Preference-grounded Token-level Guidance for Language Model Training

Shentao Yang[1], Shujian Zhang[1], Congying Xia[2], Yihao Feng[2], Caiming Xiong[2], Mingyuan Zhou[1]

[1]The University of Texas at Austin, [2]Salesforce Research

November, 2023

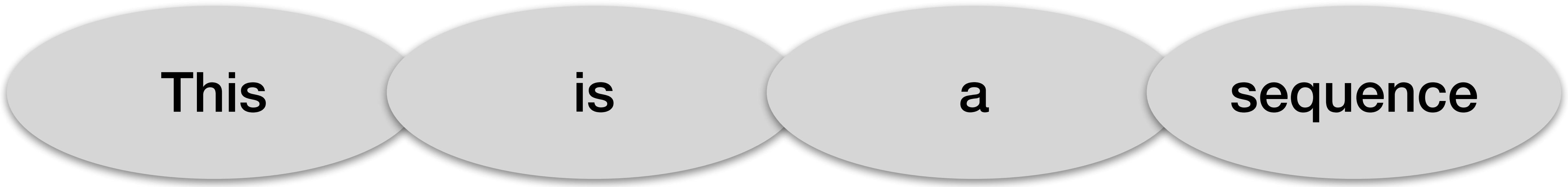
TL;DR

- Research Question: how to effectively ground sequence-level preference into dense token-level guidance for language model training

TL;DR

- Research Question: how to effectively ground sequence-level preference into dense token-level guidance for language model training

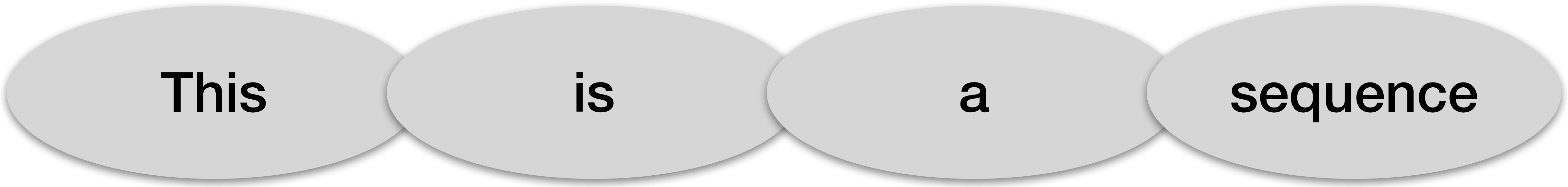
  - Sequence: text-sequence, *e.g.*, a sentence or a paragraph of words.



TL;DR

- Research Question: how to effectively ground sequence-level preference into dense token-level guidance for language model training

  - Sequence: text-sequence, *e.g.*, a sentence or a paragraph of words.



  - Preference: an ordering of multiple text-sequences based on the evaluations of *whole* sequence

    - Evaluations: automatic evaluation metrics or humans, *e.g.* length

Background: How to train a language model?

- By the token-level cross-entropy loss

- Token-level: each token in the sentence has a corresponding term in the overall training loss

This   is   a   sentence

$$\max : \Pr\left(\textbf{This} \mid \textbf{<sos>}\right) \times \Pr\left(\textbf{is} \mid \textbf{This}\right) \times \Pr\left(\textbf{a} \mid \textbf{This is}\right) \times \Pr\left(\textbf{sentence} \mid \textbf{This is a}\right)$$

Background: Preference is NOT token-level

- Preference is provided only at the sequence level

- “*Which of the two sequences is better?*”

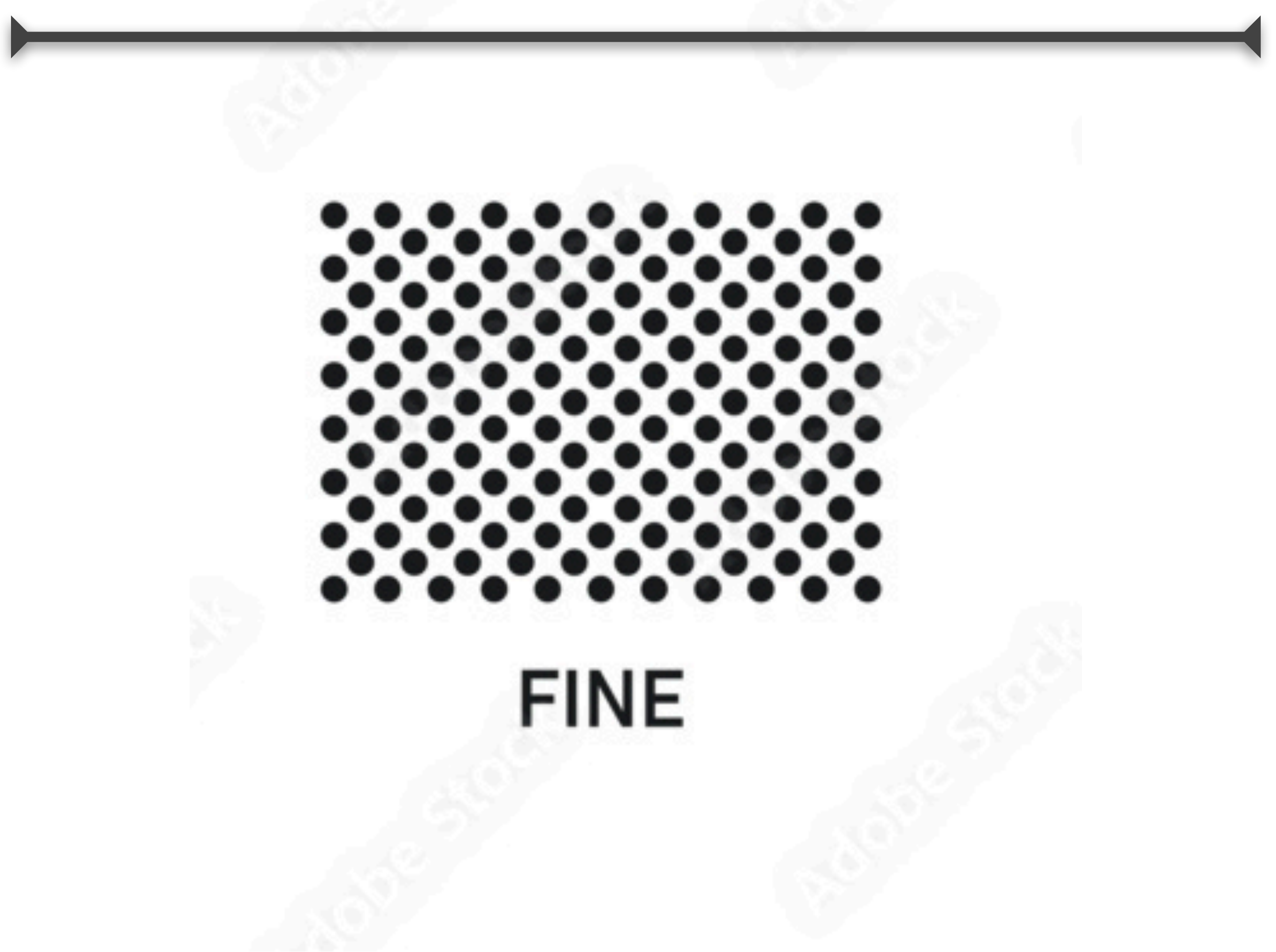
    - Only available after the entire sequence has been generated

    - Evaluates the whole sequence

Issue: Granularity mismatch

- Guiding training: granularity mismatch
  - Mismatch: sequence-level preference *v.s.* token-level training loss

v.s.



- Harm training process — higher gradient variance and lower sample efficiency!

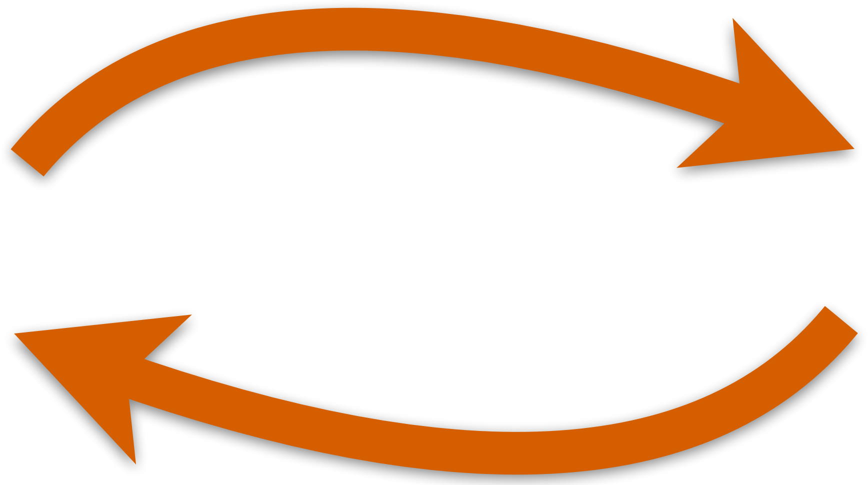
Our method: Overview

- Mismatch: sequence-level preference *v.s.* token-level training loss

Our method: Overview

- Mismatch: sequence-level preference *v.s.* token-level training loss

- Our solution: an alternate training process

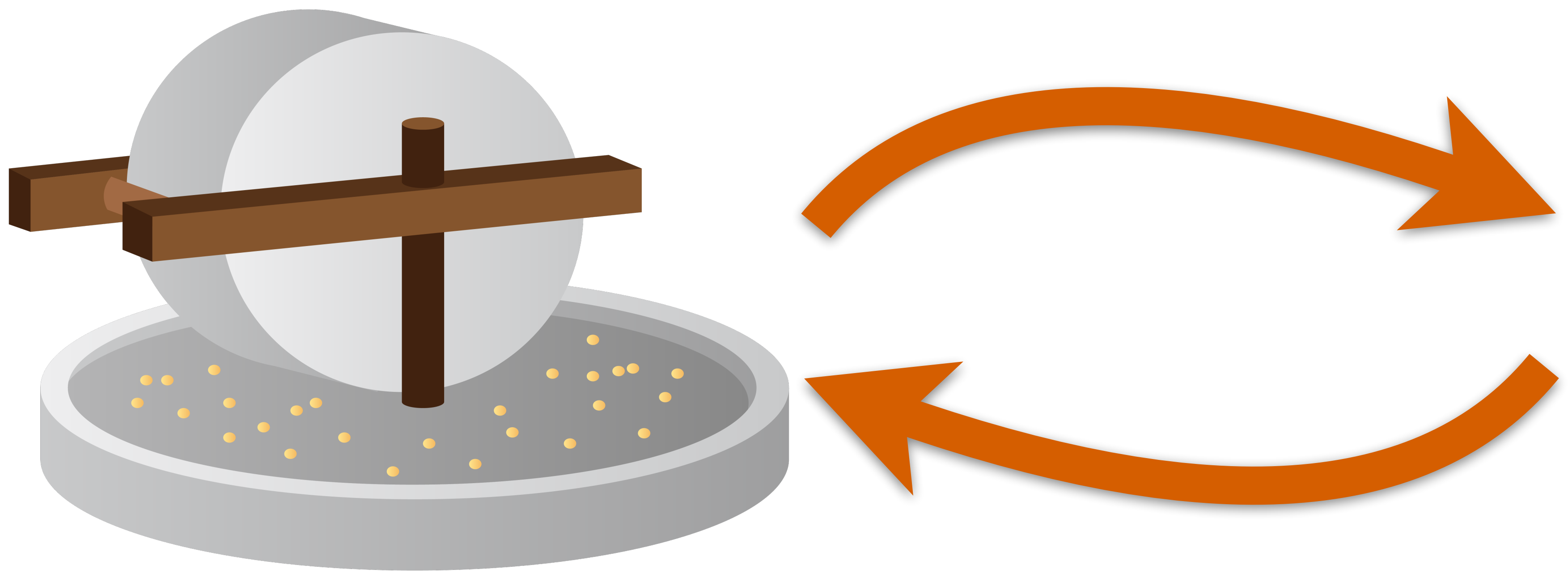


Our method: Overview

- Mismatch: sequence-level preference *v.s.* token-level training loss

- Our solution: an alternate training process

  ① Ground sequence-level preference into token-level training guidance

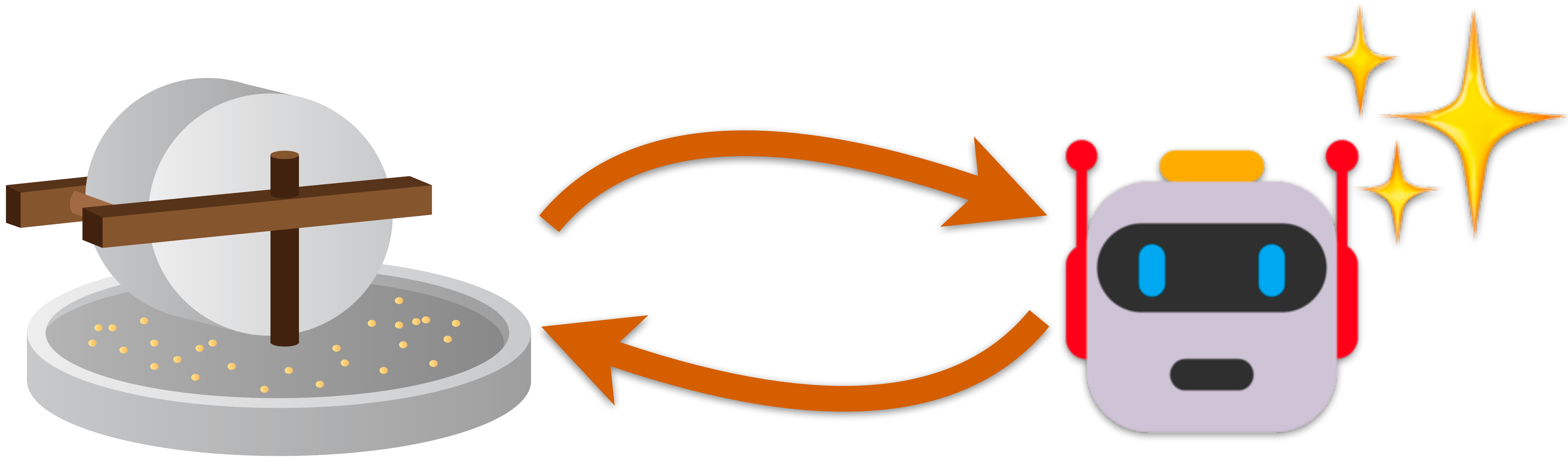


Our method: Overview

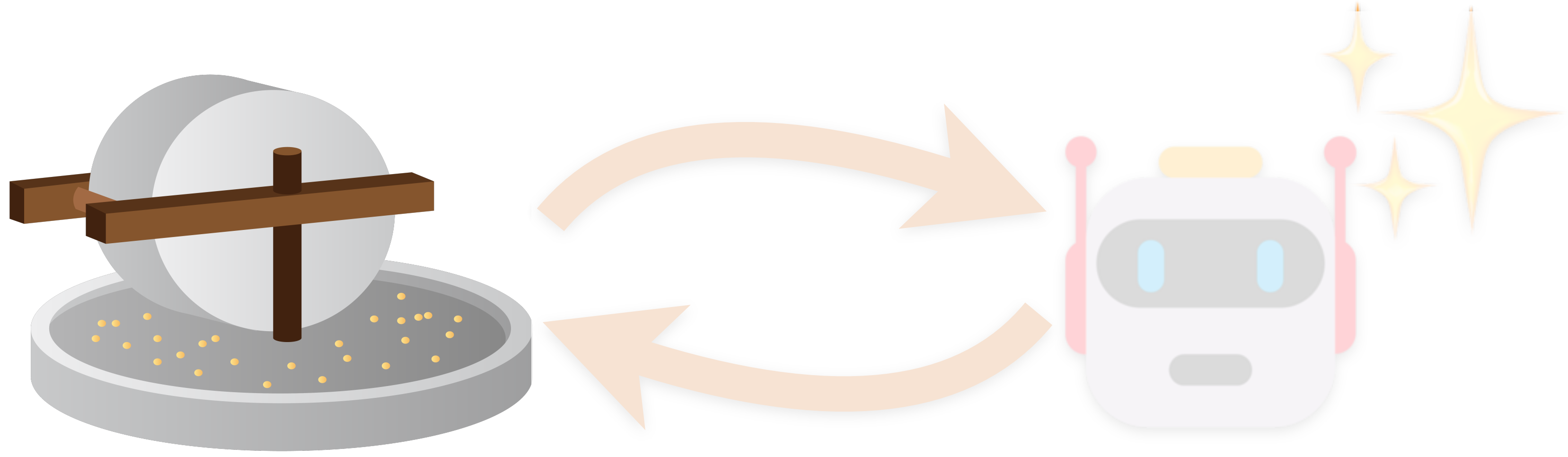
- Mismatch: sequence-level preference *v.s.* token-level training loss
- Our solution: an alternate training process

  ① Ground sequence-level preference into token-level training guidance

  ② Improve the LM π_θ using the learned guidance

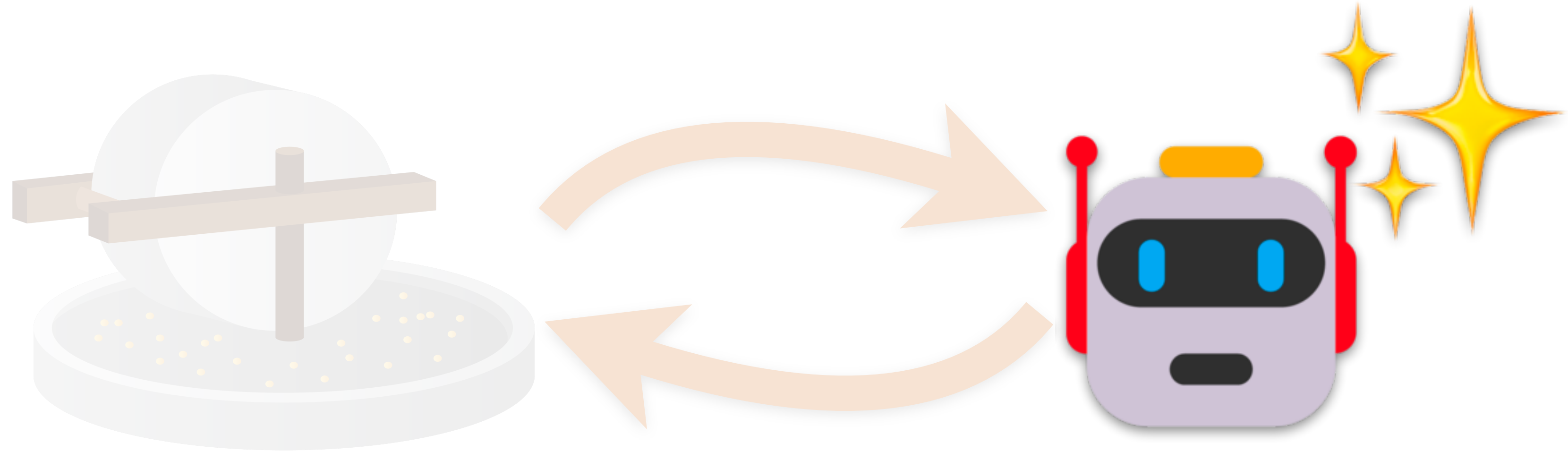


Our method: Ground preference into training guidance



- The LM is fixed
- Goal: learn a parametrized token-level "reward" function
  - Score the word selection at each step of the sequence
  - *"Is it good to select this token here?"*

Our method: Using the reward function



- Provide dense training guidance
  - Dense guidance: how to select each token in the sequence
- Setting: no supervised data, LM needs to discover good text by itself
- Select the next token such that the resulting reward is high
- Implemented by the classical REINFORCE method

Experiment: Task description

- Prompt generation for text classification
  - Goal: generate text prompts to ask a large language model to classify texts
  - Evaluation metric: test accuracy
  - Preference source: the stepwise metric in RLPrompt[1]
  - Dataset: *SST-2* and *Yelp Polarity* (sentiment, binary); *AG News* (topic, four-way)

- Also experiment on text summarization—check paper for results & discussions!

[1] Deng, Mingkai, et al. "Rlprompt: Optimizing discrete text prompts with reinforcement learning." arXiv preprint arXiv:2205.12548 (2022).

Experiment: Main results

Table 1: Test accuracy on the prompt task. Best overall result is bold and best discrete-prompt result is underlined if different. The reported results are mean (standard deviation) over three random seeds.

		SST-2	Yelp P.	AG News
Finetuning	Few-shot Finetuning	80.6 (3.9)	88.7 (4.7)	**84.9** (3.6)
Continuous Prompt	Soft Prompt Tuning	73.8 (10.9)	88.6 (2.1)	82.6 (0.9)
	BB Tuning-50	89.1 (0.9)	93.2 (0.5)	83.5 (0.9)
	AutoPrompt	75.0 (7.6)	79.8 (8.3)	65.7 (1.9)
Discrete Prompt	Manual Prompt	82.8	83.0	76.9
	In-Context Demo	85.9 (0.7)	89.6 (0.4)	74.9 (0.8)
	Instructions	89.0	84.4	54.8
	GrIPS	87.1 (1.5)	88.2 (0.1)	65.4 (9.8)
	RLPrompt	90.5 (1.5)	94.2 (0.7)	79.7 (2.1)
	Ours (AVG)	**92.6** (1.7)	94.7 (0.6)	82.8 (1.5)
	Ours (MIN)	91.9 (1.8)	94.4 (0.8)	82.4 (1.1)
	Ours (MAX)	91.2 (2.5)	**94.8** (0.5)	<u>83.3</u> (1.4)

- Competitive and stable results on all three datasets

Experiment: Main results

Table 1: Test accuracy on the prompt task. Best overall result is bold and best discrete-prompt result is underlined if different. The reported results are mean (standard deviation) over three random seeds.

		SST-2	Yelp P.	AG News
Finetuning	Few-shot Finetuning	80.6 (3.9)	88.7 (4.7)	**84.9** (3.6)
Continuous Prompt	Soft Prompt Tuning	73.8 (10.9)	88.6 (2.1)	82.6 (0.9)
	BB Tuning-50	89.1 (0.9)	93.2 (0.5)	83.5 (0.9)
	AutoPrompt	75.0 (7.6)	79.8 (8.3)	65.7 (1.9)
Discrete Prompt	Manual Prompt	82.8	83.0	76.9
	In-Context Demo	85.9 (0.7)	89.6 (0.4)	74.9 (0.8)
	Instructions	89.0	84.4	54.8
	GrIPS	87.1 (1.5)	88.2 (0.1)	65.4 (9.8)
	RLPrompt	90.5 (1.5)	94.2 (0.7)	79.7 (2.1)
	Ours (AVG)	**92.6** (1.7)	94.7 (0.6)	82.8 (1.5)
	Ours (MIN)	91.9 (1.8)	94.4 (0.8)	82.4 (1.1)
	Ours (MAX)	91.2 (2.5)	**94.8** (0.5)	<u>83.3</u> (1.4)

- RLPrompt: directly optimize sequence-level feedback by RL method

- Improvement → our finer token-level guidance is more effective than coarse sequence-level feedback

Takeaway

- To train a sequential-decision-making model, such as LM, it can be more effective to use finer guidance, compared to coarse feedback

Full Paper

GitHub Repo