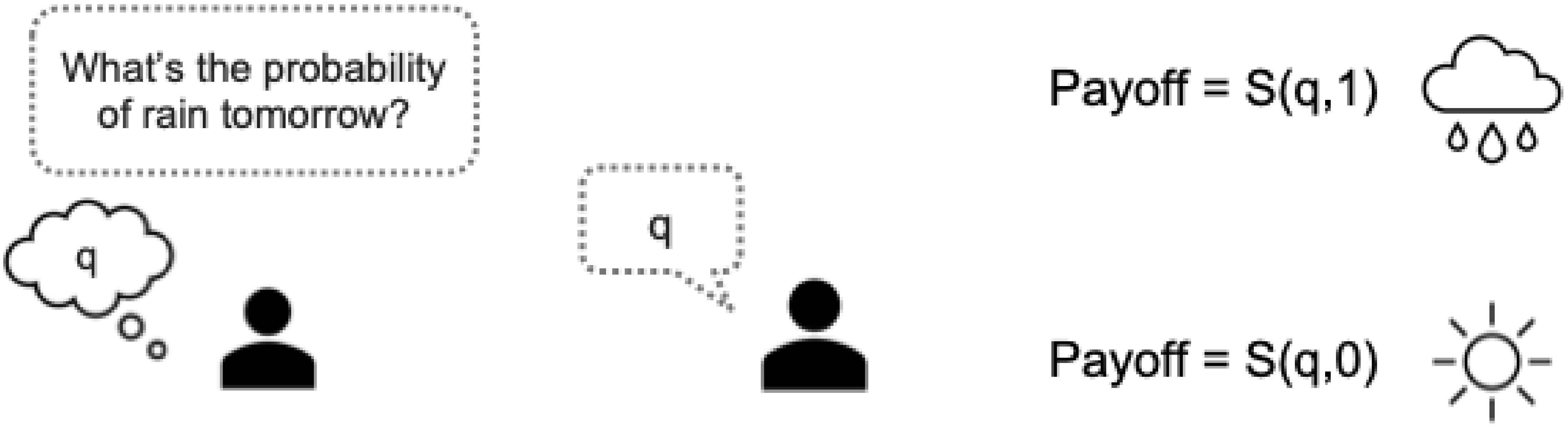
Proper Scoring Rule (PSR)

A scoring rule $\mathrm{S}: \Delta_\Omega \times \ \Omega \to \mathbb{R}$

Principal's Goal:

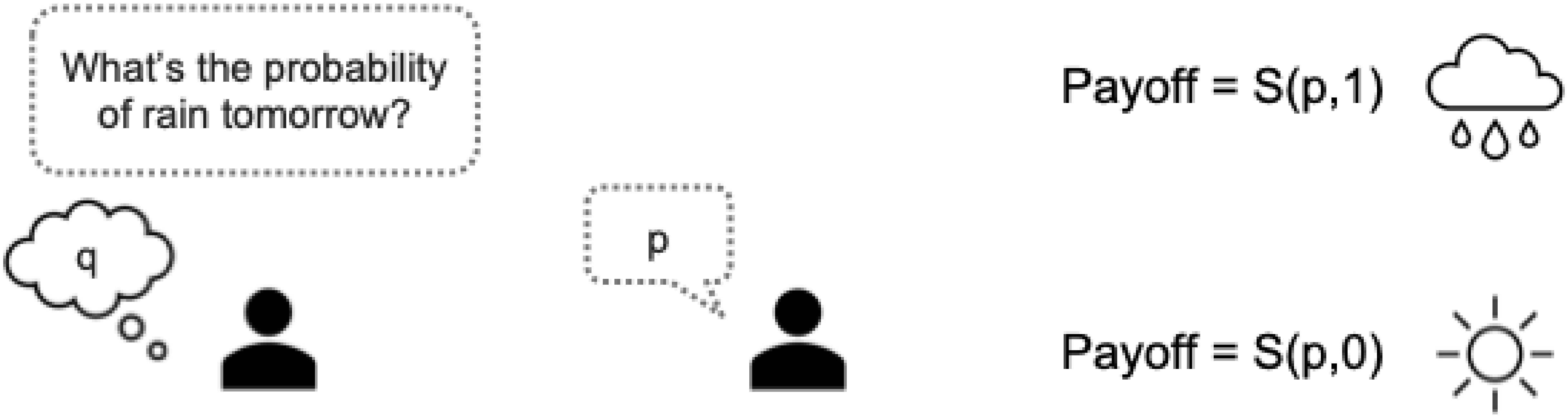
- Agent maximizes expected payoff when reporting her true belief
- Proper Scoring Rule := $\mathbb{E}_{\omega\sim q}[S(q,\omega)] \geq \mathbb{E}_{\omega\sim q}[S(p,\omega)]$

Example: Truthful Report



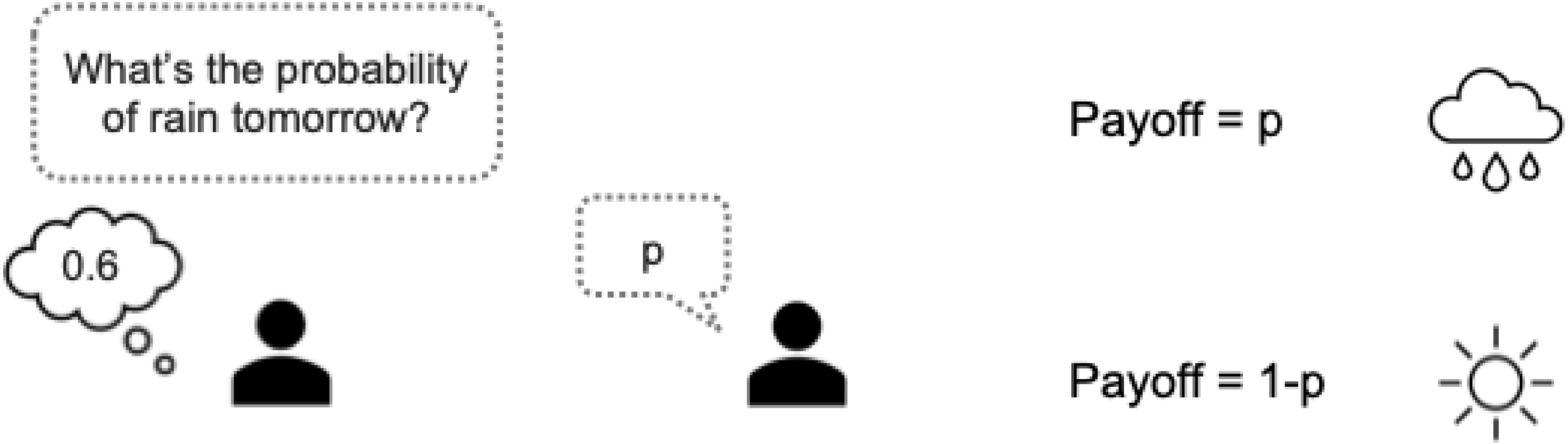
$$\text{Expected Payoff} = q\, S(q,1) + (1-q)\, S(q,0)$$

Example: Untruthful Report



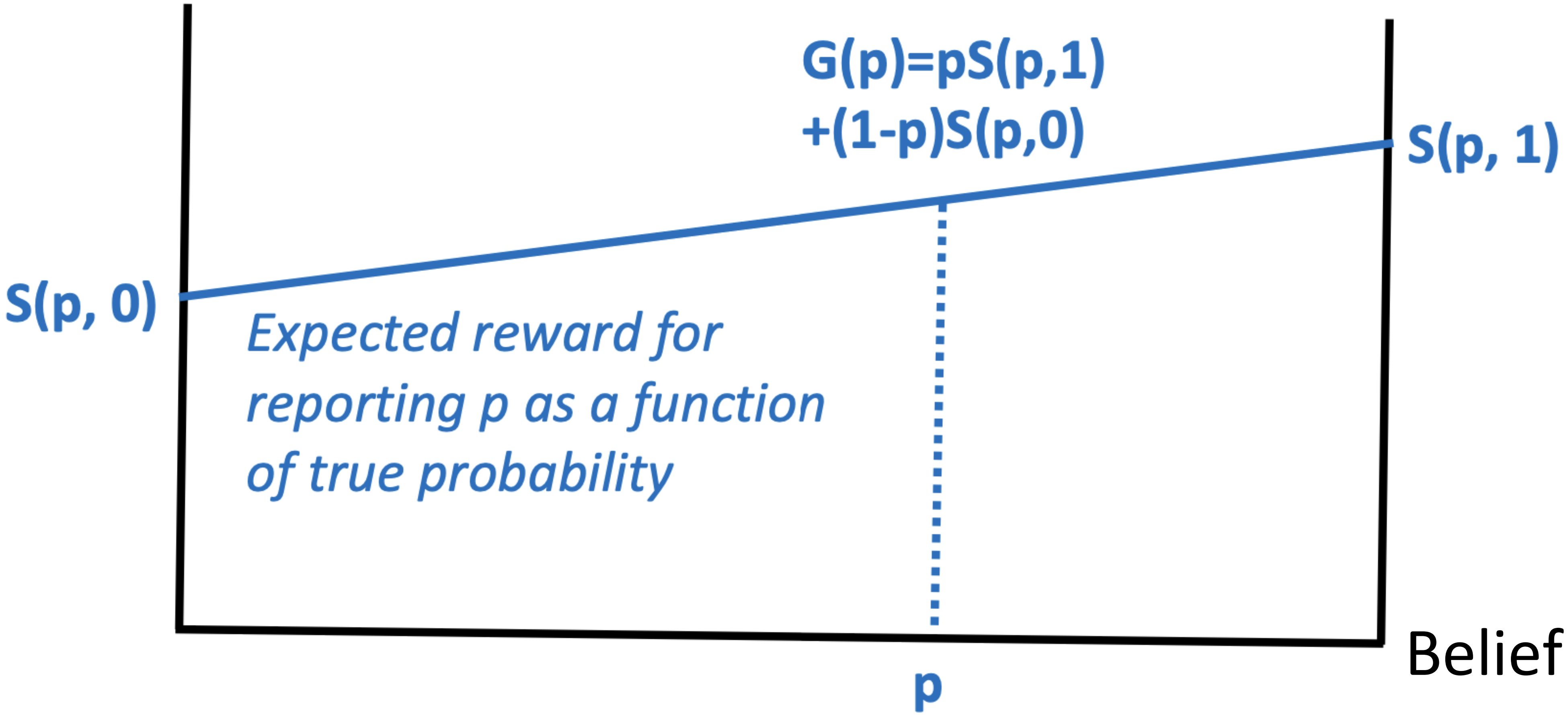
Expected Payoff = q S(p,1) + (1-q) S(p,0)

Linear Payoff: Not Proper

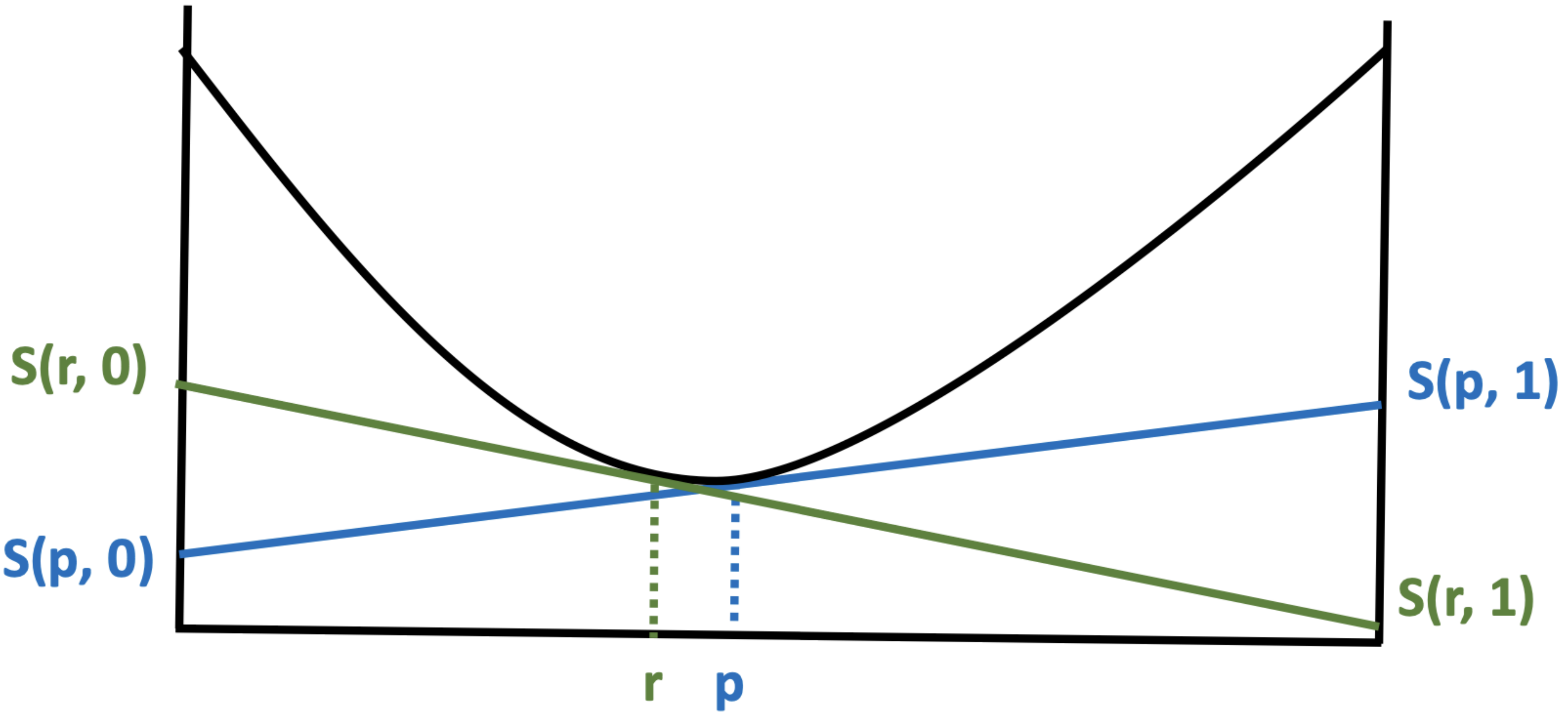


$$\text{Expected Payoff} = 0.6\ p + (1-0.6)\ (1-p)$$

How to design PSR $S(q,\omega)$?



Convexity!



How to design the PSR $S(q, \omega)$?

- Proper Scoring Rule := $\mathbb{E}_{\omega \sim q}[S(q, \omega)] \geq \mathbb{E}_{\omega \sim q}[S(p, \omega)]$
- PSR $\Leftrightarrow$ $G(q) := \sum_{\omega} q_{\omega} S(q, \omega)$ convex [McCarthy 1956]
  - $S(q, \omega)$ is the sub-gradient of G at q

How to design the PSR $S(q,\omega)$?

- Proper Scoring Rule := $\mathbb{E}_{\omega \sim q}[S(q,\omega)] \geq \mathbb{E}_{\omega \sim q}[S(p,\omega)]$
- PSR $\Leftrightarrow$ $G(q) := \sum_{\omega} q_{\omega} S(q,\omega)$ convex [McCarthy 1956]
    - $S(q,\omega)$ is the sub-gradient of G at q

Log Scoring Rule: $S(q,\omega) = \log q_{\omega}$

Brier/Quadratic Scoring Rule: $S(q,\omega) = -\sum_{\gamma} \left(1[\gamma = \omega] - q_{\gamma}\right)^2$

Proper Scoring Rule & Loss Function

Proper Scoring Rule	Loss Function
Log score	Cross-Entropy
Brier score	Mean squared error
Truthfulness	**Calibration**: when forecasting x%, roughly x% should turn out “yes”

Information Elicitation Mechanism

- Case 1: with verification: PSR ✅
- Case 2: without verification: ?

Subjectivity

Cost of verification …

Basic Model

- n states of the world $\omega \in \Omega$, with state space Ω
- Agents have a common prior belief $\mathrm{p} \in \Delta_{\Omega}$
- Agents privately observe a signal $x_i \in \Sigma$, following information structure $\Pr[\cdot \mid \omega]$
- The agents are asked to report their private signals
- Assume all agents' signals are independent conditional on ω
- Assume all agents' signals are stochastic relevance

Original Peer Prediction Mechanism

[Miller, Resnick, and Zeckhauser 2005]

- Score an agent based on the correlation between her report and her peer's.
- Agent i's report $\tilde{x}_i \in \Sigma$, agent j's (the peer) report $\tilde{x}_j \in \Sigma$, $|\Sigma| = C$

$$Score_i = \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$$

Peer's report Agent i's report

Interpret the Score of Peer Prediction

(Informal) In Peer Prediction, the expected score of agent i is

$$\sum_{x_i} \Pr[X_i = x_i] \sum_{x_j} \Pr[X_j = x_j \mid X_i = x_i] \log \Pr[X_j = x_j \mid X_i = x_i]$$
$$= \sum_{x_i, x_j} \Pr[X_i = x_i, X_j = x_j] \log \Pr[X_j = x_j \mid X_i = x_i]$$
$$= -H(X_j \mid X_i),$$

$$-H(X_j \mid X_i) = \boldsymbol{I(X_i; X_j)} - H(X_j)$$

Constant from agent i's view

Peer's report

Agent i's report

Original Peer Prediction Mechanism

[Miller, Resnick, and Zeckhauser 2005]

- Score an agent based on the correlation between her report and her peer's.
- Agent i's report $\tilde{x}_i \in \Sigma$, agent j's (the peer) report $\tilde{x}_j \in \Sigma$, $|\Sigma| = \mathrm{C}$

$$Score_i = \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$$

- **Exerting effort and truthfully reporting is a Nash Equilibrium**

Original Peer Prediction Mechanism

[Miller, Resnick, and Zeckhauser 2005]

- Score an agent based on the correlation between her report and her peer's.
- Exerting effort and truthfully reporting is a Nash Equilibrium

$$Score_i = \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$$

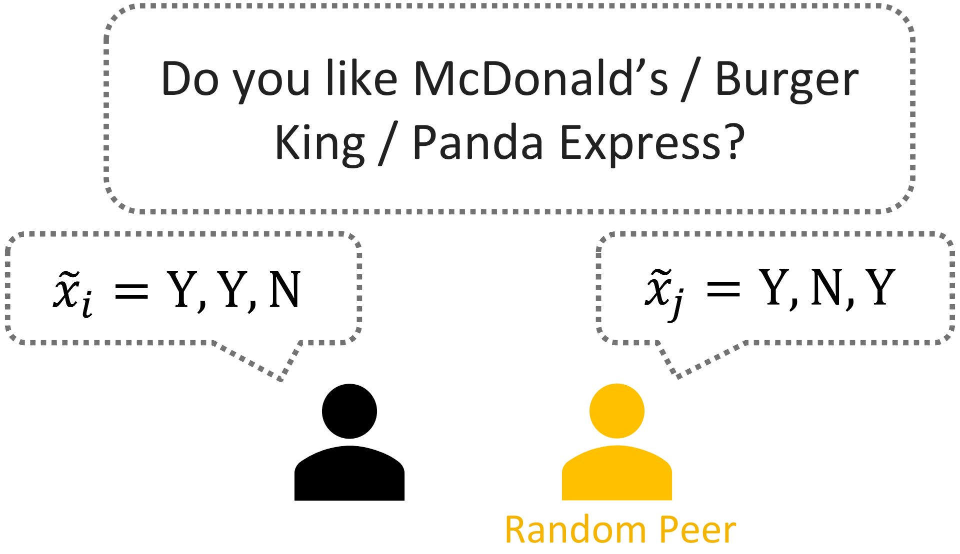
$\tilde{x}_i$ \ $\tilde{x}_j$	Y	N
Y	**1/3**	**1/6**
N	**1/6**	**1/3**

Assume knowledge of common prior

- Needed to compute $\Pr[\tilde{x}_j \mid \tilde{x}_i]$

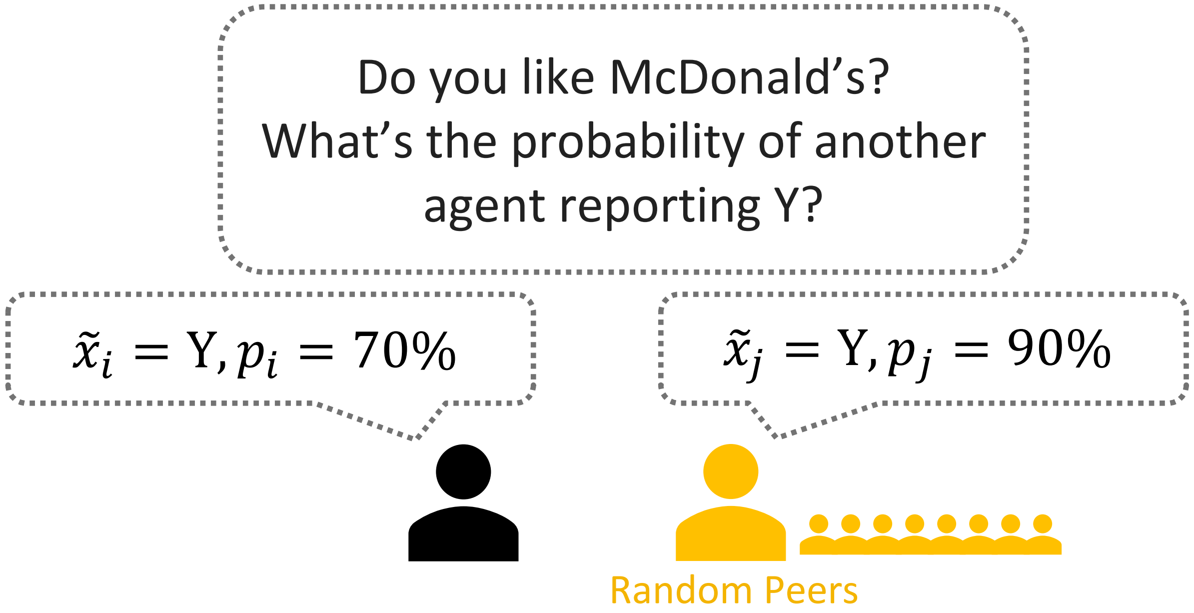
How to Be Prior-independent?

- Multi-task setting [Dasgupta and Ghosh, 2013, Kong and Schoenebeck, 2019, Schoenebeck and Yu, 2020, Shnayder, Agarwal, Frongillo, and Parke, 2016, …]
  - Learn the correlation between reports



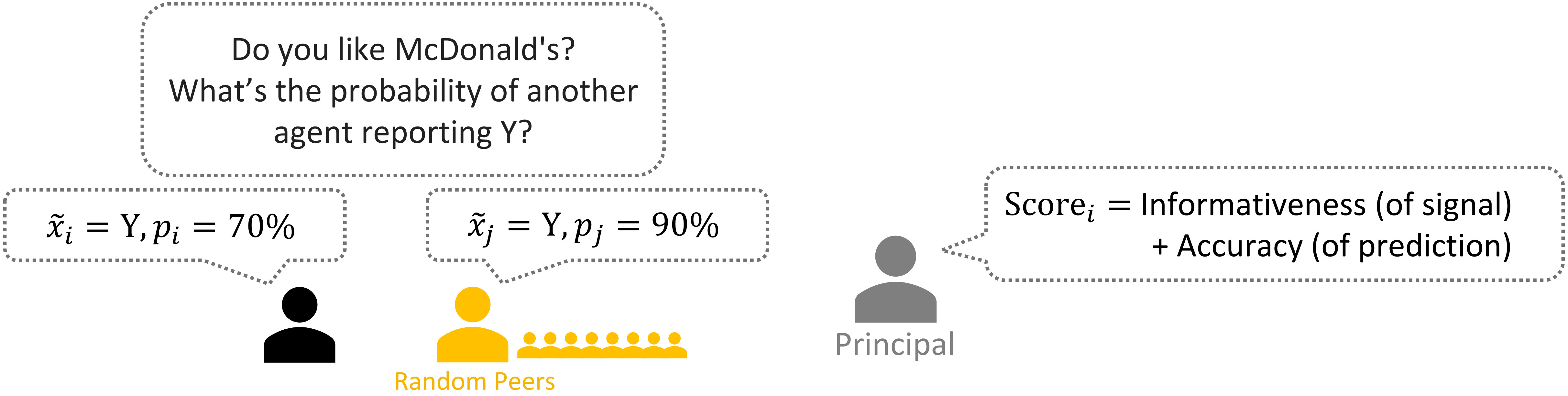
How to Be Prior-independent?

- Bayesian Truth Serum [Prelec, 2004]
  - Directly elicit the prediction $\Pr[\,\cdot \mid \tilde{x}_i]$
  - Follow-up works [Radanovic and Faltings, 2013, Schoenebeck and Yu, 2023, Witkowski and Parkes, 2012, Zhang and Chen, 2014, …]



How to Be Prior-independent?

- Bayesian Truth Serum [Prelec, 2004]
  - Directly elicit the prediction $\Pr[\cdot \mid \tilde{x}_i]$
  - Follow-up works [Radanovic and Faltings, 2013, Schoenebeck and Yu, 2023, Witkowski and Parkes, 2012, Zhang and Chen, 2014, …]



Information Elicitation Mechanism

- Case 1: with verification: PSR ✅
- Case 2: without verification: ✅
  - Knowledge of Common Prior: Peer Prediction
  - Prior Independent (Detail Free): BTS, multi-task PP

Information Elicitation Mechanism

- Case 1: with verification: PSR ✅
- Case 2: without verification: ✅
  - Knowledge of Common Prior: Peer Prediction
  - Prior Independent (Detail Free): BTS, multi-task PP

- Beyond Multiple-choice?

Motivating Example: Which Review is by ChatGPT?

… The paper is engaging and addresses a highly pertinent issue: information elicitation in the context of Large Language Models (LLMs). The concept of computing conditional probability using an LLM is both elegant and innovative. …

… A primary concern is the robustness of the method used to estimate conditional probability with an LLM, which may require additional experimentation and methodological refinement to ensure reliability and applicability across diverse scenarios. …

… The paper presents a novel application of LLMs to enhance peer prediction mechanisms, which is a significant step forward from traditional methods that focus on simpler report types. …

… While the mechanisms are theoretically sound, their practical implementation, especially in real-world settings with diverse and complex textual inputs, might pose significant challenges. …

Which Review is by ChatGPT?

… The paper is engaging and addresses a highly pertinent issue: information elicitation in the context of Large Language Models (LLMs). The concept of computing conditional probability using an LLM is both elegant and innovative. …

… A primary concern is the robustness of the method used to estimate conditional probability with an LLM, which may require additional experimentation and methodological refinement to ensure reliability and applicability across diverse scenarios. …

… The paper presents a novel application of LLMs to enhance peer prediction mechanisms, which is a significant step forward from traditional methods that focus on simpler report types. …

… While the mechanisms are theoretically sound, their practical implementation, especially in real-world settings with diverse and complex textual inputs, might pose significant challenges. …

Human Review v.s. GPT Review

… The paper is engaging and addresses a highly pertinent issue: information elicitation in the context of Large Language Models (LLMs). The concept of computing conditional probability using an LLM is both elegant and innovative. …

… A primary concern is the robustness of the method used to estimate conditional probability with an LLM, which may require additional experimentation and methodological refinement to ensure reliability and applicability across diverse scenarios. …

Our Reviewer #B

… The paper presents a novel application of LLMs to enhance peer prediction mechanisms, which is a significant step forward from traditional methods that focus on simpler report types. …

… While the mechanisms are theoretically sound, their practical implementation, especially in real-world settings with diverse and complex textual inputs, might pose significant challenges. …

ChatGPT 4o

Eliciting Textual Information

Existing methods are not practical for eliciting textual information

- Case 1: with verification ❓
  - PSR: Require "small" finite signal space (Multiple-choice tasks)
  - Even checking agreement between textual report and ground truth can be hard

Eliciting Textual Information

Existing methods are not practical for eliciting textual information

- Case 1: with verification ❓
  - PSR: Require "small" finite signal space (Multiple-choice tasks)
  - Even checking agreement between textual report and ground truth can be hard
- Case 2: without verification ❓
  - Original Peer Prediction: Requires knowledge of the prior
  - Multi-task / BTS: Require "small" finite signal space (Multiple-choice tasks)

Eliciting Textual Information

Existing methods are not practical for eliciting textual information

- Case 1: with verification ❓
  - PSR: Require "small" finite signal space (Multiple-choice tasks)
  - Even checking agreement between textual report and ground truth can be hard
- Case 2: without verification ❓
  - Original Peer Prediction: Requires knowledge of the prior
  - Multi-task / BTS: Require "small" finite signal space (Multiple-choice tasks)
- Signal space of textual information is too large

Eliciting Textual Information with LLMs

How can LLM help with these practical challenges?

Need for “small” finite space =>

- Text Embedding?
- Dimension reduction?

Need for knowledge of the prior =>

- Use LLMs to estimate the prior?

CONTENT

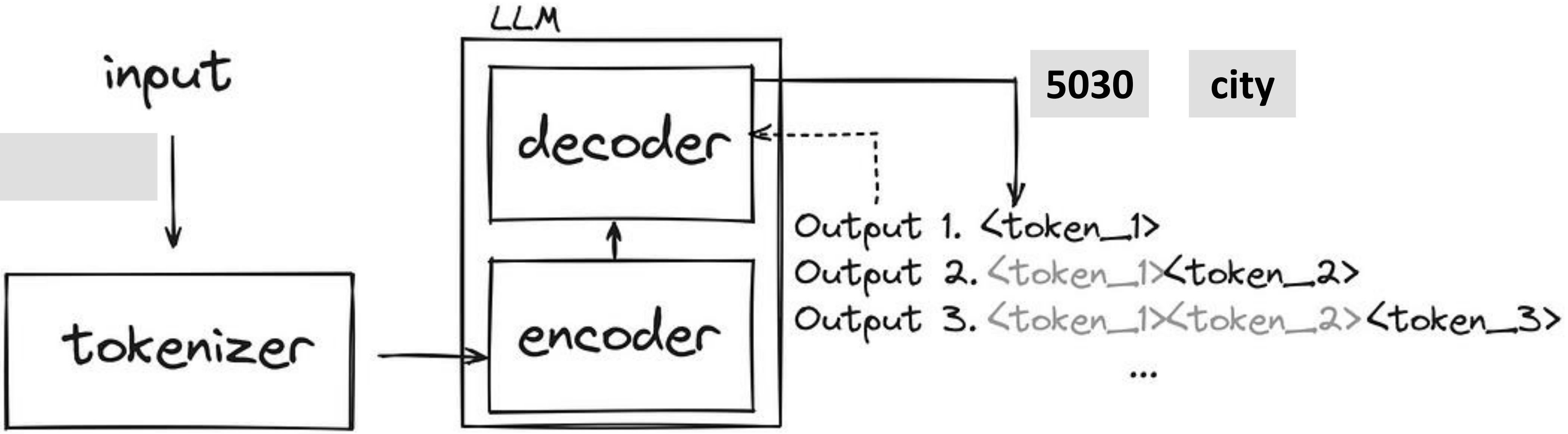
Large Language Models

- A large language model is like a complex **automaton** designed to understand and generate human language, processing vast amounts of text data to simulate conversation and comprehend context.

LLMs: Tokens

- Text generation overview:

Paris is the



5030 city

Paris is the

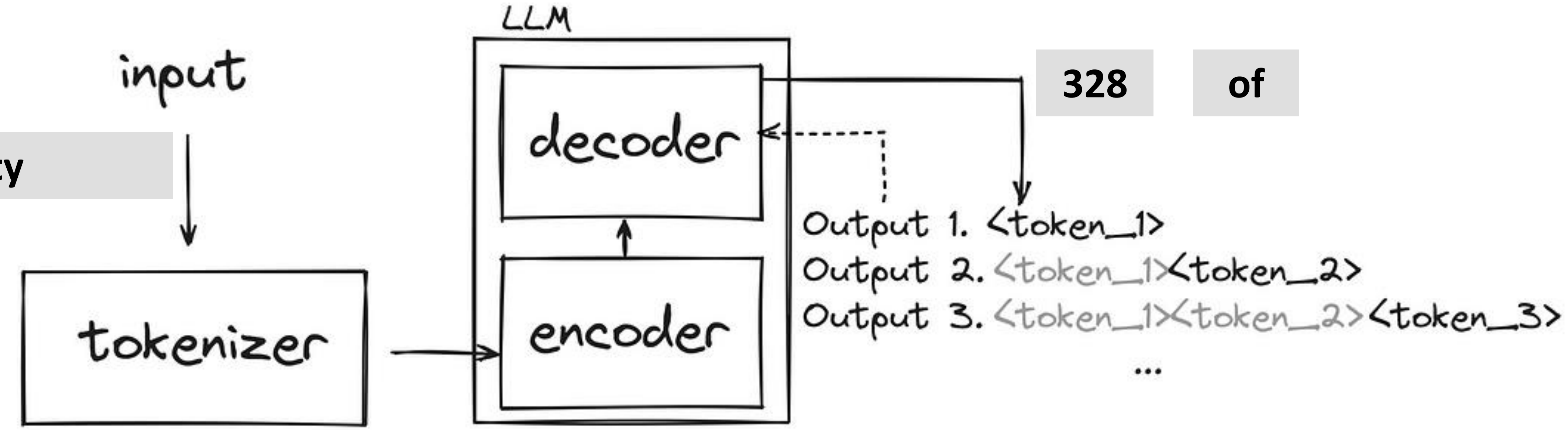
[72782, 382, 290]

How Does an LLM Generate Text? Ivan Reznikov
https://pub.towardsai.net/how-does-an-llm-generate-text-fd9c57781217

LLMs: Tokens

- Text generation overview:

Paris is the city



328

of

Paris is the city

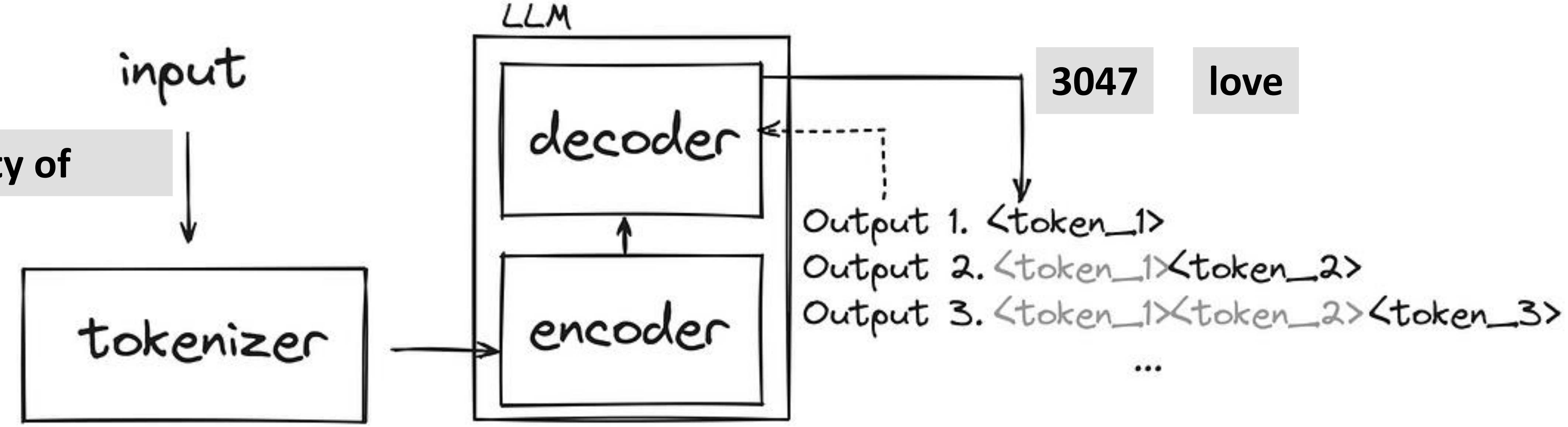
[72782, 382, 290, 5030]

How Does an LLM Generate Text? Ivan Reznikov
https://pub.towardsai.net/how-does-an-llm-generate-text-fd9c57781217

LLMs: Tokens

- Text generation overview:

Paris is the city of



3047

love

Paris is the city of

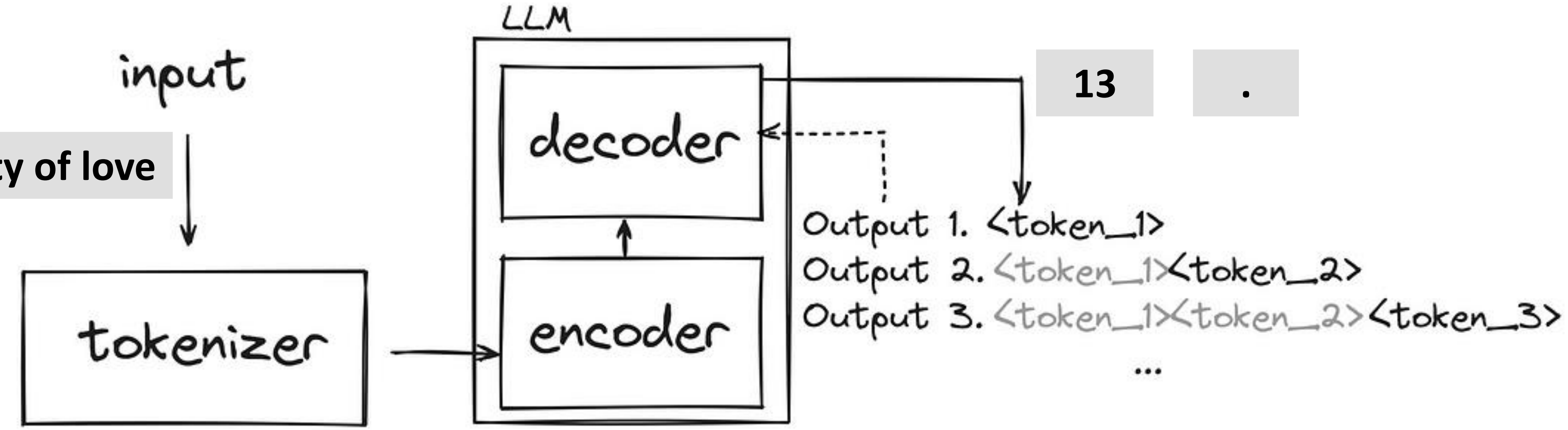
[72782, 382, 290, 5030, 328]

How Does an LLM Generate Text? Ivan Reznikov
https://pub.towardsai.net/how-does-an-llm-generate-text-fd9c57781217

LLMs: Tokens

- Text generation overview:

Paris is the city of love



13 .

Paris is the city of love

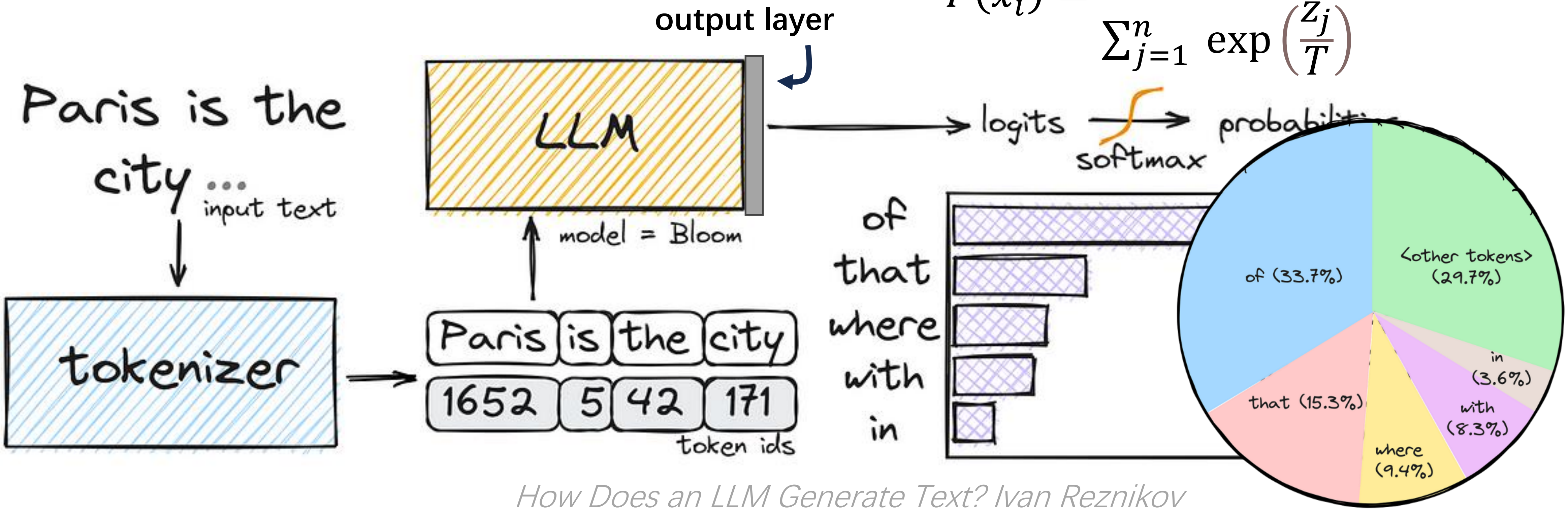
[72782, 382, 290, 5030, 328, 3047]

How Does an LLM Generate Text? Ivan Reznikov
https://pub.towardsai.net/how-does-an-llm-generate-text-fd9c57781217

LLMs: Generating Tokens

- How does an LLM generate each new token?

$$P(x_i) = \frac{\exp\left(\frac{z_i}{T}\right)}{\sum_{j=1}^{n} \exp\left(\frac{z_j}{T}\right)}$$



How Does an LLM Generate Text? Ivan Reznikov
https://pub.towardsai.net/how-does-an-llm-generate-text-fd9c57781217

LLMs: Logits

- Logits are good tools when designing mechanisms!
  - Easy to access: as long as we can run the LLM locally
  - From text to number: numbers are more tractable than texts

- We can obtain **embedding** or **log probabilities** from logits

LLMs: Sentence Embedding

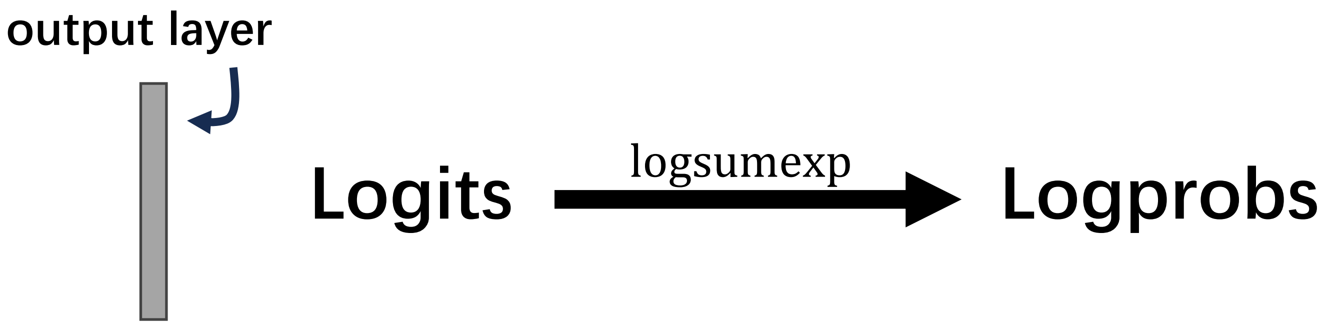
- Some LLMs are fine-tuned to focus on embedding.
- The logits of these LLMs are aggregated to create a comprehensive vector representation of the entire sentence.
  - (often by averaging or using the [CLS] token)
- Sentence embeddings provide a compact and efficient way to represent the semantic meaning of sentences.

LLMs: Logprobs

- For LLMs that focus on generating text, normalizing the logits allows us to obtain log probabilities.

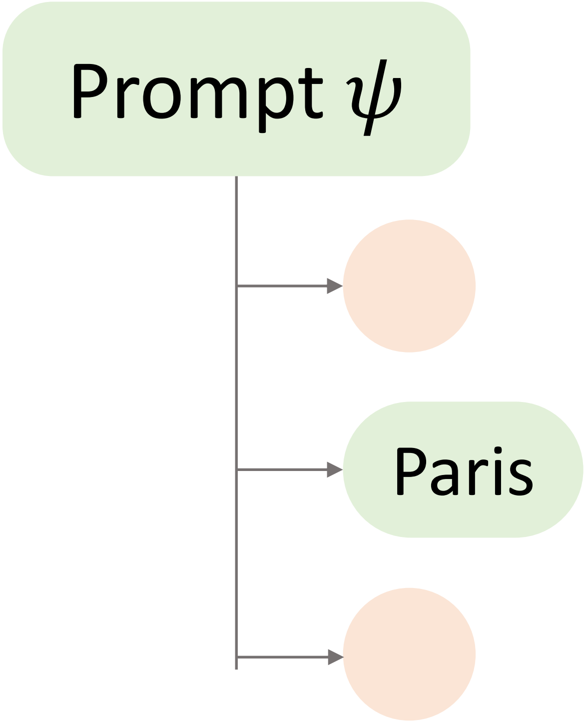
- Theoretical properties: **estimate the conditional probability**

$$\Pr[\text{output} \mid \text{input}]$$



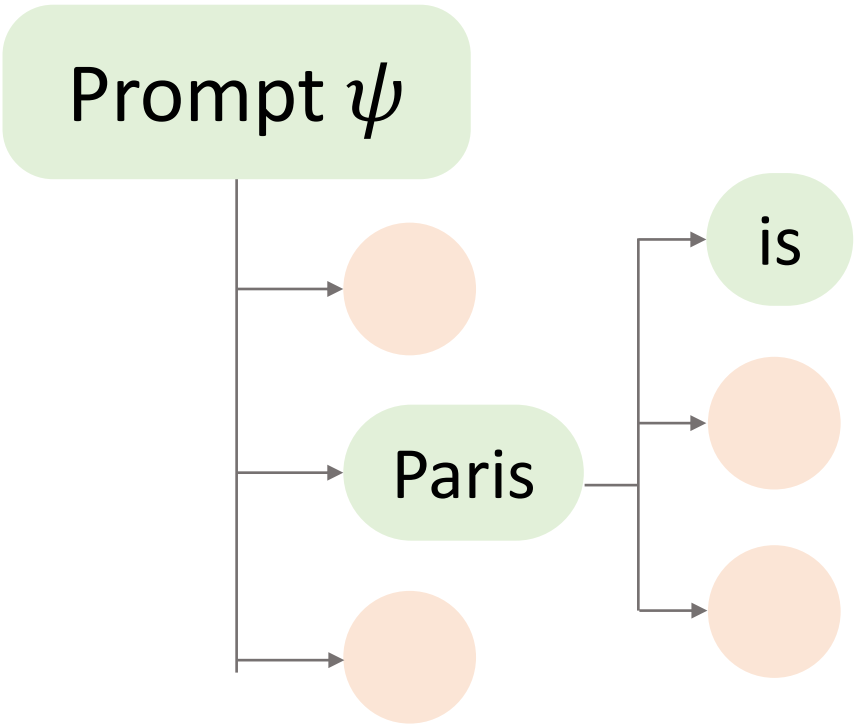
LLMs: Logits and Logprobs

- ψ = “What is Paris known as?”
- Pr[next token = “Paris” | Prefix = “”, Prompt = ψ]



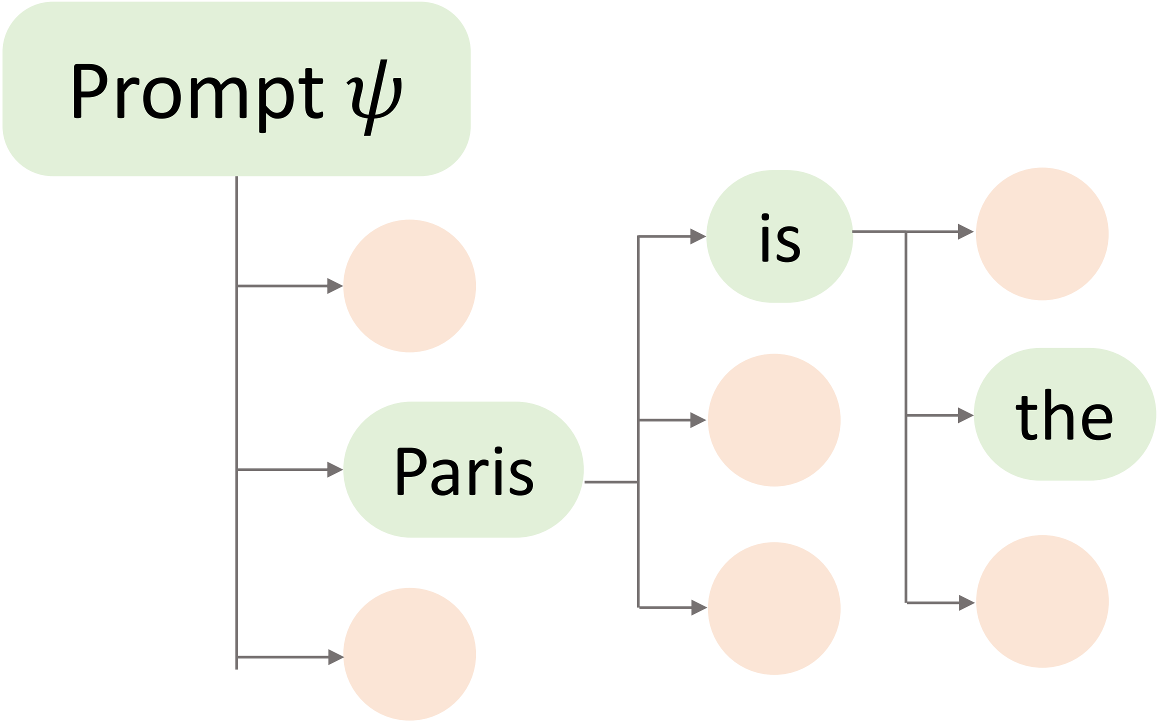
LLMs: Logits and Logprobs

- ψ = "What is Paris known as?"
- Pr[next token = "_is" | Prefix = "Paris", Prompt = ψ]



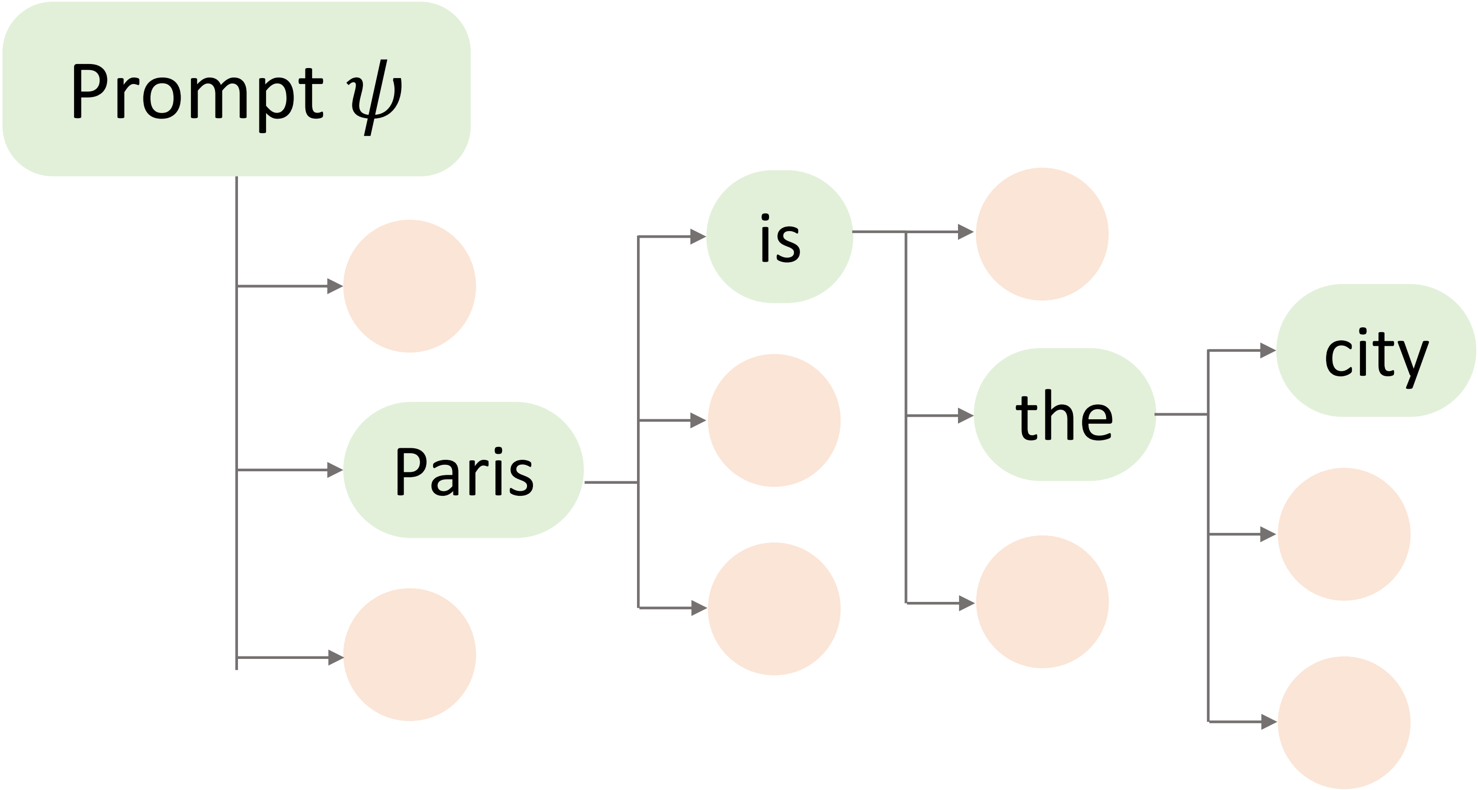
LLMs: Logits and Logprobs

- ψ = “What is Paris known as?”
- Pr[next token = “_the” | Prefix = “Paris is”, Prompt = ψ]



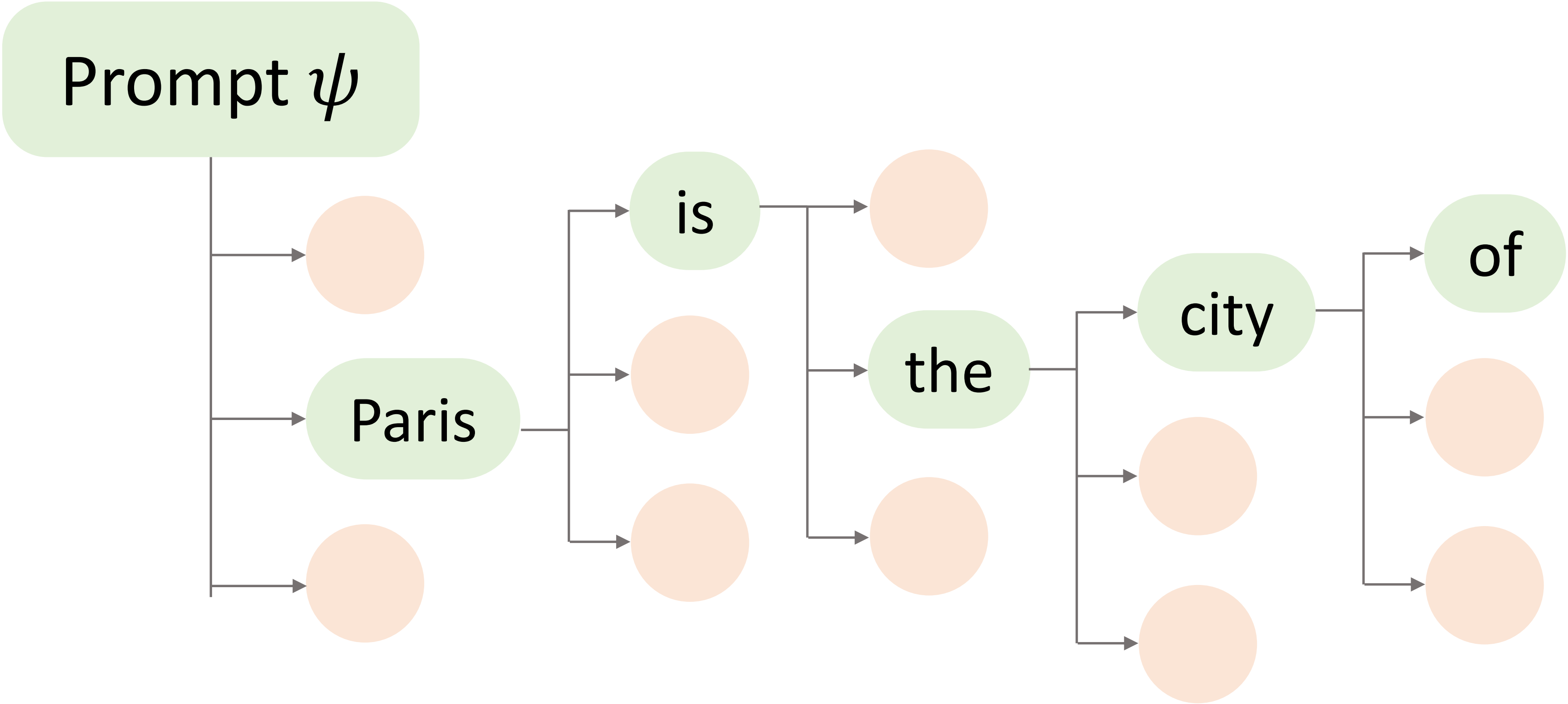
LLMs: Logits and Logprobs

- ψ = “What is Paris known as?”
- Pr[next token = “_city” | Prefix = “Paris is the”, Prompt = ψ]



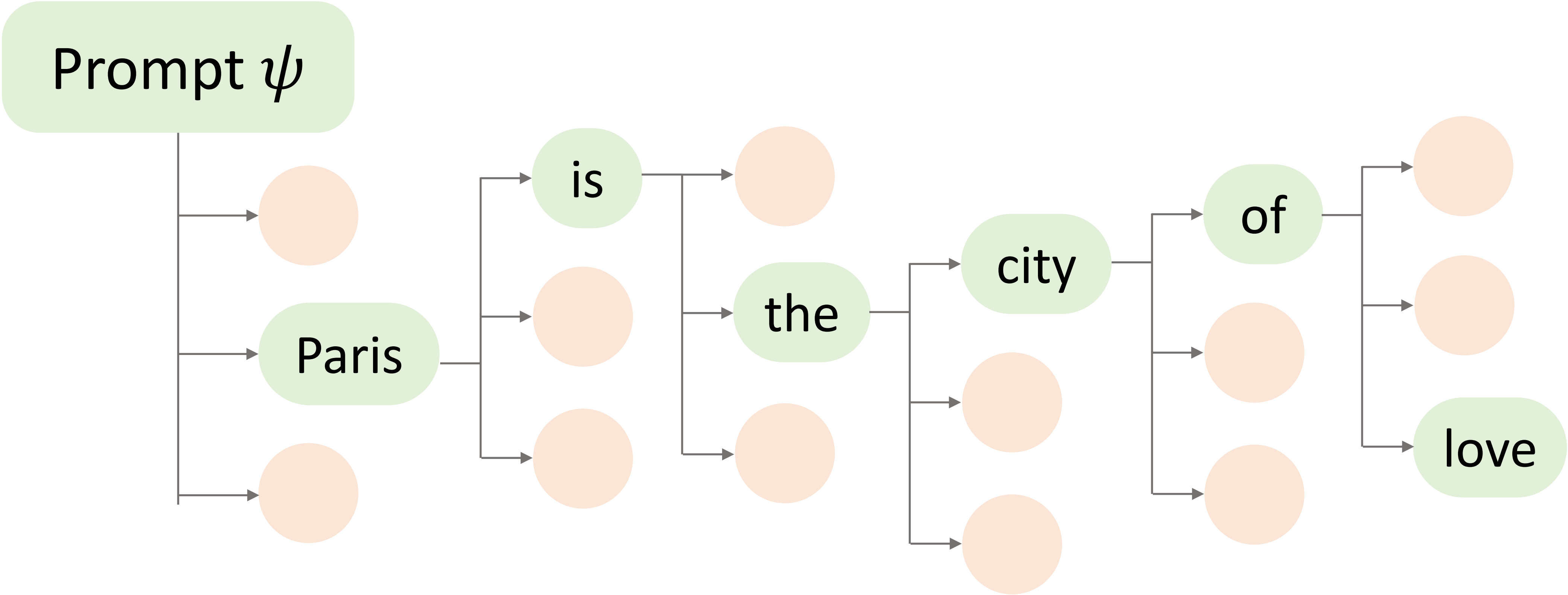
LLMs: Logits and Logprobs

- ψ = "What is Paris known as?"
- Pr[next token = "_of" | Prefix = "Paris is the city", Prompt = ψ]



LLMs: Logits and Logprobs

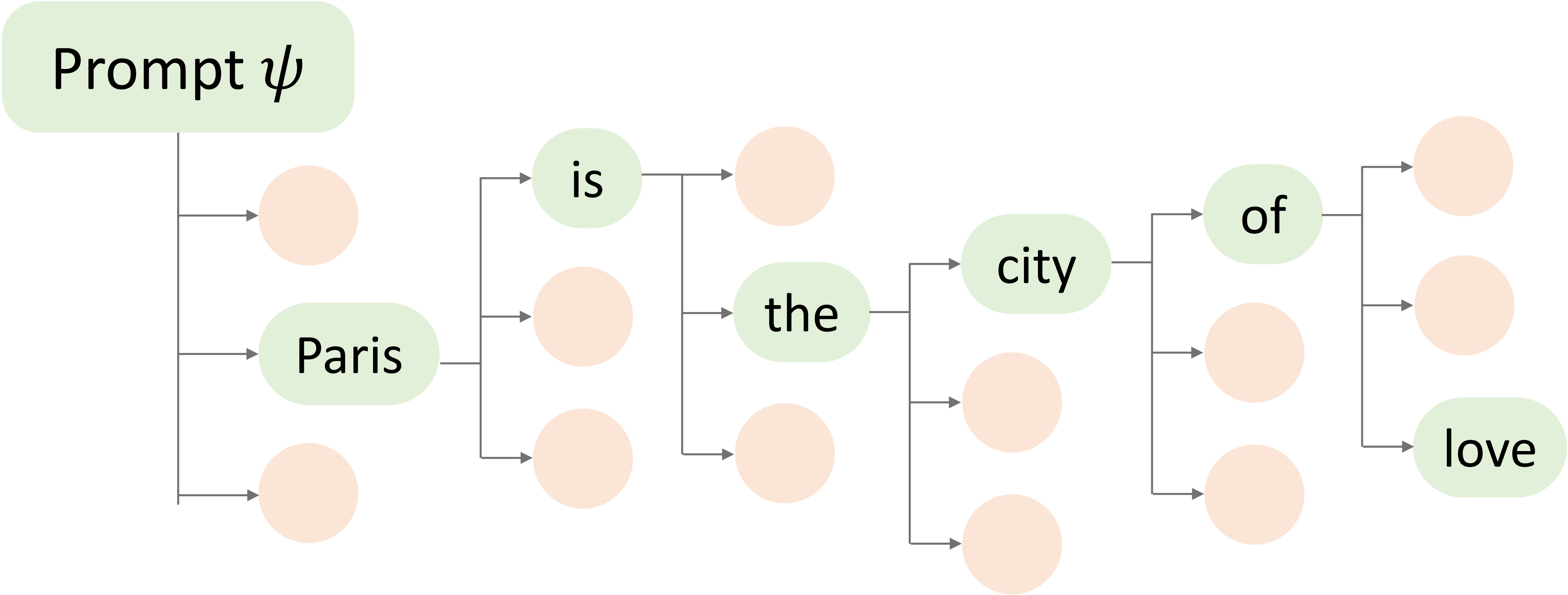
- ψ = "What is Paris known as?"
- Pr[next token = "_love" | Prefix = "Paris is the city of", Prompt = ψ]



LLMs: Logits and Logprobs

Multiply these together,

we have Pr[output = "Paris is the city of love" | Prompt = ψ]



CONTENT

Elicitation: Beyond the Boundary!

- Can we elicit textual information with the help of LLMs?
  - Yes!

- Elicit textual information through

High-dimensional Scoring Rules	Hartline and Wu, 24
Generative Peer Prediction	**Lu**, **Xu**, Zhang, Kong, and Schoenebeck, EC'24

Question: Elicit Text **with** Ground Truth

- Q: How to elicit truthful report when we have ground truth?
- A: Proper Scoring Rules

- Q: How to elicit truthful textual report when we have textual ground truth?

Running Example

- Score to elicit reviews in **peer grading**

Peer Rreview A

The submission seems wrong, but I don't know which part is wrong.

Peer Review B

The statement can't be *proven by an example*, which might not be true for all cases.

compare with

Ground Truth (Instructor Review)

Proof by example is not sufficient.

- Target: score textual review based on instructor review

textual report

textual ground truth

Empirical Validation

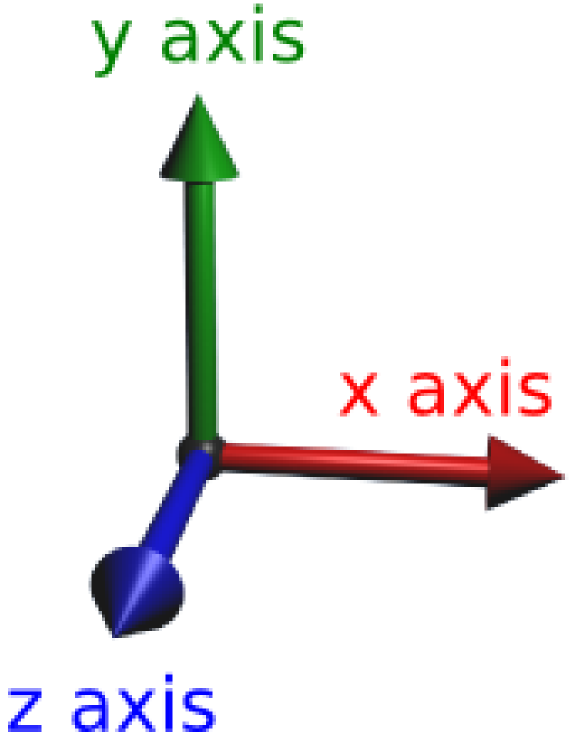
- **Dataset**: Peer Grading from 3 classes. Each dataset has
  - $\sim$ 10 assignments, $\sim$ 5 submissions / assignment.
  - Each submission: $\sim$ 5 peer reviews, 1 instructor review.
  - human preference
    - instructor score of peer review quality
    - Students’ final grades (avg over exams, homework, peer grading, etc.)

Empirical Validation

- **Dataset**: Peer Grading from 3 classes.
- **Metric**: Spearman's rank correlation in $[-1, 1]$. Correlation between student rankings.
  - 0 for no correlation;
  - 1 for perfect correlation, -1 for perfect negative correlation;
  - ≥ 0.6 high correlation, ≥ 0.8 very strong.
- **Validation**: If the ranking induced by the mechanism aligns with peer's final grades, it is an evidence of its effectiveness.

Main Idea

- To break down the textual report to several dimensions,
  - and then try some "**responsive**" scoring rules.



Theoretical Basis: Setting

- Ground truth: m-dimensional state for opinions.
  - $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_m)$
  - $\theta_i \in \Theta = \{0,1\}, 1 =$ positive, $0 =$ negative
  - e.g. θ_1 for overall correctness, θ_2 for using examples as proof, etc.
- Agent holds belief $D \in \Delta(\Theta)$
- He reports the marginals $r \in R = [0,1]^m$.
- Principal reveals $\boldsymbol{\theta}$.
- Agent receives score S: $R \times \Theta \to [0,1]$.

Theoretical Basis: PSR In This Setting

- A scoring rule is proper if for any belief distribution $\hat{D} \in \Delta(\Theta)$,
- $\hat{r} \in [0,1]^m$ is the marginal means of $\hat{D}$,
- $\mathbf{E}_{\theta \sim \hat{D}}[S(\hat{r}, \theta)] \geq \mathbf{E}_{\theta \sim \hat{D}}[S(r, \theta)]$
- for any deviation $r \in [0,1]^m$.

- E.g., average quadratic scoring rule
  - $S(\boldsymbol{r}, \boldsymbol{\theta}) = \frac{1}{m} \sum_{i \in [m]} 1 - (r_i - \theta_i)^2$

ElicitationGPT: First Thought

- Ground truth: m-dimensional state for opinions.
  - $\boldsymbol{\theta} = (\theta_1, \theta_2, \dots, \theta_m)$
  - $\theta_i \in \Theta = \{0,1\}, 1 =$ positive, $0 =$ negative
  - e.g. θ_1 for overall correctness, θ_2 for using examples as proof, etc.
- Agent holds belief $D \in \Delta(\Theta)$
- He reports the marginals $r \in R = [0,1]^m$.
- Principal reveals $\boldsymbol{\theta}$.
- Agent receives score S: $R \times \Theta \to [0,1]$.

How to define the states from text?

How to translate text into probabilities?

ElicitationGPT: State & Report

- How to define the states?
  - Identify by split the ground truth text to summary points.
  - e.g. hypothesis, base case, and induction step.

- How to translate text into probabilities?
  - Mapping “I don't know” to the prior (frequency) of each state.
  - Assumption “know-it-or-not”
    - For each dimension i, belief is in $\{0,1,\Pr[\theta_i = 1]\}$, positive/negative/prior.
  - Agent expresses uncertainty by saying “I don't know”.

ElicitationGPT: State & Report

- How to define the states?
  - Identify by split the ground truth text to summary points.
  - e.g. hypothesis, base case, and induction step.

Leveraging LLM to retrieve state and report

- How to translate text into probabilities?
  - Mapping “I don't know” to the prior (frequency) of each state.
  - Assumption “know-it-or-not”
    - For each dimension i, belief is in $\{0,1,\Pr[\theta_i = 1]\}$, positive/negative/prior.
  - Agent expresses uncertainty by saying “I don't know”.

ElicitationGPT: State & Report

Leveraging LLM to retrieve state and report

Summarize the following homework reviews into main points...
Review: g

textual ground truth g ▶ $(g_1, \theta_1), (g_2, \theta_2), (g_3, \theta_3), \ldots$

textual report r ▶

ElicitationGPT: State & Report

Leveraging LLM to retrieve state and report

Does this particular review has a negative or positive opinion on the following statement?
Review: r
Statement: g_1

textual ground truth g → $(g_1, \theta_1), (g_2, \theta_2), (g_3, \theta_3), \ldots$

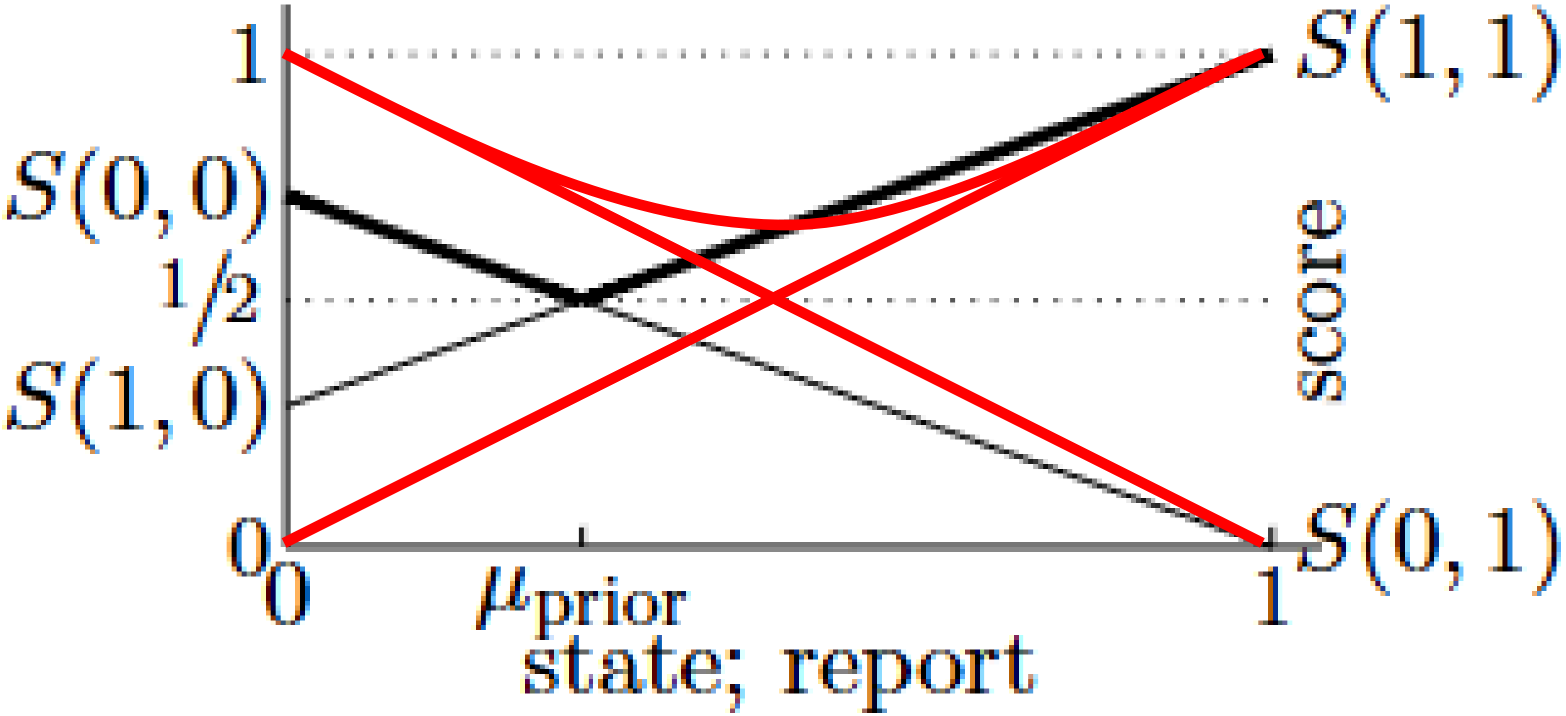
textual report r → r_1, r_2, r_3, …

ElicitationGPT: Process Overview

1. **extracting semantic dimensions of state** Summarization (ground truth texts)
2. **calculating prior**
   Question Answering (ground truth, summarized points)
   Prior: count the frequency of 1's on each state.
3. **mapping reported text to report**
   Question Answering (reported text, summarized points)
   Map $\perp$ to the prior frequency in truth.
4. **proper scoring rule** score S the translated report and the translated ground truth.

ElicitationGPT: Better Scoring Rule

- Since the belief is assumed to only be one of $\{0, p, 1\}$, we can employ a simple and more "**responsive**" scoring rule.

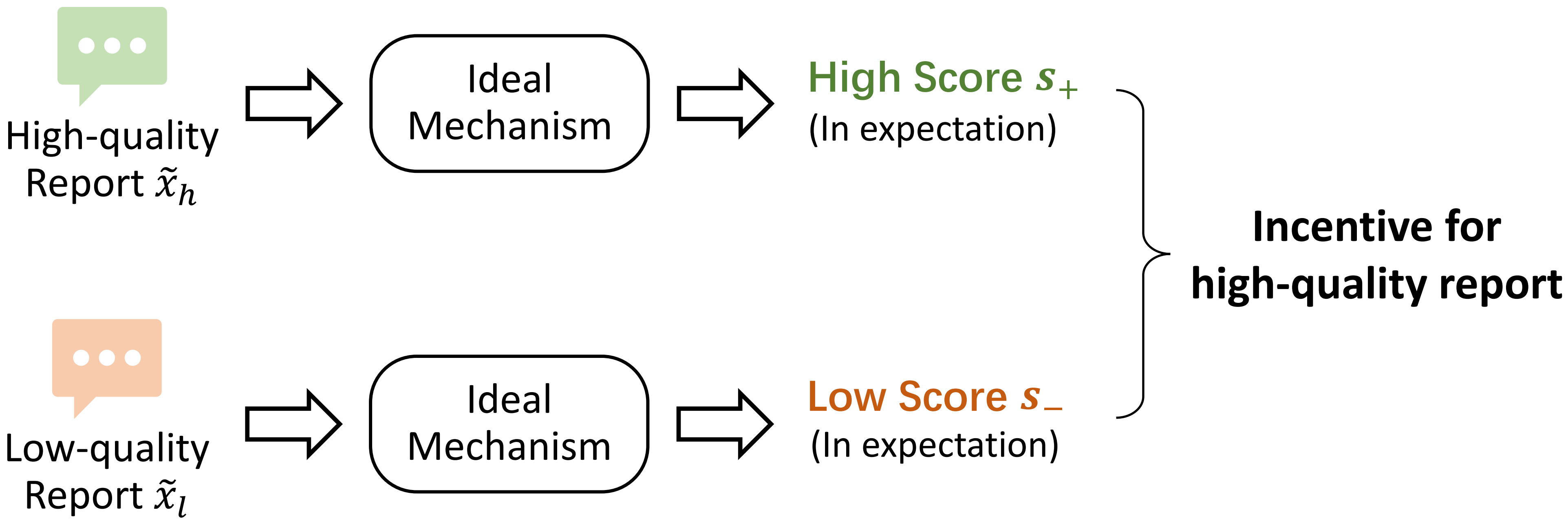


ElicitationGPT: Experiments

- **Dataset**: Peer Grading from 3 classes.
- **Metric**: Spearman's rank correlation in [−1, 1]. Correlation between student rankings.
- **Results**:

Correlation	instructor score	direct GPT	$\text{Elicitation}^{\text{GPT}}$
Algorithm Class 1	0.55	0.58	**0.65**
Algorithm Class 2	0.48	0.46	**0.63**
Mechanism Design	0.47	0.43	**0.59**

Next Question: Elicit Text **without** Ground Truth



Running Example

- Score to elicit reviews in **paper review**

Peer Rreview A

… A primary concern is the robustness of the method used to estimate conditional probability with an LLM, which may require additional experimentation and methodological refinement to ensure reliability and applicability across diverse scenarios. …

Peer Review B

… While the mechanisms are theoretically sound, their practical implementation, especially in real-world settings with diverse and complex textual inputs, might pose significant challenges. …

~~compare with~~

~~**Ground Truth (Instructor Review)**~~

~~Proof by example is not sufficient.~~

- Target: score textual review based on others' reviews

textual report

peers' textual reports

Peer Prediction: Recall

- Agent i's report $\tilde{x}_i \in \Sigma$, agent j's (the peer) report $\tilde{x}_j \in \Sigma$
- Score of agent i $= \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$
  - When applying a log scoring rule

Original Peer Prediction Mechanism

$\tilde{x}_i \backslash \tilde{x}_j$	$\boldsymbol{Y}$	$\boldsymbol{N}$
$\boldsymbol{Y}$	$\mathbf{1/3}$	$\mathbf{1/6}$
$\boldsymbol{N}$	$\mathbf{1/6}$	$\mathbf{1/3}$

Assume knowledge of common prior

Generative Peer Prediction

- Agent i's report $\tilde{x}_i \in \Sigma$, agent j's (the peer) report $\tilde{x}_j \in \Sigma$
- Score of agent i $= \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$
  - When applying a log scoring rule

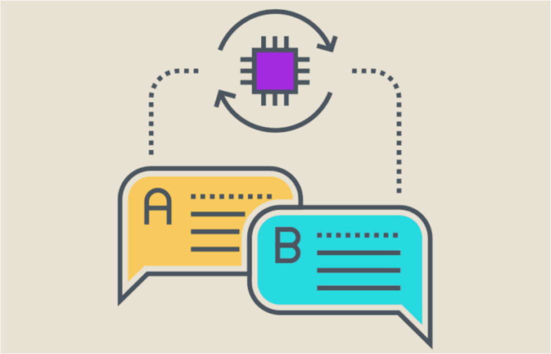
Leverage LLM to estimate

Original Peer Prediction Mechanism

$\tilde{x}_i$ \ $\tilde{x}_j$	Y	N
Y	$1/3$	$1/6$
N	$1/6$	$1/3$

Assume common knowledge of common prior

Generative Peer Prediction Mechanism



Use LLM to estimate the underlying prior

More Than Truthfulness

- We also want agents to take effort.
  - Obtaining the signal is costly:
    - Paper reading, proof checking, assessment formulating, etc.

Reviewer #2:

The paper addresses an interesting problem. The writing is clear and concise. The organization is logical. Results appear correct. Overall, the study is well-conducted. This work adds value to the field. Overall, I think this paper worths acceptance.

Score: Weak Accept

GPP: Theoretical Basis

Need for a good estimation

Theorem (Informal):

- When the KL-divergence between the real distribution $\log \Pr[x_j \mid x_i]$ and the LLM estimated $\log \Pr_{\text{LLM}}[x_j \mid x_i]$ can be **bounded** by ϵ
  - And this distribution is common knowledge for all agents
- **Exerting effort & reporting truthfully is $\alpha\epsilon$-Nash equilibrium**
  - α depends on the cost of effort
  - When ignoring the cost of effort, truthful reporting is ϵ-Nash equilibrium

GPP: First Thought

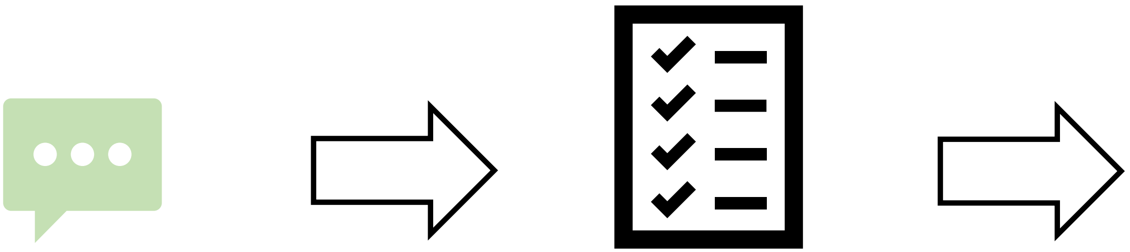
- Agent i's report $\tilde{x}_i \in \Sigma$, agent j's (the peer) report $\tilde{x}_j \in \Sigma$
- Score of agent i $= \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$
  - When applying a log scoring rule

Leverage LLM to estimate

- First Thought: We want the conditional probability, why don't we design some heuristics and ask LLM to answer?
  - Semantic similarity, support or contradict, etc.

GPP-judgment

- GPP-Judgment uses LLMs as an oracle!
  - Only API calls, no need for local (open-source) model
    - Can profit from powerful commercial models (GPT-4o, Claude-3.5, etc.)



$$\Pr[\tilde{x}_j \mid \tilde{x}_i] \approx \prod \Pr[w \in \tilde{x}_j \mid \tilde{x}_i]$$

Report
$\tilde{x}_j$

“Independent” Judgments
$w_1 \ldots w_m$

GPP-judgment

- GPP-Judgment uses LLMs as an oracle!
  - Only API calls, no need for local (open-source) model
    - Can profit from powerful commercial models (GPT-4o, Claude-3.5, etc.)

- Weakness:
  - It is difficult to say how accurate this heuristic is.
  - Requires extensive prompt engineering for each different task.

GPP: Second Thought

- Agent i's report $\tilde{x}_i \in \Sigma$, agent j's (the peer) report $\tilde{x}_j \in \Sigma$
- Score of agent i $= \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$
  - When applying a log scoring rule

Leverage LLM to estimate

- Second Thought: We want the conditional probability, why don't we try pure Logprobs

GPP: Second Thought

- Agent i's report $\tilde{x}_i \in \Sigma$, agent j's (the peer) report $\tilde{x}_j \in \Sigma$
- Score of agent i $= \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$
  - When applying a log scoring rule

Leverage LLM to estimate

- We integrate $\tilde{x}_i$ in the prompt ψ and then force the LLM to generate $\tilde{x}_j$
- Use the probability of generating $\tilde{x}_j$ as an estimation of $\log \Pr[\tilde{x}_j \mid \tilde{x}_i]$

GPP: Second Thought

- Agent i's report $\tilde{x}_i \in \Sigma$, agent j's (the peer) report $\tilde{x}_j \in \Sigma$
- Score of agent i $= \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$
  - When applying a log scoring rule

Leverage LLM to estimate

Prompt $\psi(\tilde{x}_i)$**:** You are the second reviewer for a scientific paper. You are given a peer review from the other reviewer: **[Review** $\tilde{x}_i$**]** Your task is to provide your own judgments of the paper based on the given materials.

Response: [Predicted Review $\tilde{x}_j$]

Logprob $= \log \Pr_{\mathrm{LLM}(\psi)}[X_j = \tilde{x}_j \mid X_i = \tilde{x}_i]$

GPP: Empirical Validation

- **Dataset**: ICLR 2020 Peer Review
  - accessed via the OpenReview API
  - Randomly select 300 papers, 911 peer reviews

GPP: Empirical Validation

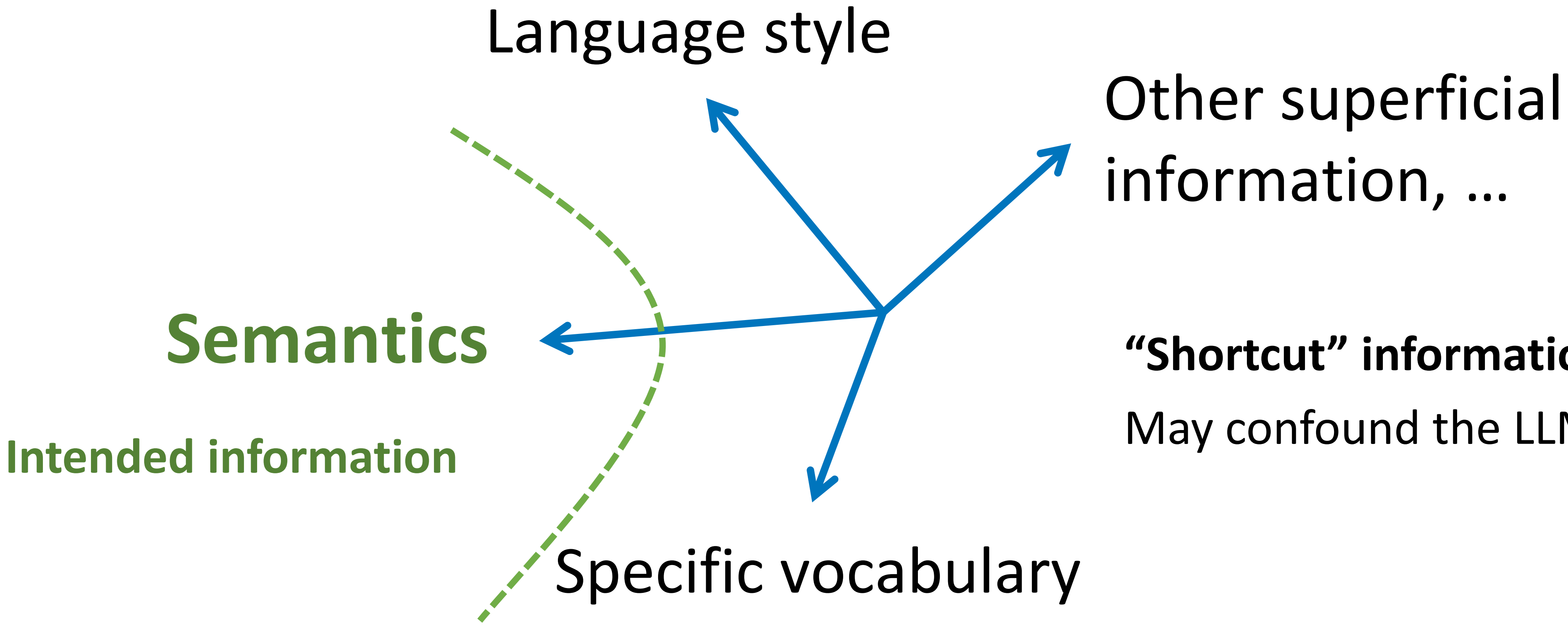
- **Dataset**: ICLR 2020 Peer Review
- **Metric**: Comparison between the scores of LLM generated review and human written review.
  - The reviews generated by LLM are often considered worse than those written by humans.
- **Validation**: If the mechanism can distinguish them, it is an evidence of its effectiveness.

LLM is not **That** Powerful…

- This Attempt was not a success.
    - Pure Logprobs are very unstable
    - Sometimes Logprobs goes to –inf

LLM prediction may be influenced by …

- Textual responses are high-dimensional



"Shortcut" information

May confound the LLM predictions

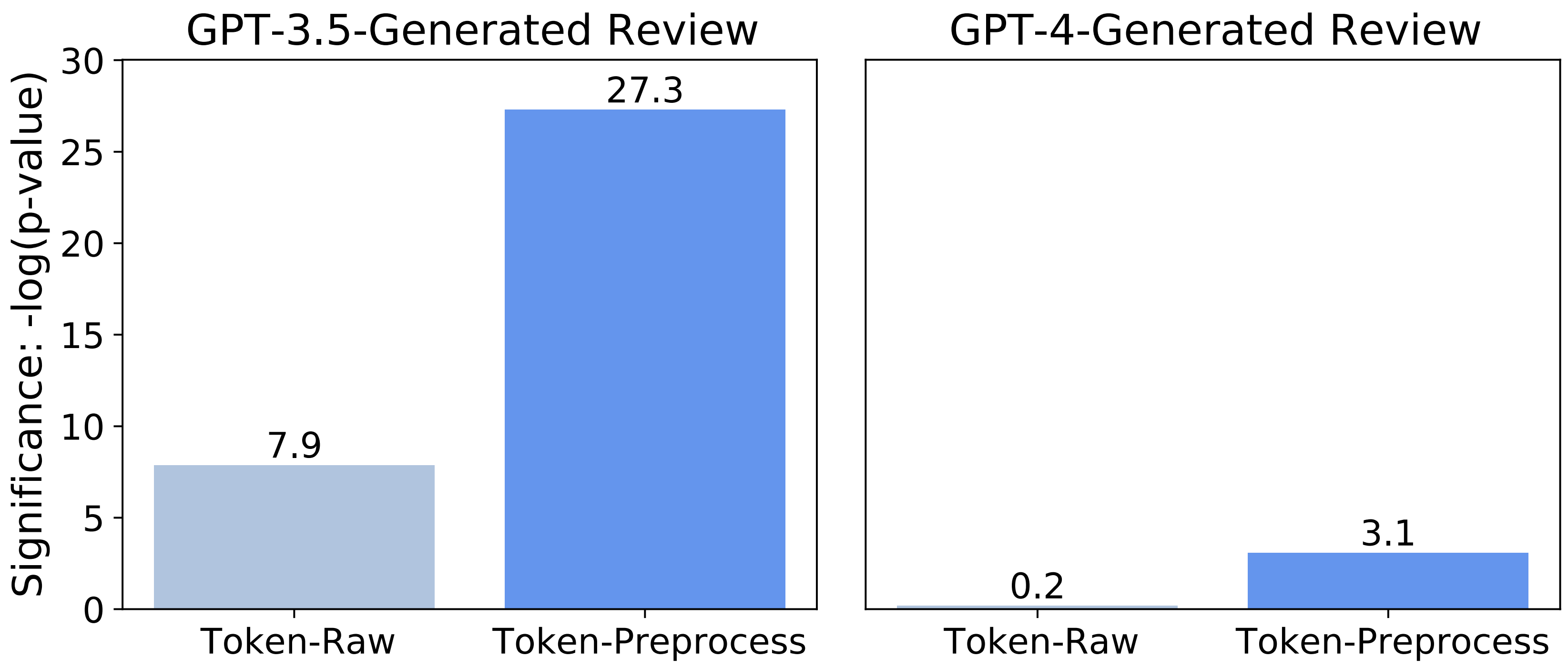
Filter out the shortcut information

Preprocessing

- Use an LLM to rephrase the text reports into a pre-set format
  - Standardize language style
    - including vocabulary use, sentence structure, and grammatical errors
  - Remove superficial information
    - such as a summary of the paper in peer review
- This leads to **GPP-token**

Necessity of Preprocessing

- GPP-token(raw) can not differentiate GPT-4-generated reviews and human-written reviews. But GPP-token can.



Filter out the shortcut information

Preprocessing

- Use an LLM to rephrase the text reports into a pre-set format
  - Standardize language style and remove superficial information
- Conditioning out a “synopsis”
  - Adding a summary of the item in the prompt
  - Generative Synopsis Peer Prediction Mechanism (GSPP)

Generative Synopsis Peer Prediction (GSPP)

- When there is a commonly known synopsis θ of the item
  - E.g. the abstract of the paper
  - Agents may construct low-quality reports solely based on the synopsis

- By conditioning out the "synopsis"
  - We only reward the information beyond the synopsis
- GSPP: Score of agent i $= \log \Pr[\tilde{x}_j \mid \tilde{x}_i, \theta]$

GSPP: Interpret the Score

(Informal) in GSPPM, the expected score of agent i is

Synopsis

$$-H\left(X_j \,\middle|\, X_i, \Theta\right) = \mathbf{I}\left(\boldsymbol{X_i}; \boldsymbol{X_j} \mid \boldsymbol{\Theta}\right) - H(X_j \mid \Theta)$$

Peer's report Agent i's report

Constant from agent i's view

By conditioning out the "synopsis"

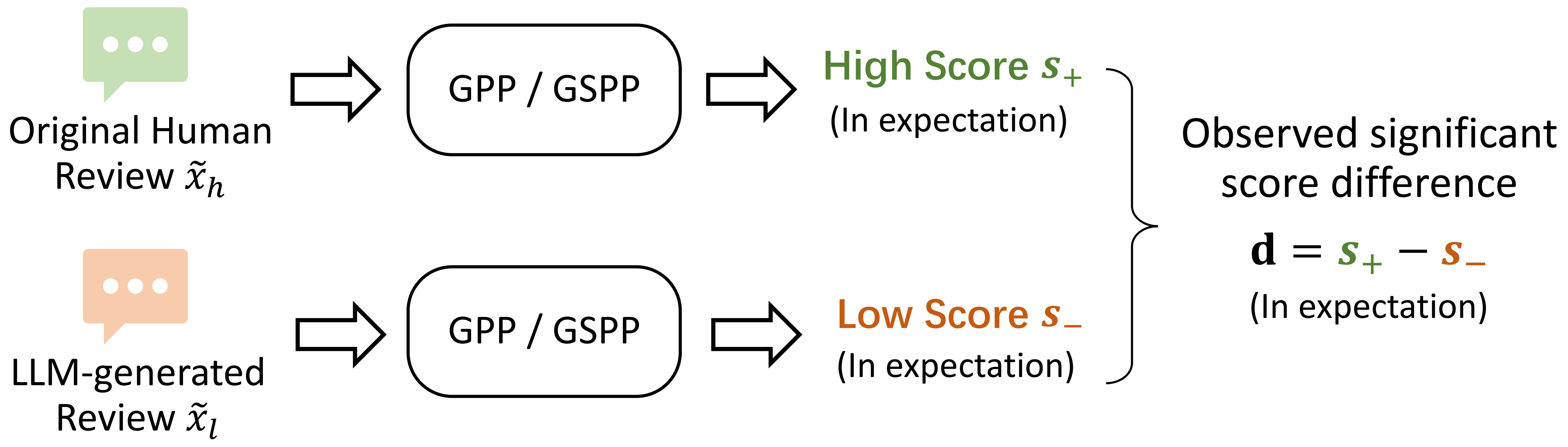
- We only reward the information beyond the synopsis

GPP: Empirical Validation (Recall)

- **Dataset**: ICLR 2020 Peer Review
- **Metric**: Comparison between the scores of LLM generated review and human written review.
  - The reviews generated by LLM are often considered worse than those written by humans.
- **Validation**: If the mechanism can distinguish them, it is an evidence of its effectiveness.

Evaluation Method

- Use paired difference t-test to test $\mathbb{E}[\boldsymbol{s}_+ - \boldsymbol{s}_-] > \mathbf{0}$
  - Score difference between human-written review and LLM-generated review



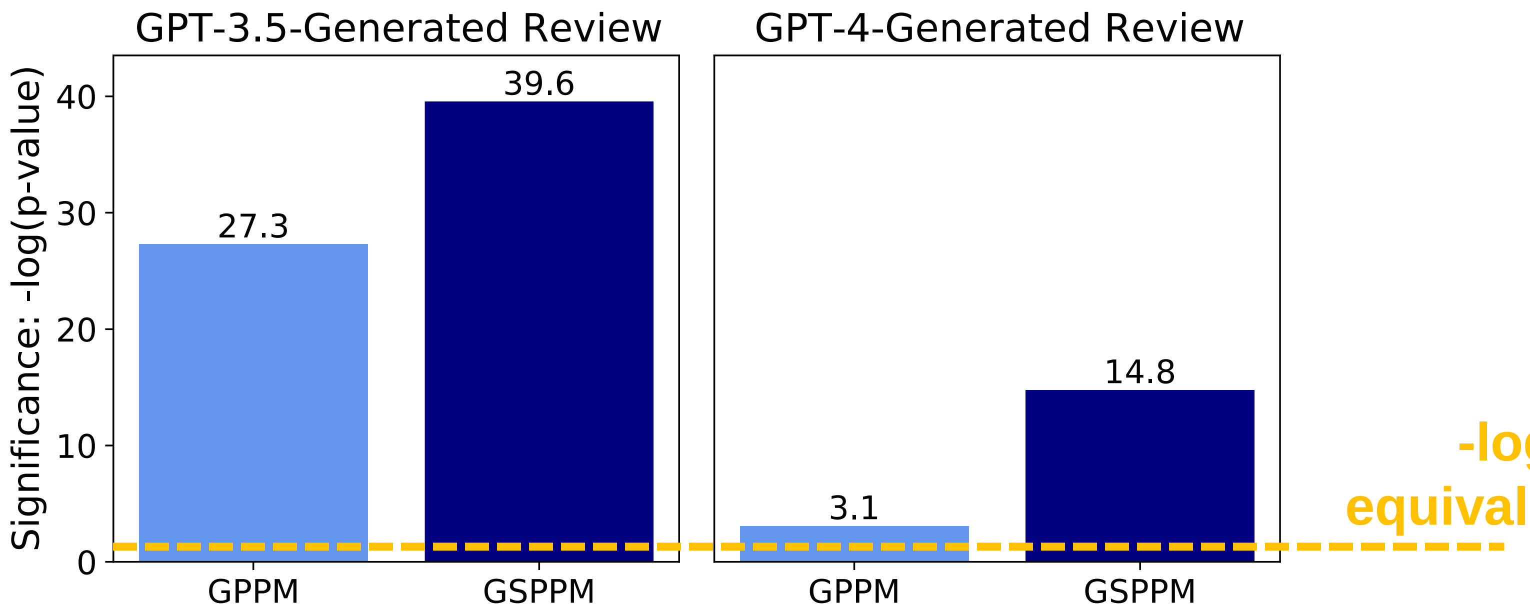
Evaluation Method

- Use paired difference t-test to test $\mathbb{E}[\boldsymbol{s}_+ - \boldsymbol{s}_-] > \mathbf{0}$
  - Score difference between human-written review and LLM-generated review.

- **-log (p-value)** as a metric for significance of $\mathbb{E}[\boldsymbol{s}_+ - \boldsymbol{s}_-] > \mathbf{0}$
  - between human-written review and GPT-generated review.
  - Higher is more significant (equivalent to lower p-value)
  - -log (p-value) > 1.3 is equivalent to p-value < 0.05

Results: human review vs. LLM review

- GPP / **GSPP** can effectively penalize LLM-generated review
- **GSPP** have a higher significance in penalizing LLM-generated reviews
  - **-log (p-value)** of the expected score difference $\mathbb{E}[\tilde{\boldsymbol{s}}_{+} - \tilde{\boldsymbol{s}}_{-}] > \mathbf{0}$, higher is more significant



-log (p-value) > 1.3
equivalent to p-value < 0.05

CONTENT

Information Elicitation Mechanism

Apply to human agents

- Design more suitable mechanisms for preference elicitation tasks (e.g. RLHF) [Chen, Feng, and Yu, 2024]

Apply to LLMs

Use information elicitation mechanisms

- To evaluate/benchmark LLMs [Xu, Lu, Schoenebeck, and Kong, 2024]
- To calibrate LLMs in finetuning [Band, Li, Ma, and Hashimoto, 2024]

Information Elicitation Mechanism

Apply to human agents

- Design more suitable mechanisms for preference elicitation tasks (e.g. RLHF) [Chen, Feng, and Yu, 2024]

Apply to LLMs

Use information elicitation mechanisms

- To evaluate/benchmark LLMs [Xu, Lu, Schoenebeck, and Kong, 2024]
- To calibrate LLMs in finetuning [Band, Li, Ma, and Hashimoto, 2024]

Generalization of GPPM

Elicit textual information without verification

=> Benchmarking LLMs’ judgments without gold-standard reference

[Xu, Lu, Schoenebeck, and Kong, 2024]

Research Question

- Can we use GPPM and GSPPM as **accurate**, **manipulation-resistant**, and **automated** evaluation metrics
- for natural language generation (NLG)
- with no gold standard reference to compare with?

GPPM as an Evaluation Metric

Ideal Evaluation Metric

$$Score_i > Score_p \Leftrightarrow \text{review } \tilde{x}_i \text{ better than } \tilde{x}_p$$

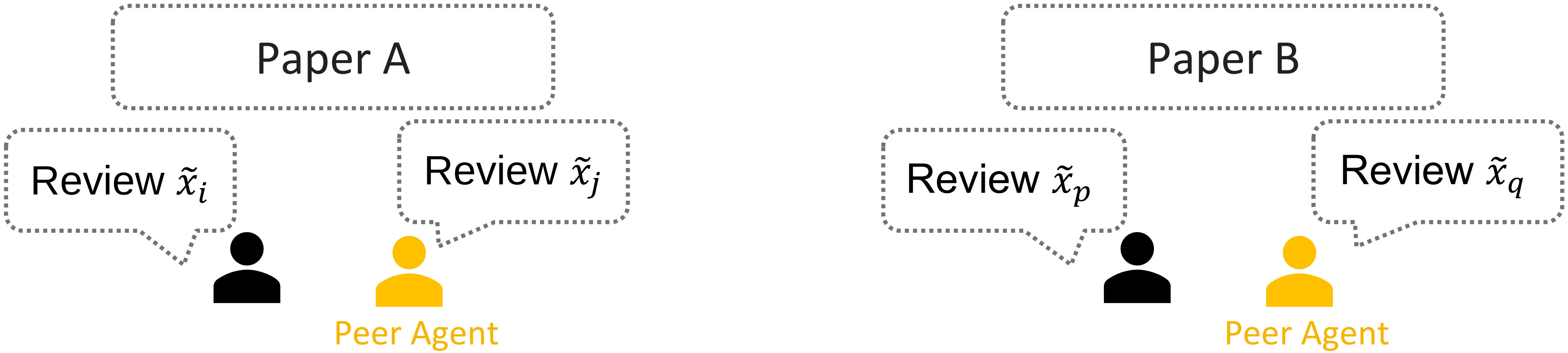
$$Score_i = \log \Pr[\tilde{x}_j \mid \tilde{x}_i]$$

$$Score_p = \log \Pr[\tilde{x}_q \mid \tilde{x}_p]$$

GPPM as an Evaluation Metric

Ideal Evaluation Metric

$$Score_i > Score_p \Leftrightarrow \text{review } \tilde{x}_i \text{ better than } \tilde{x}_p$$



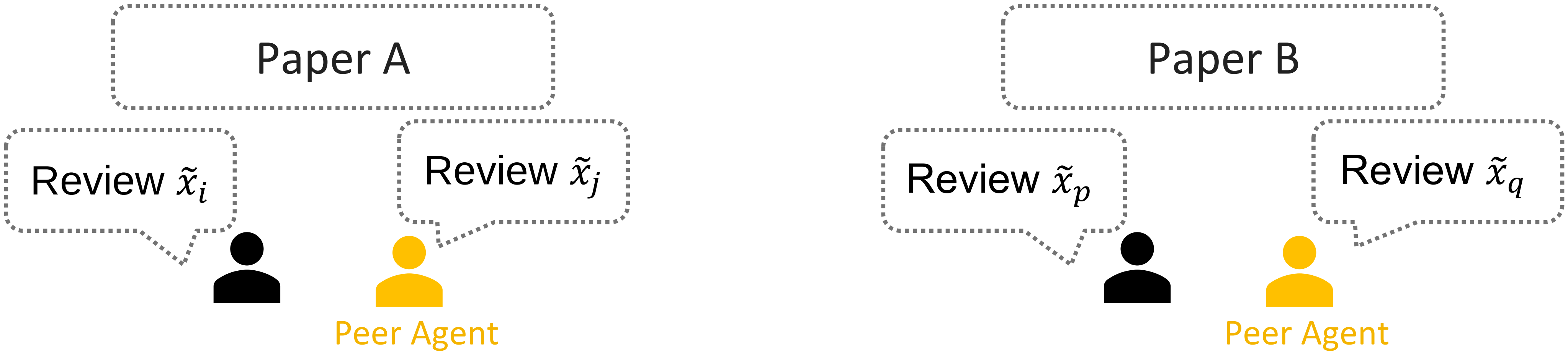
$$Score_i = \log \Pr[\tilde{x}_j \mid \tilde{x}_i] - \boldsymbol{\log \Pr[\tilde{x}_j]}$$

$$Score_p = \log \Pr[\tilde{x}_q \mid \tilde{x}_p] - \boldsymbol{\log \Pr[\tilde{x}_q]}$$

GPPM as an Evaluation Metric

Ideal Evaluation Metric

$$Score_i > Score_p \Leftrightarrow \text{review } \tilde{x}_i \text{ better than } \tilde{x}_p$$



$$Score_i = \log \Pr[\tilde{x}_j \mid \tilde{x}_i] - \mathbf{\log \Pr[\tilde{x}_j]} \qquad Score_p = \log \Pr[\tilde{x}_q \mid \tilde{x}_p] - \mathbf{\log \Pr[\tilde{x}_q]}$$

$$= \mathbf{PMI}(\tilde{\boldsymbol{x}}_i; \tilde{\boldsymbol{x}}_j) \text{ Pointwise Mutual Information}$$

From Information Elicitation to Evaluation

- GPPM => Generative Estimator for Mutual Information (GEM)

$$\mathrm{PMI}\left(\tilde{x}_i; \tilde{x}_j\right) = \log Pr[\tilde{x}_j \mid \tilde{x}_i] - \log \Pr[\tilde{x}_j]$$

- GSPPM => GEM-S

$$\mathrm{PMI}\left(\tilde{x}_i; \tilde{x}_j \mid \theta\right) = \log Pr[\tilde{x}_j \mid \tilde{x}_i, \theta] - \log \Pr[\tilde{x}_j \mid \theta]$$

Validating GEM’s Effectiveness

Accurate

- Positive correlation with human annotation
- Sensitively penalize degradation

Validating GEM's Effectiveness

Accurate

- Positive correlation with human annotation
- Sensitively penalize degradation

Manipulation-resistant

- Robust against manipulations

Validating GEM's Effectiveness

Accurate (corresponding to effort elicitation)

- Positive correlation with human annotation
- Sensitively penalize degradation

Manipulation-resistant (corresponding to truthfulness)

- Robust against manipulations

Baselines

- BLEU and ROUGE-L: pre-LLM era metrics
- BERTScore: embedding-based metric
- BARTScore: probability-based metric
- **LMExaminer**: GPT-4o as the examiner. Our prompt adopts criteria based on Review Quality Indicators (RQIs), including four aspects, understanding, coverage, substantiation, constructiveness.

Positive Correlation with Human Annotation

- Human-Annotated Peer Grading Dataset
  - Graduate-level machine learning class
  - 30 project proposals, ~180 peer reviews
  - Each peer review has
    - ``Strengths of the project“,
    - ``Weaknesses of the project'', and
    - ``Ideas for improvement or specific directions’’
    - TA grade: A, B, or C

Positive Correlation with Human Annotation

- Human-Annotated Peer Grading Dataset
  - Graduate-level machine learning class
  - 30 project proposals, ~180 peer reviews
  - Each peer review has
    - ``Strengths of the project‘’,
    - ``Weaknesses of the project'', and
    - ``Ideas for improvement or specific directions’’
    - TA grade: A, B, or C
- Test correlation between mechanism scores and TA grades

Positive Correlation with Human Annotation

Evaluation Metric	Spearman's ρ	p-value
BLEU	0.023	0.772
ROUGE-L	-0.244	0.002
BERTScore	-0.061	0.439
BARTScore-F1	-0.237	0.002
BARTScore-precision	-0.511	2.3e-12

Evaluation Metric	Spearman's ρ	p-value
BARTScore-recall	**0.164**	0.036
LMExaminer	**0.537**	1.1e-13
GEM-raw	**0.300**	9.2e-05
GEM	**0.431**	7.5e-09
GEM-S	**0.479**	7.4e-11

- Spearman's correlation coefficient between evaluation metrics and instructor-annotated grades.
- Significant positive correlations ($p<0.05$) are bolded.

Sensitivity to Degradation / Robustness against Manipulation

ICLR 2023 Peer Review dataset

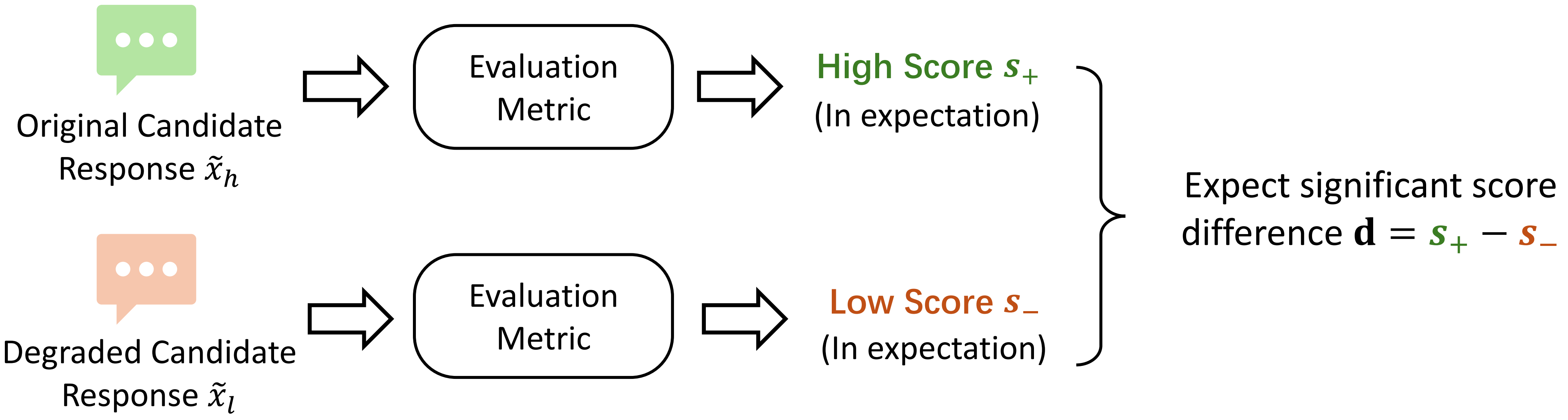
- randomly select 300 papers
- for each paper, randomly select 3 original human reviews
  - one as a human candidate
  - two as peer references

Sensitivity to Degradation

- Sentence Deletion: delete every other sentence of the response.
- Deletion & Completion: after deletion, use GPT-4o to complete the deleted sentences.
- Abstract-only Review: use Claude-3-sonnet to create a fictitious review with only the abstract of the paper

Sensitivity to Degradation

- Sentence Deletion: delete every other sentence of the response.
- Deletion & Completion: after deletion, use GPT-4o to complete the deleted sentences.
- Abstract-only Review: use Claude-3-sonnet to create a fictitious review with only the abstract of the paper



Sensitivity to Degradation

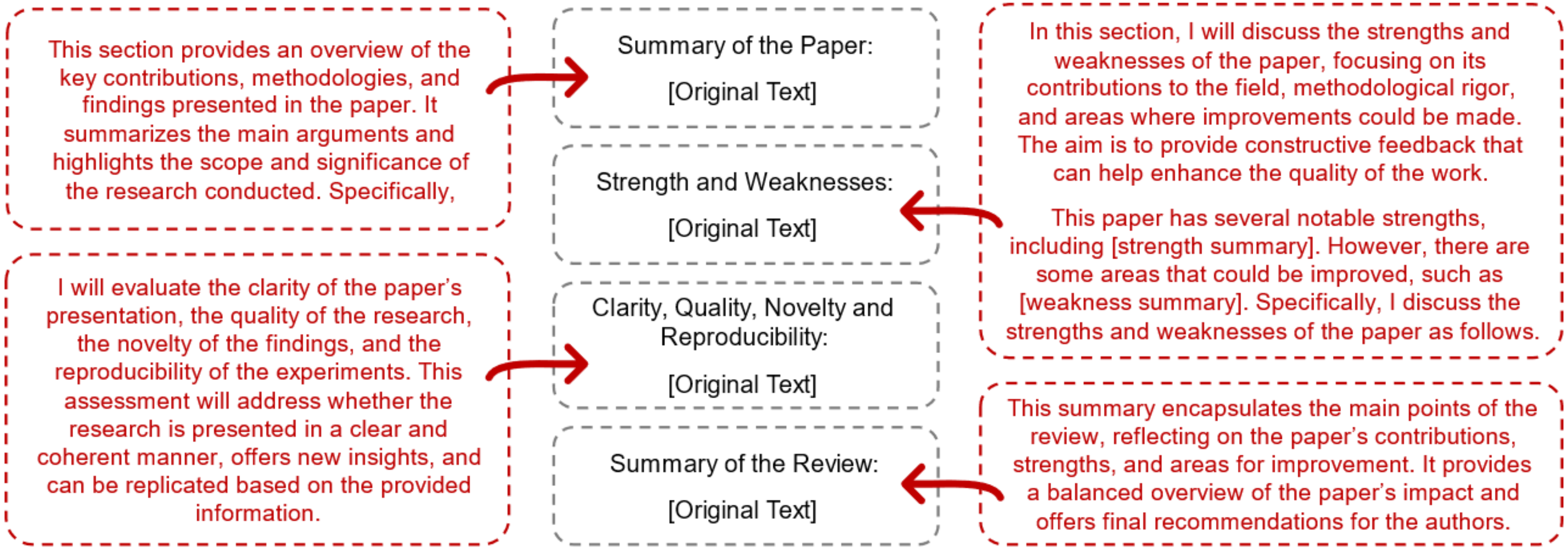
- Standardized Mean Differences (SMD) of scores after manipulations with 95% CI.
- Significant score decreases ($p<0.05$) after degradations are highlighted in bold green, implying the metric can **effectively penalize the degradation**.

Evaluation Metric	Sentence Deletion	Deletion & Completion	Abstract-only Review
BLEU	**-0.282** (-0.346,-0.218)	0.016 (-0.013,0.046)	**-0.692** (-0.778,-0.607)
ROUGE-L	**-0.073** (-0.122,-0.025)	0.091 (0.062,0.121)	0.022 (-0.054,0.098)
BERTScore	**-0.100** (-0.131,-0.069)	**-0.188** (-0.222,-0.155)	0.840 (0.769,0.910)
BARTScore-F1	0.401 (0.380,0.422)	0.201 (0.184,0.218)	0.394 (0.344,0.445)
LMExaminer	**-1.290** (-1.343,-1.238)	**-0.417** (-0.472,-0.363)	0.715 (0.630,0.799)
GEM-raw	**-0.123** (-0.126,-0.120)	**-0.126** (-0.132,-0.121)	0.020 (0.017,0.023)
GEM	**-0.401** (-0.448,-0.354)	**-0.308** (-0.358,-0.258)	**-0.191** (-0.261,-0.122)
GEM-S	**-0.409** (-0.455,-0.362)	**-0.206** (-0.254,-0.158)	**-0.566** (-0.639,-0.492)

Robustness against Manipulation

After manipulation, if the score significantly increases, the evaluation metric fails to pass the robustness check.

- GPT-4o/Llama-3.1 Rephrase.
- Meaningless Elongation.



Robustness against Manipulation

- Standardized Mean Differences (SMD) of scores after manipulations with 95% CI.
- Significant score increase ($p<0.05$) are highlighted in bold red, implying the metric are **not robust** against the manipulation.

Evaluation Metric	GPT-4o Rephrase	Llama3.1 Rephrase	Meaningless Elongation
BLEU	-0.975 (-1.020,-0.930)	-0.920 (-0.971,-0.870)	-0.165 (-0.230,-0.101)
ROUGE-L	**0.028** (0.009,0.047)	**0.120** (0.080,0.160)	-0.196 (-0.241,-0.151)
BERTScore	**0.134** (0.113,0.155)	**0.063** (0.032,0.093)	**0.064** (0.042,0.086)
BARTScore-F1	**0.130** (0.120,0.140)	**0.332** (0.299,0.365)	-0.304 (-0.308,-0.299)
LMexaminer	**0.187** (0.153,0.221)	**0.104** (0.060,0.147)	**0.105** (0.069,0.140)
GEM-raw	-0.123 (-0.126,-0.120)	-0.126 (-0.132,-0.121)	**0.020** (0.017,0.023)
GEM	-0.058 (-0.090,-0.026)	-0.107 (-0.143,-0.070)	-0.063 (-0.097,-0.030)
GEM-S	-0.046 (-0.079,-0.013)	-0.114 (-0.149,-0.078)	-0.070 (-0.104,-0.036)

Generating Review Evaluation Benchmark (GRE-bench)

Evaluation Metric + Dataset = Benchmark

- GEM/GEM-S + ICLR Dataset = GRE-bench

Generating Review Evaluation Benchmark (GRE-bench)

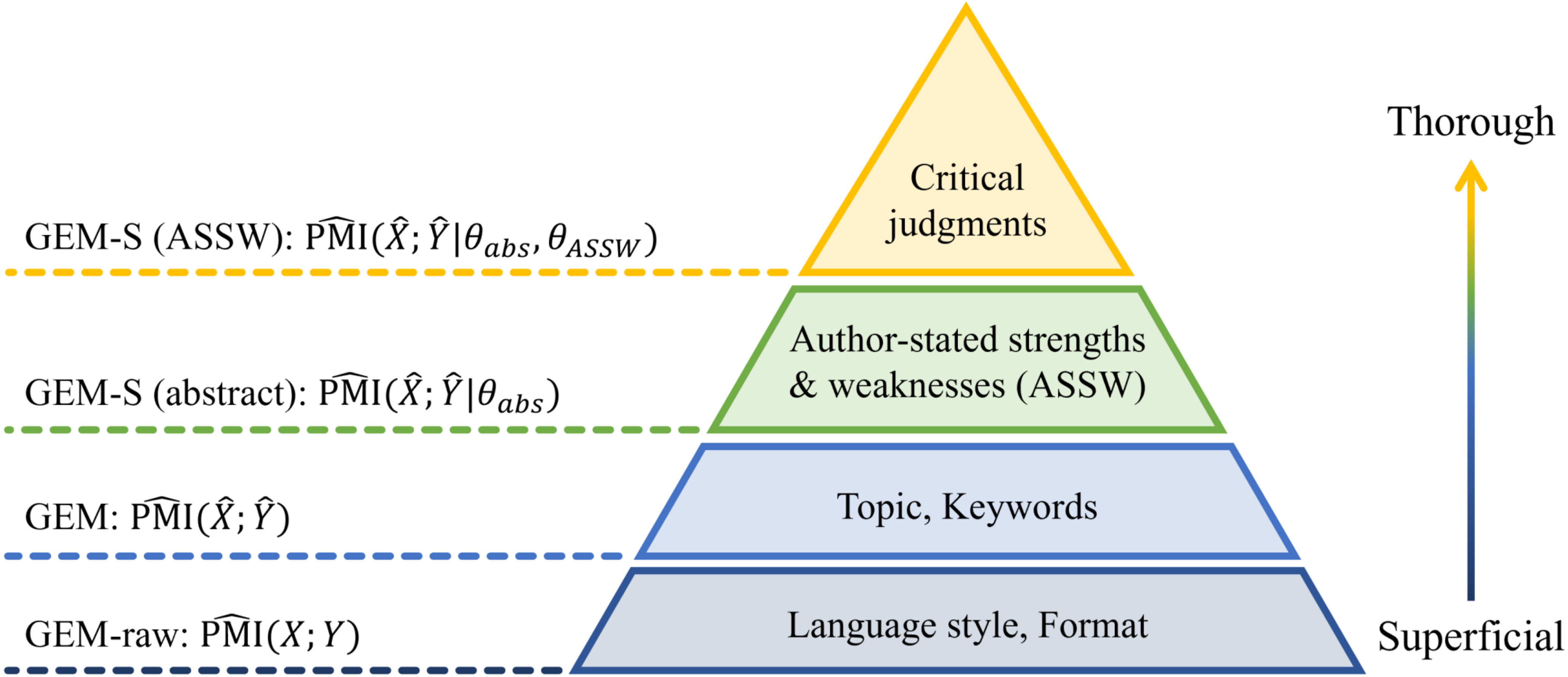
Evaluation Metric + Dataset = Benchmark

- GEM/GEM-S + ICLR Dataset = GRE-bench

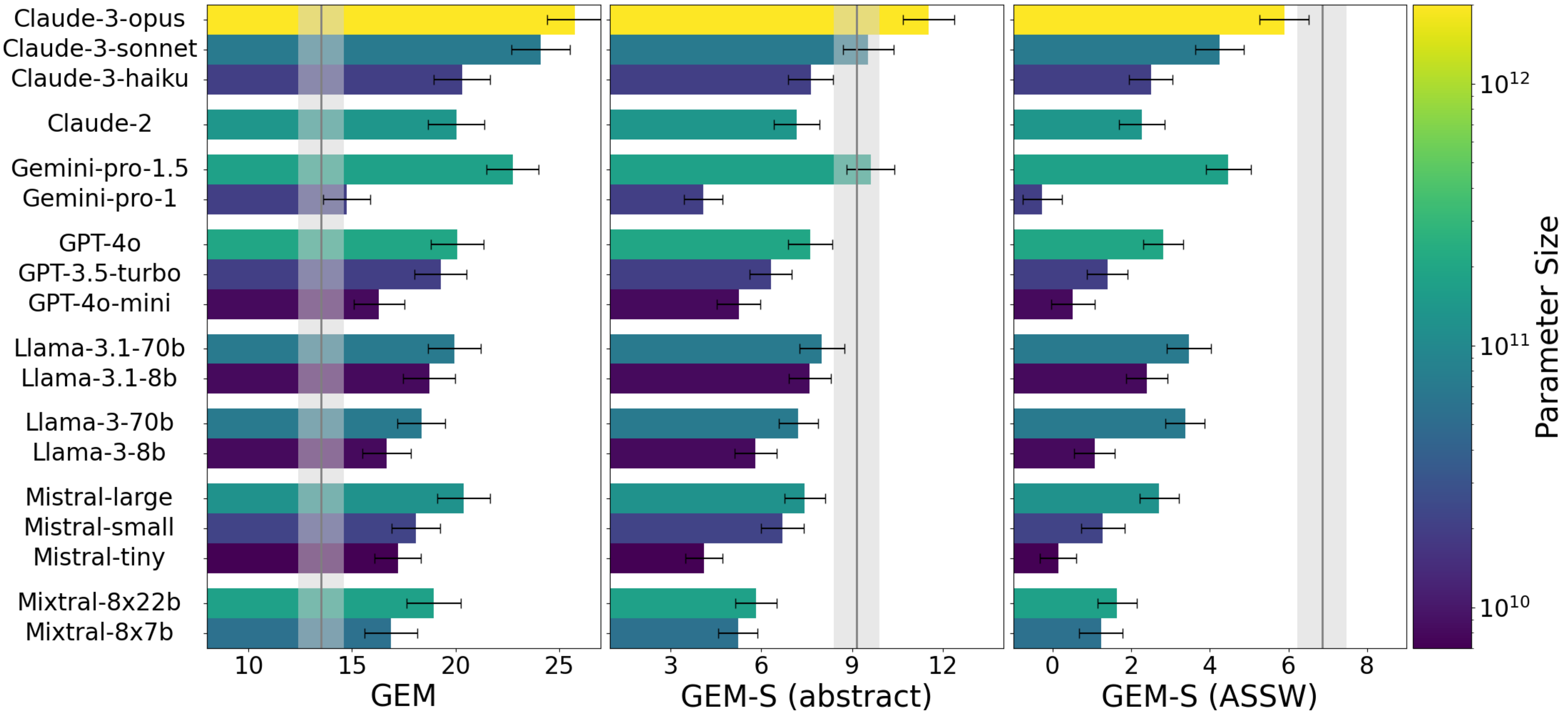
Evaluate LLMs' ability to generate high-quality peer reviews

- Inherit GEM's accuracy and robustness properties.
- Circumvent data contamination by using the continuous influx of new open-access research papers and peer reviews each year.

Hierarchical Information Structure



Results on ICLR2023



GRE-bench vs. other benchmarks

Ability to generate informative reviews relies on several key factors

- GRE-bench highly correlates with benchmarks for reasoning (HellaSWAG, ARC-C)
- Less correlates with benchmarks for coding (HumanEval) or math (MATH, GSM8K)

Base Metric	MMLU	ARC-C	HellaSwag	GSM8K	MATH	HumanEval	GPQA
GEM	0.55	0.68	0.74	0.58	0.37	0.36	0.60
GEM-S (abstract)	0.66	0.70	0.82	0.73	0.43	0.43	0.67
GEM-S (ASSW)	0.68	0.78	0.84	0.73	0.43	0.48	0.71

Conclusion: Benchmarking LLMs’ Judgments with No Gold Standard

- Propose GEM/GEM-S for natural language generation (NLG) evaluation
  - GEM’s manipulation resistance aligned to GPPM’s incentive compatibility
  - Make necessary changes to be more suitable for the NLG evaluation
  - Validate GEM‘s accuracy and manipulation resistance empirically

- Propose the GRE-bench
  - Inherit GEM’s accuracy and manipulation resistance properties
  - Mitigate data contamination issues

Information Elicitation Mechanism

Apply to human agents

- Design more suitable mechanisms for preference elicitation tasks (e.g. RLHF) [Chen, Feng, and Yu, 2024]

Apply to LLMs

Use information elicitation mechanisms

- To evaluate/benchmark LLMs [Xu, Lu, Schoenebeck, and Kong, 2024]
- To calibrate LLMs in finetuning [Band, Li, Ma, and Hashimoto, 2024]

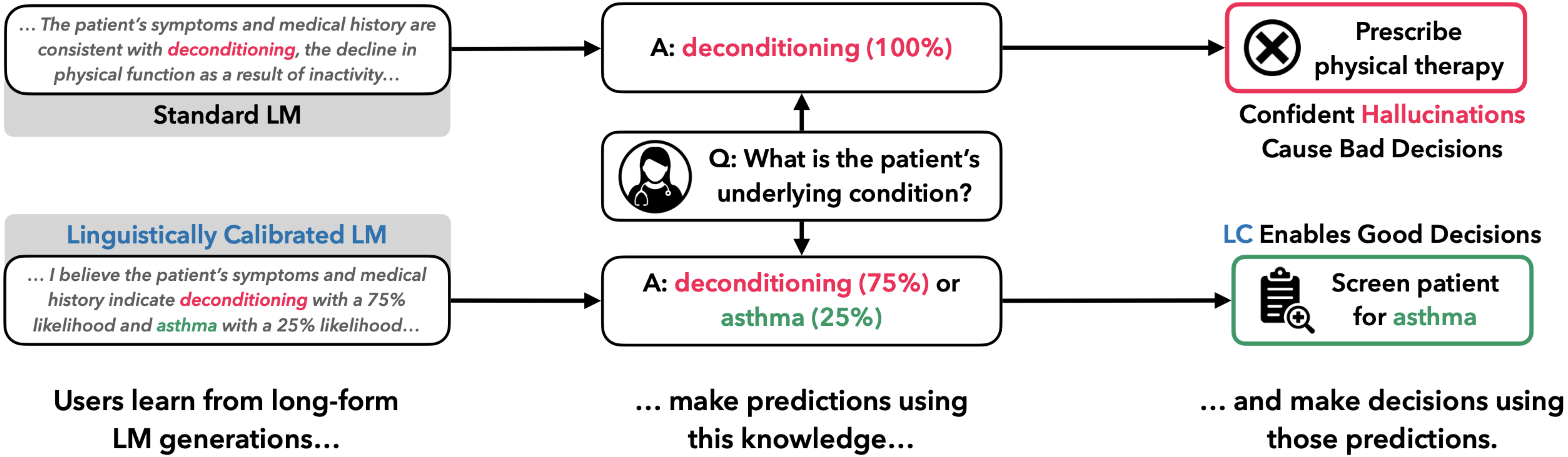
Recall: Proper Scoring Rule

- Proper Scoring Rule := $\mathbb{E}_{\omega \sim q}[S(q, \omega)] \geq \mathbb{E}_{\omega \sim q}[S(p, \omega)]$

Proper Scoring Rule	Loss Function
Log score	Cross-Entropy
Brier score	Mean squared error
Truthfulness	**Calibration**: when forecasting x%, roughly x% should turn out “yes”

Linguistic Calibration [Band, Li, Ma, and Hashimoto, 2024]

- Use a proper scoring rule in finetuning to calibrate confidence statements in natural language, enabling better downstream decisions.



Information Elicitation Mechanism

Apply to human agents

- Design more suitable mechanisms for preference elicitation tasks (e.g. RLHF) [Chen, Feng, and Yu, 2024]

Apply to LLMs

Use information elicitation mechanisms

- To evaluate/benchmark LLMs [Xu, Lu, Schoenebeck, and Kong, 2024]
- To calibrate LLMs in finetuning [Band, Li, Ma, and Hashimoto, 2024]

Human Preference is Needed to Align LLMs

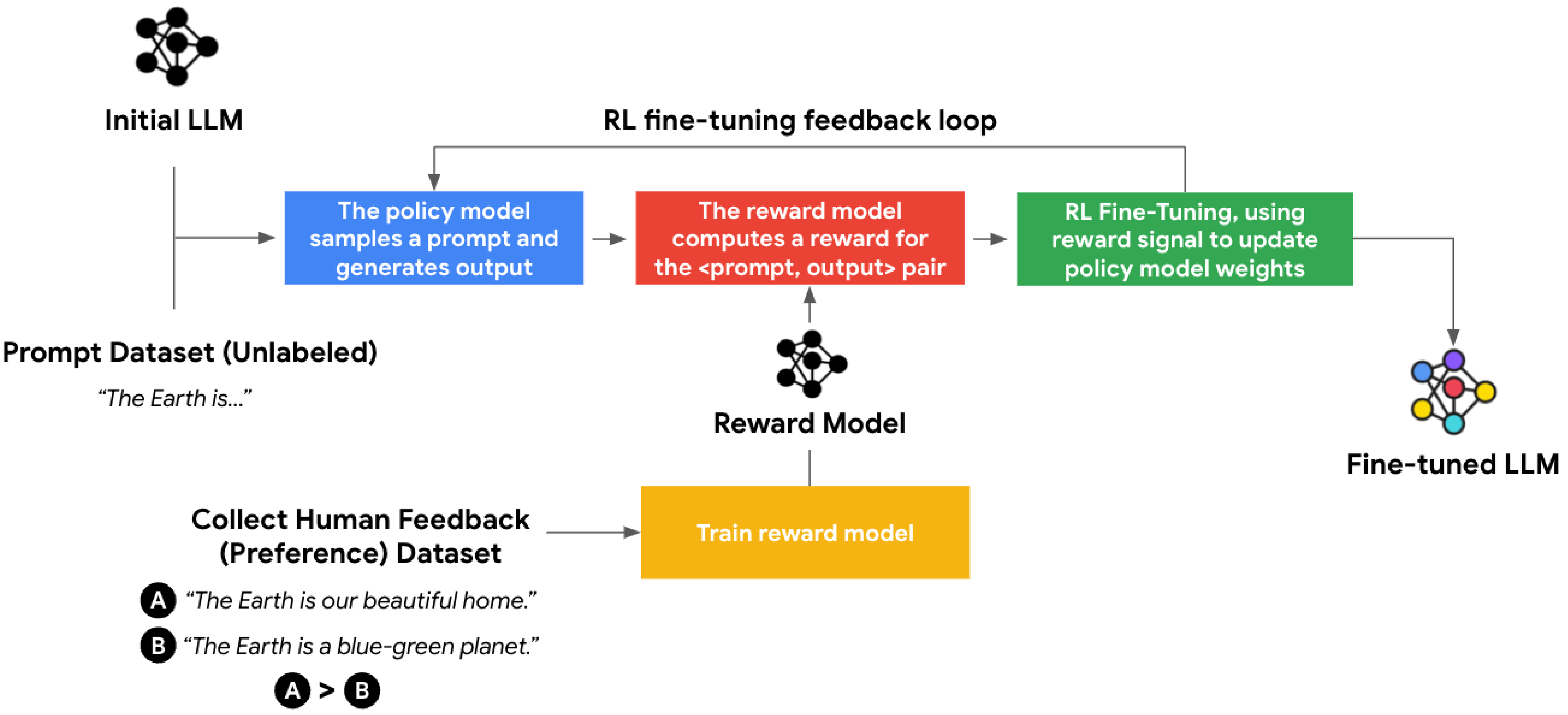


Figure: https://cloud.google.com/blog/products/ai-machine-learning/rlhf-on-google-cloud

Human Preference is Needed to Align LLMs

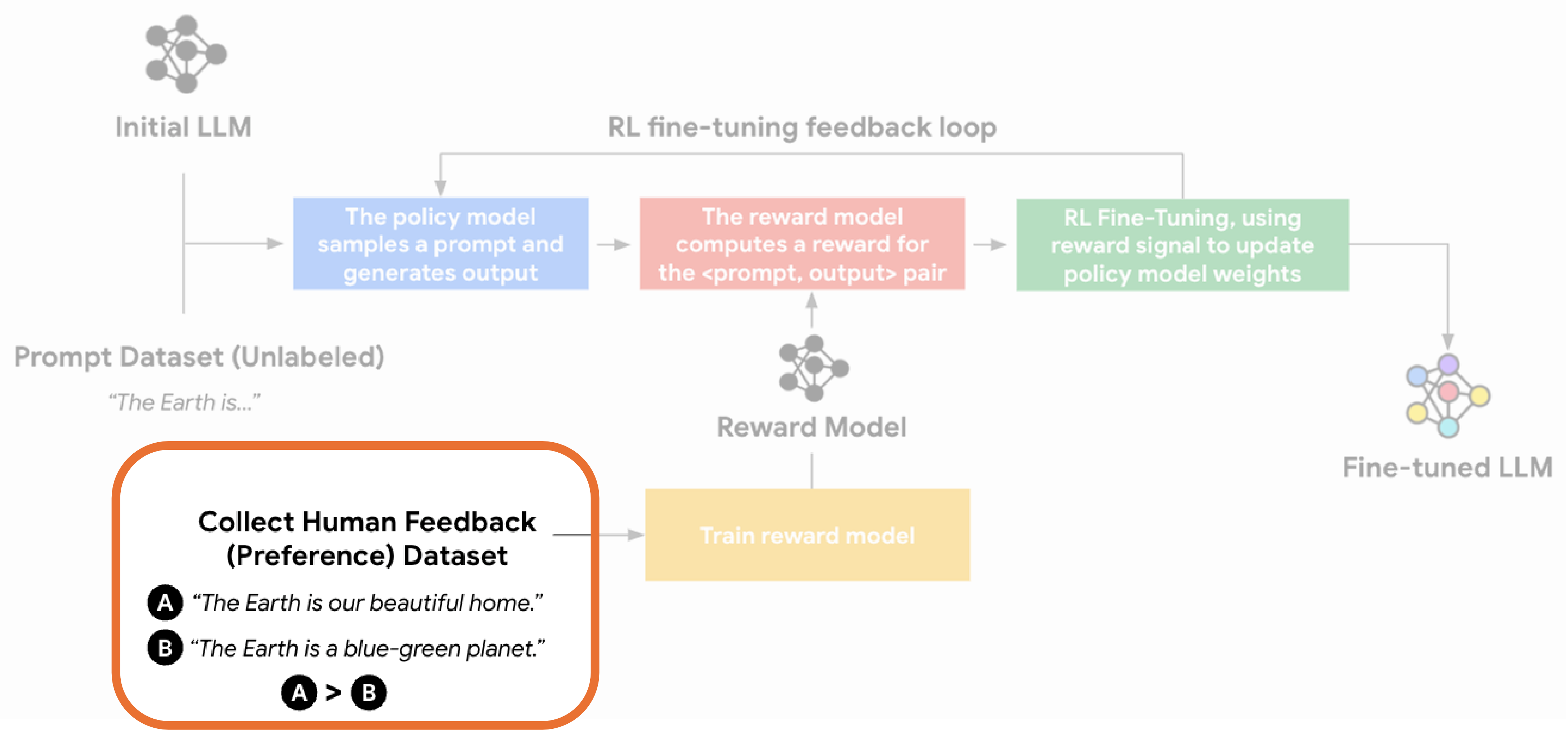


Figure: https://cloud.google.com/blog/products/ai-machine-learning/rlhf-on-google-cloud

Utilize the structure behind Preference

Carrot and Stick: Eliciting Comparison Data and Beyond [Chen, Feng, and Yu, 2024]

- Preference elicitation tasks are not independent

Utilize the structure behind Preference

Carrot and Stick: Eliciting Comparison Data and Beyond [Chen, Feng, and Yu, 2024]

- Preference elicitation tasks are not independent
- Bayesian Strong Stochastic Transitivity (Bayesian SST) model

  [informal]

  - for any three items a, a′, a′′
  - if a is more favorable than a′ and a′ is more favorable than a′′
  - then a is even more favorable than a′′

Utilize the structure behind Preference

Carrot and Stick: Eliciting Comparison Data and Beyond [Chen, Feng, and Yu, 2024]

- Preference elicitation tasks are not independent
- Bayesian Strong Stochastic Transitivity (Bayesian SST) model
    - for any three items a, a′, a′′
    - if a is more favorable than a′ and a′ is more favorable than a′′
    - then a is even more favorable than a′′
- Bonus-Penalty Payment mechanism
    - Achieve symmetrically strongly truthful
    - Require no knowledge of prior (detail-free) and only single task for each agent

CONTENT

Toolkit: Goal

- We hope this lightweight toolkit enables theorists to easily conduct LLM + Information Elicitation research.
  - Making leveraging & deploying LLM undemanding.

Toolkit: Necessities

- You should:
  - Be familiar with ML coding
  - Buy or rent any server
  - Buy or rent any GPU
  - Have your own dataset

Toolkit: Necessities

- ~~You should~~ There is no need for you to:
  - Be familiar with ML coding
  - Buy or rent any server
  - Buy or rent any GPU
  - Have your own dataset

Toolkit: Necessities

- ~~You should~~ There is no need for you to:
  - ~~Be familiar with ML coding~~ Know how to write Python code
  - Buy or rent any server
  - Buy or rent any GPU
  - Have your own dataset

Toolkit: Necessities

- ~~You should~~ There is no need for you to:
  - ~~Be familiar with ML coding~~ Know how to write Python code
  - ~~Buy or rent any server~~ Buy a Google Drive plan ($2/month)
  - Buy or rent any GPU
  - Have your own dataset

Toolkit: Necessities

- ~~You should~~ There is no need for you to:
  - ~~Be familiar with ML coding~~ Know how to write Python code
  - ~~Buy or rent any server~~ Buy a Google Drive plan ($2/month)
  - ~~Buy or rent any GPU~~ Buy a Google Colab pro ($10/month)
  - Have your own dataset

Toolkit: Necessities

- ~~You should~~ There is no need for you to:

~~Be familiar with ML coding~~	Know how to write Python code
~~Buy or rent any server~~	Buy a Google Drive plan ($2/month)
~~Buy or rent any GPU~~	Buy a Google Colab pro ($10/month)
~~Have your own dataset~~	Try your thoughts on our plug-and-play dataset before collecting your own

Toolkit: Detailed Necessities

- A Google account
  - Use your Google Drive as the disk and Google Colab as the server.
  - Colab offers NVIDIA A100 GPUs, capable of running 70B LLMs in 4-bit quantization.
- A Huggingface account
  - Download the models that you want to run.
- Any LLM API
  - Call LLM APIs when you only need LLM output.
    - Cheaper & enables muti-threading
  - Want access to a variety of LLMs?
    - Try using an LLM unified interface like OpenRouter.

Toolkit: Dataset (Peer Review)

- The data from ICLR is fully open to anyone.
- We have prepared a processed ICLR peer review dataset from 2019 to 2024
- You can easily access
  - the text version of the paper
  - the paper’s judgments on itself
  - the summary points induced from review comments
  - many other contents that worth exploring

Toolkit: Dataset (Yelp)

- Yelp dataset contains the review data to restaurants, hospitals, and other businesses.

- To access the Yelp Dataset, you should first get permission.
  - See https://www.yelp.com/dataset

- Instead offering the processed Yelp dataset, we provide code that can convert the raw Yelp dataset to processed ones.

Toolkit: LLM Logits / Logprobs / Embedding

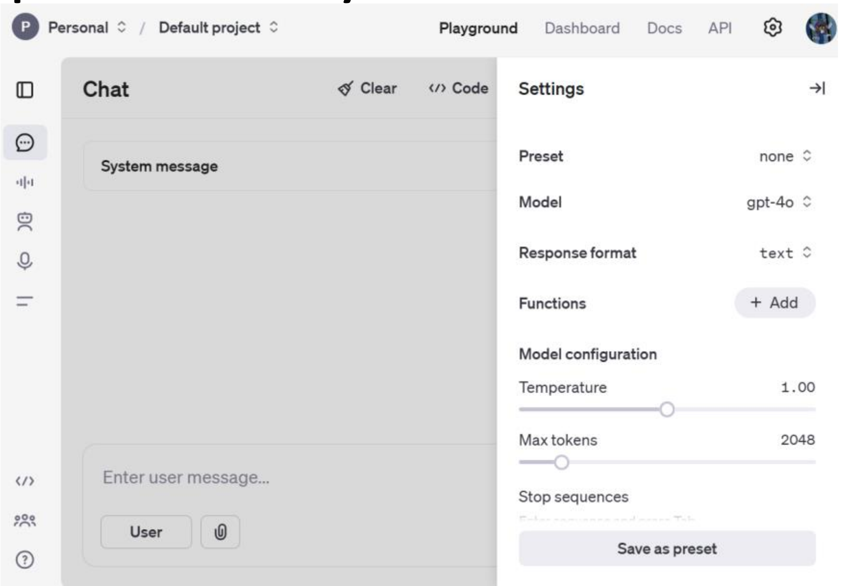
- We provide code to package the LLM's logits, logprobs, and text embedding information into functions that can be directly called.
  - Just choose the open-source LLM you want, you can efficiently access this information.

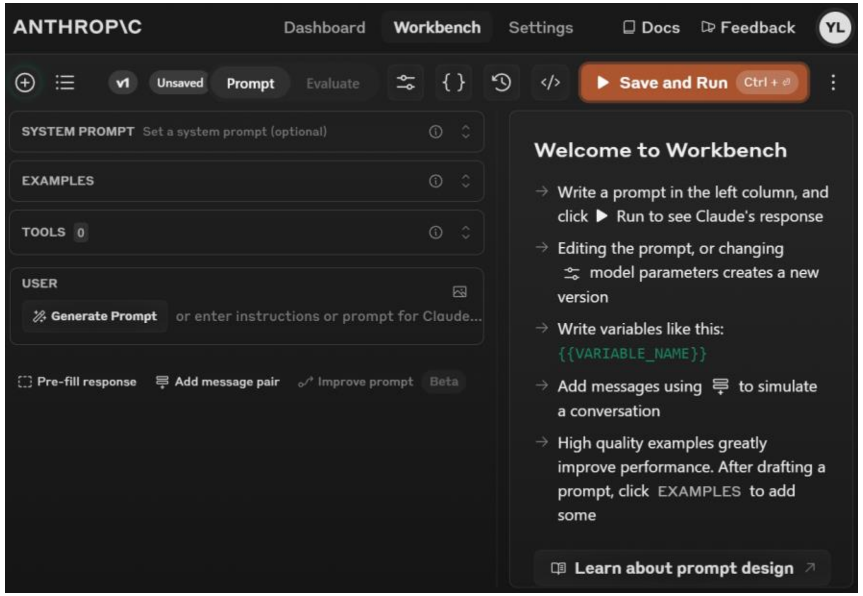
Toolkit: LLM API Call

- We offer an enhanced LLM API call interface, similar to LangChain but simpler, which includes necessary error handling and result caching.

Other than Toolkit: Prompt Engineering

- We recommend that theorists use OpenAI's Playground or Anthropic's API console for initial prompt engineering.
  - Instead of Chat, these platforms are standard LLM API environment
  - They also integrate the MetaPrompt, an automated prompt generation powered by LLM.





Demo

- If we have time…

Eliciting Textual Information with LLM

Current Progress:

- Eliciting Informative Text Evaluations with Large Language Models.

  [Lu, Xu, Zhang, Kong, and Schoenebeck, 2024]

  Generative Peer Prediction

- ElicitationGPT: Text Elicitation Mechanisms via Language Models.

  [Wu and Hartline, 2024]

  High-dimensional Scoring Rules

Information Elicitation enhancing LLMs

Information Elicitation Mechanism

Apply to human agents

- Design more suitable mechanisms for preference elicitation tasks (e.g. RLHF) [Chen, Feng, and Yu, 2024]

Apply to LLMs

Use information elicitation mechanisms

- To evaluate/benchmark LLMs [Xu, Lu, Schoenebeck, and Kong, 2024]
- To calibrate LLMs in finetuning [Band, Li, Ma, and Hashimoto, 2024]

Future Work

- Evaluation (ex-post) vs. Elicitation (ex-ante)
  - E.g., detect low-quality peer reviews on the semantic level
- Generalize these methods to more mechanisms
  - E.g., multi-task peer prediction, Bayesian Truth Serum, prediction market
- Aggregate textual information for better decision-making
- Investigate interpretable/semantic embedding for textual responses
  - For better implementation of the information elicitation mechanisms
- Mitigate hallucination in LLM outputs

Acknowledgment

Advisors & Collaborators

- Grant Schoenebeck, University of Michigan
- Yuqing Kong, Peking University
- Yichi Zhang, DIMACS, Rutgers University

Consulting

- Yifan Wu, Northwestern University

Thanks for your listening!

Contact

Yuxuan Lu, yx_lu@pku.edu.cn

Shengwei Xu, shengwei@umich.edu

Code Repo Link at yxlu.me/projects

QR Code of our EC'24 Paper