Prompt Optimization in the Wild
Challenges and Opportunities

Xiaoqiang Lin
https://xqlin98.github.io/
Nov 28, 2024

Invited Talk at Max Planck Research School for Intelligent Systems (IMPRS-IS) and University of Stuttgart

Agenda

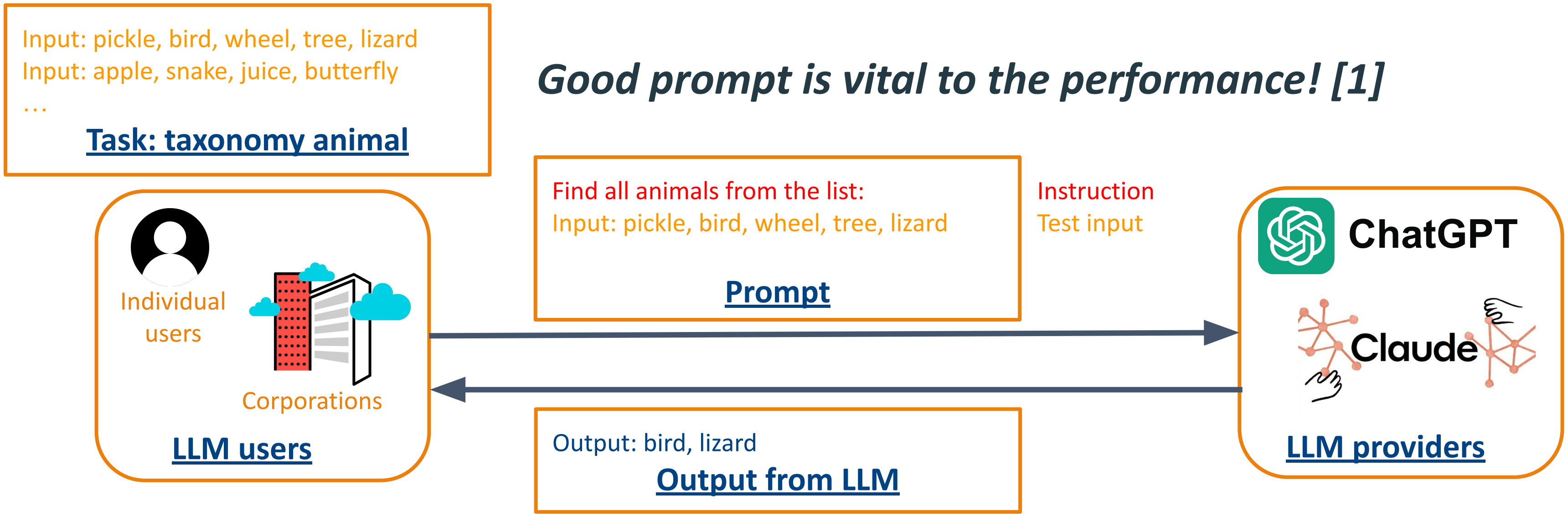
What is prompt optimization

What are the challenges

What's next

What is prompt optimization

Good prompt is vital to the performance! [1]



[1] Mishra, S., Khashabi, D., Baral, C., Choi, Y., & Hajishirzi, H. (2021). Reframing Instructional Prompts to GPTk's Language. In *Proc. ACL Findings*.

What is prompt optimization

- Human designed prompt can be costly and suboptimal
- Prompt optimization: Automatically optimize the prompts (including the instruction and exemplars) to obtain the best performance of LLMs

Find all animals from the list: — Instruction

Input: sweater, octopus, giraffe, orange
Output: octopus, giraffe
Input: apple, lion, ladder
Output: lion — Exemplars

Input: pickle, bird, wheel, tree, lizard — Test input

Prompt

Agenda

What is prompt optimization

What are the challenges

What's next

What are the challenges

- Best performing LLMs are black-box models
  - ChatGPT (e.g., GPT3.5, GPT 4), Claude: only API access is available
  - Gradient-based approaches are not applicable
- Access to black-box LLMs is costly
  - API calls are expensive
  - A query-efficient approach is needed: query as less as possible to find the best prompt
- Sometimes, no scoring method to quantify the quality of prompt
  - A validation dataset is unavailable
  - Scoring method can be unreliable

To tackle the challenges

Use Your INSTINCT: INSTruction optimization for LLMs usIng Neural bandits Coupled with Transformers (ICML 2024)

- Black-box query efficient instruction optimization

Prompt Optimization with EASE? Efficient Ordering-aware Automated Selection of Exemplars (NeurIPS 2024)

- Black-box query efficient exemplar selection

Prompt Optimization with Human Feedback (ICML 2024 Workshop Oral)

- Optimize the prompt when scoring method is unavailable

To tackle the challenges

Use Your INSTINCT: INSTruction optimization for LLMs usIng Neural bandits Coupled with Transformers (ICML 2024)

- Black-box query efficient instruction optimization

Prompt Optimization with EASE? Efficient Ordering-aware Automated Selection of Exemplars (NeurIPS 2024)

- Black-box query efficient exemplar selection

Prompt Optimization with Human Feedback (ICML 2024 Workshop Oral)

- Optimize the prompt when scoring method is unavailable

USE YOUR INSTINCT: INSTRUCTION OPTIMIZATION USING NEURAL BANDITS COUPLED WITH TRANSFORMERS

Xiaoqiang Lin*, Zhaoxuan Wu*, Zhongxiang Dai, Wenyang Hu, Yao Shu, See-Kiong Ng, Patrick Jaillet, Bryan Kian Hsiang Low

In ICML 2024

03 Formulation: Instruction Optimization

- Black-box LLM f
- Instruction ρ
- Input-output pairs: (x, y)
- A validation dataset: $D_V = \{(x_i, y_i)\}_{i=1}^n$
- LLM takes instruction ρ prepended to a test input x, then output y
- Evaluation function: $s(\cdot, \cdot)$
- Objective:

$$\rho^* = \text{argmax}_\rho h(\rho)$$

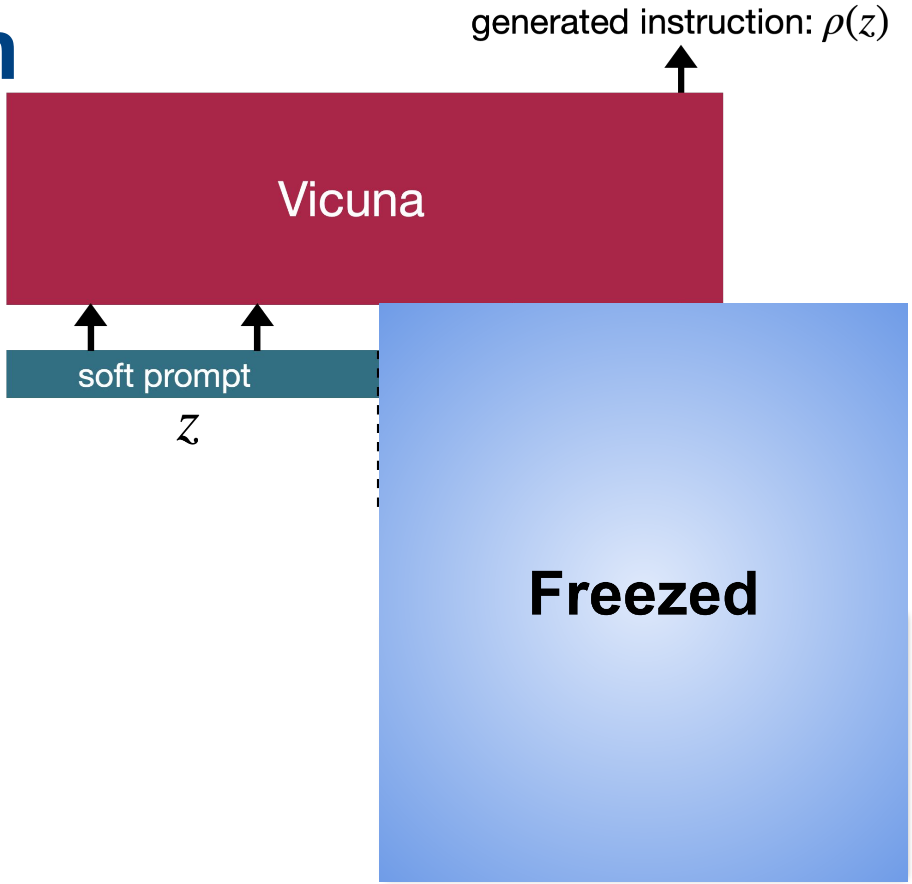
$$h(\rho) := \mathbb{E}_{(x,y)\in D_V} s\big(f(\rho, x), y\big)$$

Preliminary - Bayesian Optimization (BO)

- Sequential black-box optimization: find $\rho^* = \mathrm{argmax}_\rho h(\rho)$
- To choose sequential queries $\rho_1, \ldots, \rho_t$ intelligently:
  - Uses a Gaussian process (GP) as a surrogate to model the objective function
  - Chooses queries by maximizing an acquisition function to balance exploration vs exploitation

INSTINCT Algorithm

- Map a soft prompt z (a vector in continuous space) into instruction $\rho(z)$
  - Search in the continuous space

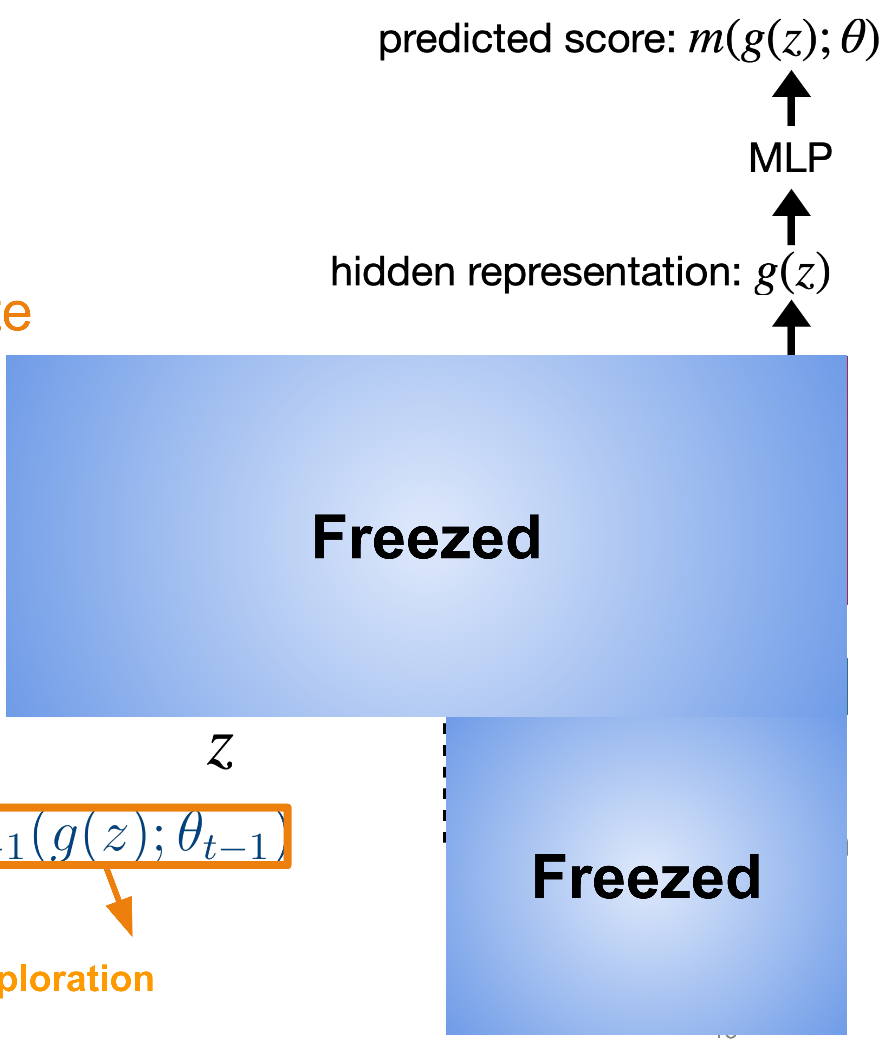


INSTINCT Algorithm

- Uses the whole Vicuna as surrogate model to leverage the expressive power of transformer: $m(g(z);\theta)$
- Acquisition function from **NeuralUCB** algorithm:

$$z_t = \text{argmax}_{z \in Z} \text{NeuralUCB}_t(z)$$

$$\text{NeuralUCB}_t(z) := \underbrace{\boxed{m(g(z);\theta_{t-1})}}_{\textbf{Exploitation}} + \nu_t \underbrace{\boxed{\sigma_{t-1}(g(z);\theta_{t-1})}}_{\textbf{Exploration}}$$



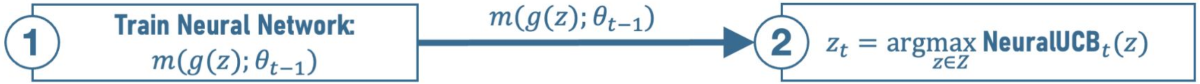
04 INSTINCT Algorithm

Step ①: Training the neural network for score prediction

① **Train Neural Network:**
$m(g(z); \theta_{t-1})$

04 INSTINCT Algorithm

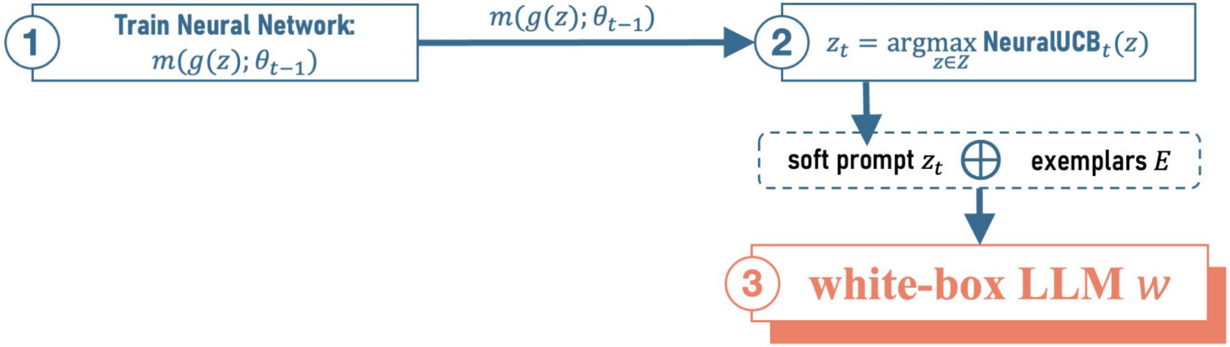
Step ②: Selecting the next soft prompt using the NeuralUCB algorithm



04

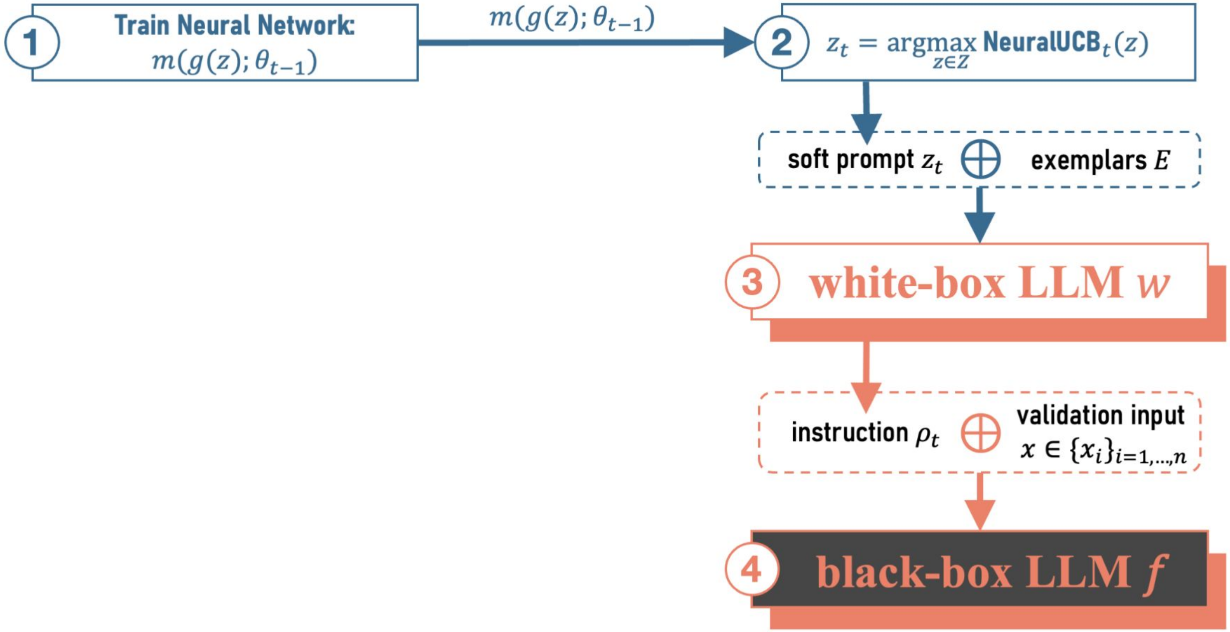
INSTINCT Algorithm

Step ③: Generating the instruction using a white-box LLM



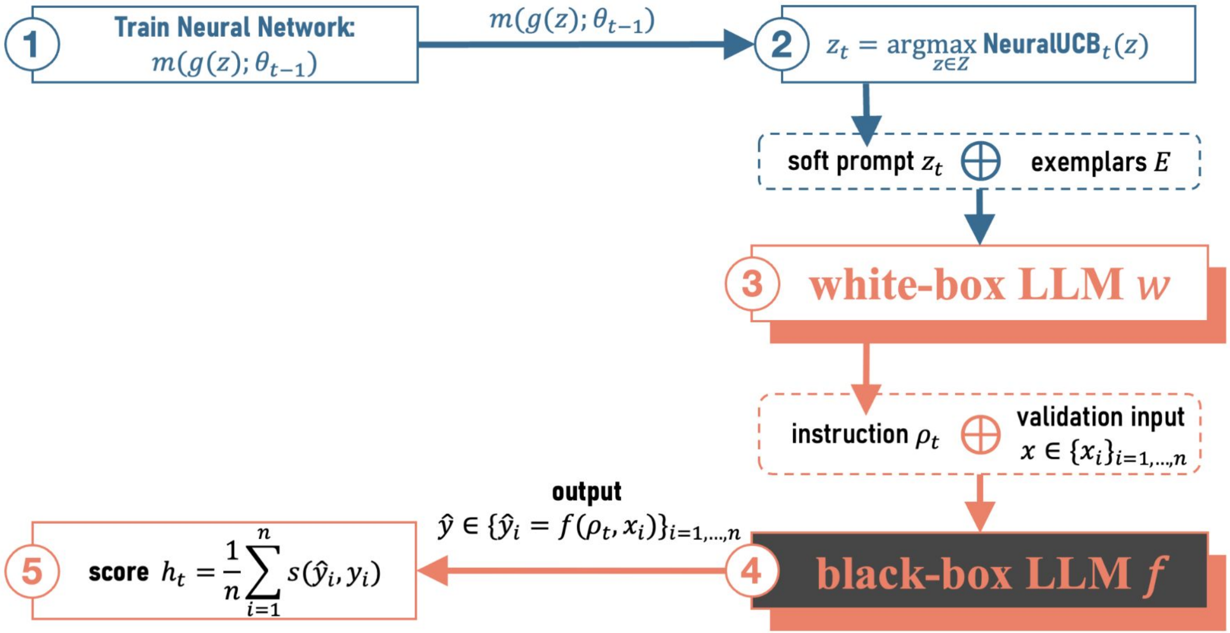
04 INSTINCT Algorithm

Step ④: Predicting the label for a validation dataset using black-box LLM and the generated instruction



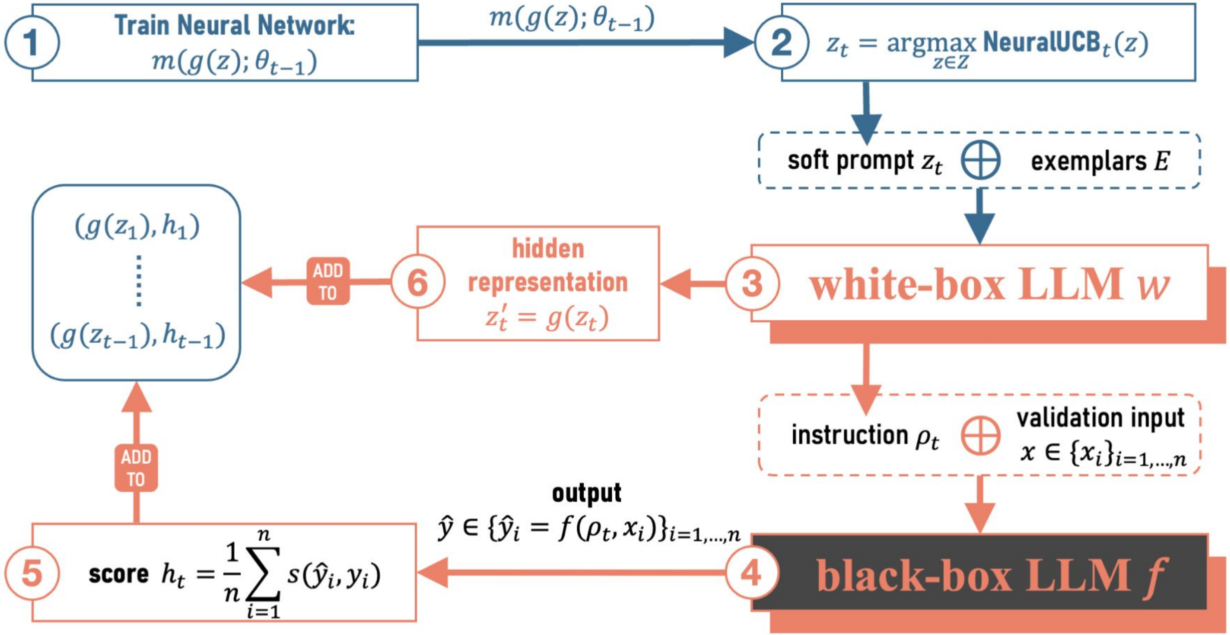
04 INSTINCT Algorithm

Step ⑤: Evaluating the predicted results (i.e., the performance of the instruction)



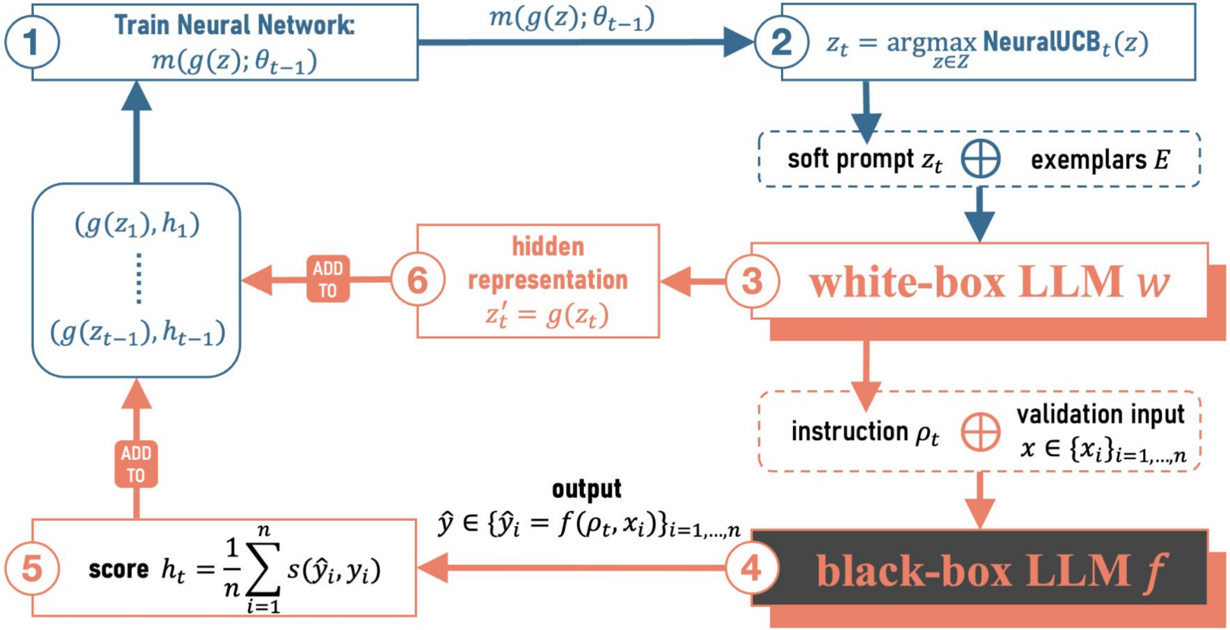
04 INSTINCT Algorithm

Step ⑥: Extracting the hidden representation from the white-box LLM for the instruction



04 INSTINCT Algorithm

Adding the hidden representation and the evaluated score to the dataset which is used to train the neural network. Repeat.



Instruction Induction

Task	APE	InstructZero	INSTINCT (ours)
antonyms	0.6367(0.1416)	0.8267(0.0072)	**0.8467(0.0027)**
auto_categorization	0.2500(0.0094)	**0.2567(0.0119)**	0.2500(0.0330)
auto_debugging	0.2917(0.0340)	**0.3750(0.0000)**	0.2917(0.0340)
cause_and_effect	0.5733(0.0891)	**0.8133(0.0109)**	0.5867(0.0871)
common_concept	0.0691(0.0207)	0.0864(0.0398)	**0.2129(0.0019)**
diff	0.6733(0.2667)	0.6933(0.2224)	**1.0000(0.0000)**
informal_to_formal	**0.5736(0.0026)**	0.5310(0.0024)	0.5534(0.0000)
letters_list	**1.0000(0.0000)**	0.5900(0.1674)	**1.0000(0.0000)**
negation	0.7533(0.0109)	0.7767(0.0136)	**0.8167(0.0027)**
object_counting	**0.3633(0.0191)**	0.3600(0.0929)	0.3400(0.0698)
odd_one_out	0.6333(0.0144)	0.6133(0.0871)	**0.7000(0.0163)**
orthography_starts_with	0.4567(0.1477)	0.5067(0.0871)	**0.6667(0.0272)**
rhymes	0.1567(0.0640)	**1.0000(0.0000)**	**1.0000(0.0000)**
second_word_letter	**0.7467(0.2028)**	0.4333(0.1872)	0.1000(0.0411)
sentence_similarity	0.0000(0.0000)	0.0000(0.0000)	**0.1400(0.0047)**
sum	0.6733(0.2667)	**1.0000(0.0000)**	**1.0000(0.0000)**
synonyms	**0.3600(0.0759)**	0.2767(0.0925)	0.3067(0.0491)
taxonomy_animal	0.3467(0.2341)	0.7167(0.0838)	**0.8567(0.0599)**
word_sorting	0.3300(0.0374)	0.3100(0.1143)	**0.5133(0.0027)**
word_unscrambling	0.4400(0.1389)	0.5500(0.0170)	**0.6333(0.0072)**
# best-performing tasks	5	5	13
# second-best-performing tasks	5	10	5
average rank	2.25	2.0	1.45

Instruction Induction (Summarization Task)

- INSTINCT also performs the best in another commonly used *SAMSum* benchmark dataset

Method	ROUGE-1	ROUGE-2	ROUGE-L
APE	0.32549	0.10308	0.30245
InstructZero	0.32595	0.10528	0.30061
INSTINCT	**0.35580**	**0.13350**	**0.33600**

Improving Zero-shot CoT

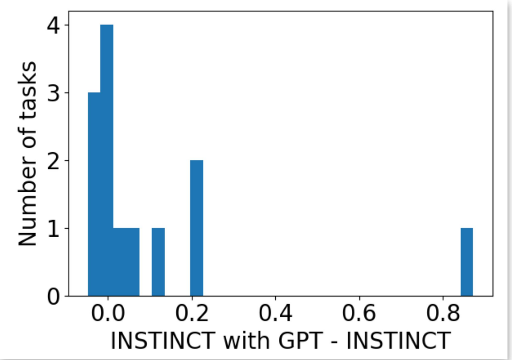
- An well-known zero-shot instruction for chain-of-thought (CoT) reasoning form [1] is “Let’s think step by step.”
- INSTINCT finds better ones:

Method	Dataset	Best Zero-Shot CoT Instruction	Score
Kojima et al. (2022)	GSM8K	Let’s think step by step.	0.71797
InstructZero	GSM8K	Let’s use the instruction to solve the problem.	0.74299
INSTINCT (ours)	GSM8K	**Let’s think about it.**	**0.74526**
Kojima et al. (2022)	AQUA-RAT	Let’s think step by step.	0.52362
InstructZero	AQUA-RAT	Let’s break down the problem.	0.54331
INSTINCT (ours)	AQUA-RAT	**I have a new solution.**	**0.54724**
Kojima et al. (2022)	SVAMP	Let’s think step by step.	0.7625
InstructZero	SVAMP	Let’s use the equation.	0.795
INSTINCT (ours)	SVAMP	**Let’s use our brains.**	**0.81**

[1] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In Proc. NeurIPS, 2022.

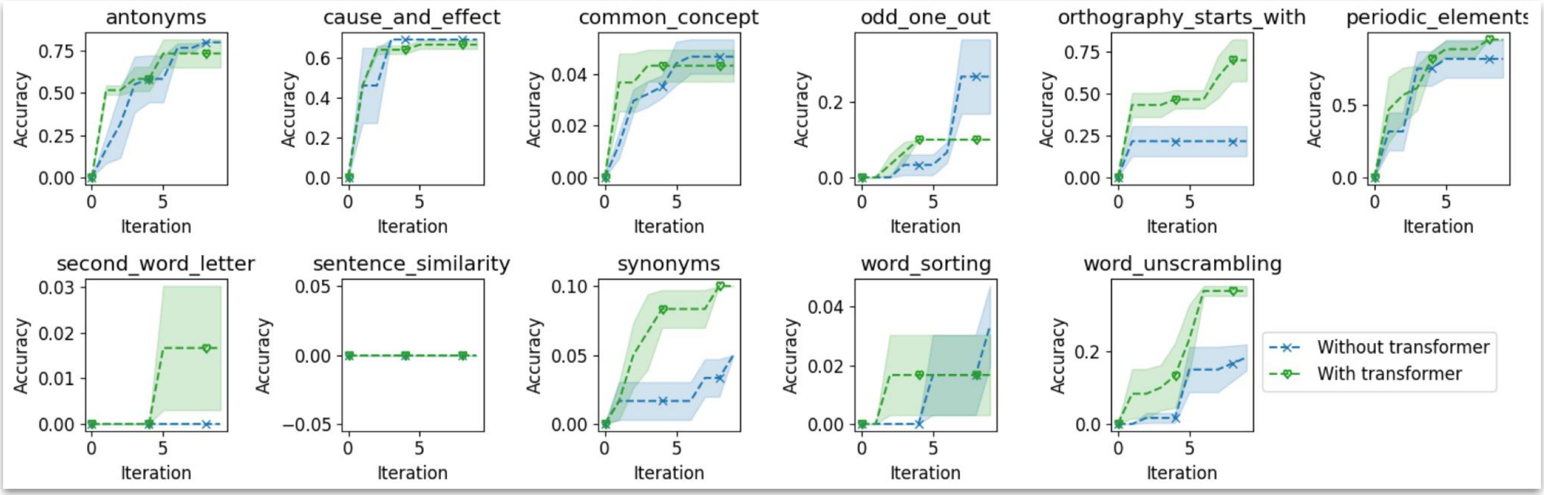
“We could further improve INSTINCT by asking GPT to rephrase for us”

- [1] proposed an *“instruction resampling”* technique for instruction induction
- Following the same spirit, we firstly pass the instruction to ChatGPT and instruct it to rephrase for us
- Experiments on difficult tasks



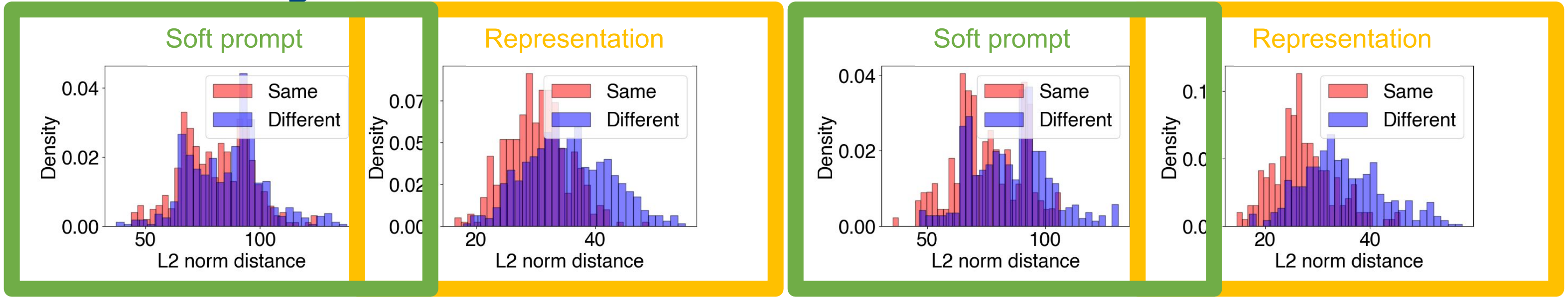
[1] Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. In Proc. ICLR, 2023.

“The hidden representation from the pre-trained transformer is effective”



- The use of the hidden representation allows our NN surrogate to quickly learn to accurately predict the scores and hence achieve high accuracies

"The hidden representation gives a better similarity measure"



activte_to_passive

first_word_letter

- Red group: Soft prompts that map to the ***same*** instruction
- Blue group: Soft prompts that map to ***different*** instructions
- We compute the pairwise L2 distance between both the original ***soft prompts*** and their hidden ***representations***
- InstructZero relies on Matérn kernel which solely relies on L2 distance

Conclusion

- We introduced the INSTINCT to optimize task-specific instructions for black-box LLMs

- Our INSTINCT
  - replaces the GP surrogate in BO by an NN while preserving BO's ability to handle ***exploration v.s. exploitation***
  - leverages the ***expressive power*** of a pre-trained transformer by coupling the NN surrogate with the hidden representation learned by the transformer
  - achieved exceptional performance across extensive empirical evaluations

Prompt Optimization with EASE? Efficient Ordering-aware Automated Selection of Exemplars

Zhaoxuan Wu*, Xiaoqiang Lin*, Zhongxiang Dai, Wenyang Hu, Yao Shu, See-Kiong Ng, Patrick Jaillet, Bryan Kian Hsiang Low

In NeurIPS 2024

Motivation

- In-context learning (ICL): LLM learns from the input-label demonstrations/exemplars in the prompt. The prompt consists of several exemplars and an instruction
- ICL performance is heavily dependent on the selection of exemplars and instructions

Challenges

- Only black-box access to the best LLMs
- Query to black-box LLMs is expensive
- Combinatorial optimization problem with a large search space
  - Retrieval based methods avoid this problem by ignoring ordering
- Best exemplars change when the instruction changes

We propose a query-efficient ordering-aware exemplar selection method that is able to optimize instruction and exemplars jointly

03 Formulation

LLM inference: $\hat{y} = f([\underbrace{e_1, e_2, \ldots, e_k}_{\text{context}}, x]) = f([E, x])$.

$E = (e_1, e_2, \ldots, e_k)$ is an ordered sequence of exemplars

Optimization objective: $\max_{E \in \Omega} F(E) \triangleq \mathbb{E}_{(x,y) \in D_V} [s(f(E, x), y)]$

Let's say we want to select a sequence of 5 exemplars from an exemplar dataset of size 1000. Size of the search space is A_5^{1000}

03

Our EASE algorithm - Reducing search space through optimal transport

$$OT(\mu_s, \mu_v) = \min_{\pi \in \Pi(\mu_s, \mu_v)} \int_{\mathcal{Z}^2} c(z, z') d\pi(z, z')$$

- Intuition: a subset of exemplars that is closer to the validation dataset is more helpful for the task
- Why OT?
  - OT is shown to be useful in data selection work in ML [1]
  - OT takes data diversity into consideration

[1] Just, H. A., Kang, F., Wang, J. T., Zeng, Y., Ko, M., Jin, M., & Jia, R. (2023). Lava: Data valuation without pre-specified learning algorithms. ICLR 2023

03 Our EASE algorithm - NeuralUCB

- NeuralUCB is a query-efficient black-box optimization algo which selects a prompt to query at each iteration
- Uses m() – an NN – to model the mapping from prompt E to performance
- **NeuralUCB** algorithm select the next prompt to query:

$$E_t = \arg\max_{E \in \Omega} \text{NeuralUCB}_t(E),$$

$$\text{NeuralUCB}_t(E) \triangleq \boxed{m(h(E); \theta_t)} + \nu_t \boxed{\sigma_{t-1}(h(E); \theta_t)},$$

Exploitation: the predicted performance of the prompt

Exploration: the uncertainty of the predicted performance

03 Our EASE algorithm - Jointly optimize instruction and exemplars

- Our framework allows us to naturally include instruction p to define a new search space
- This new search space allows us to find a optimal combination of exemplars and instruction

$$E = (p, e_1, e_2, \dots, e_k)$$
$$Q'_t \leftarrow P \times Q'_t$$

03 Experimental results

"Our algorithm outperforms existing retrieval-based algorithms and evolutionary algorithm"

Table 1: Average accuracy ± standard error achieved by the best exemplar sequence discovered by different algorithms over 3 independent trials. For better distinguishability, we do not include easy tasks here (i.e., with 100% accuracy across baselines) and show full results in Tab. 5 of App. C.1.

	DPP	**MMD**	**OT**	**Cosine**	**BM25**	**Active**	**Inf**	**Evo**	**Best-of-N**	EASE
antonyms	70.0±0.0	80.0±0.0	81.7±1.4	85.0±0.0	85.0±0.0	80.0±0.0	86.7±1.4	88.3±1.4	**90.0±0.0**	**90.0±0.0**
auto_categorization	3.3±1.4	8.3±1.4	0.0±0.0	25.0±0.0	16.7±1.4	10.0±2.4	21.7±1.4	21.7±1.4	20.0±0.0	**30.0±0.0**
diff	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**
larger_animal	70.0±0.0	91.7±1.4	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**	66.7±1.4	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**
negation	**95.0±0.0**	**95.0±0.0**	**95.0±0.0**	**95.0±0.0**	**95.0±0.0**	**95.0±0.0**	**95.0±0.0**	**95.0±0.0**	**95.0±0.0**	**95.0±0.0**
object_counting	55.0±2.4	56.7±1.4	48.3±1.4	61.7±1.4	66.7±1.4	51.7±1.4	63.3±3.6	70.0±0.0	70.0±0.0	**73.3±1.4**
orthography_starts_with	20.0±2.4	35.0±0.0	61.7±1.4	**78.3±1.4**	70.0±0.0	43.3±1.4	70.0±2.4	75.0±0.0	**78.3±1.4**	**78.3±1.4**
rhymes	60.0±0.0	51.7±1.4	0.0±0.0	**100.0±0.0**	80.0±0.0	65.0±8.2	70.0±10.8	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**
second_word_letter	10.0±2.4	30.0±0.0	28.3±1.4	**50.0±0.0**	**50.0±0.0**	26.7±8.3	40.0±0.0	46.7±1.4	**50.0±0.0**	**50.0±0.0**
sentence_similarity	20.0±0.0	21.7±2.7	40.0±2.4	46.7±1.4	53.3±1.4	5.0±4.1	18.3±5.4	45.0±0.0	51.7±1.4	**56.7±1.4**
sentiment	85.0±0.0	90.0±0.0	85.0±0.0	96.7±1.4	**100.0±0.0**	85.0±4.1	91.7±1.4	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**
sum	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**
synonyms	10.0±0.0	25.0±0.0	20.0±0.0	**35.0±0.0**	30.0±0.0	3.3±1.4	26.7±1.4	30.0±0.0	30.0±0.0	30.0±0.0
taxonomy_animal	43.3±3.6	40.0±2.4	46.7±1.4	85.0±2.4	80.0±0.0	45.0±6.2	70.0±4.1	80.0±0.0	80.0±0.0	**88.3±2.7**
translation_en-de	**90.0±0.0**	80.0±0.0	80.0±0.0	**90.0±0.0**	85.0±0.0	56.7±13.0	**90.0±0.0**	**90.0±0.0**	**90.0±0.0**	**90.0±0.0**
translation_en-es	90.0±0.0	**100.0±0.0**	96.7±1.4	**100.0±0.0**	**100.0±0.0**	96.7±1.4	98.3±1.4	**100.0±0.0**	**100.0±0.0**	**100.0±0.0**
translation_en-fr	76.7±1.4	76.7±1.4	81.7±1.4	85.0±0.0	85.0±0.0	81.7±1.4	85.0±0.0	86.7±1.4	85.0±0.0	**88.3±1.4**
word_sorting	26.7±1.4	88.3±1.4	88.3±1.4	90.0±0.0	71.7±1.4	80.0±0.0	88.3±1.4	**93.3±1.4**	91.7±1.4	91.7±1.4
word_unscrambling	68.3±1.4	56.7±1.4	71.7±1.4	75.0±0.0	76.7±1.4	63.3±3.6	66.7±1.4	75.0±0.0	75.0±0.0	**78.3±2.7**
# best-performing tasks	2	2	2	8	5	1	5	9	11	**17**

03

Experimental results

When does selection of exemplars important? “When the LLM has not seen the task in its training dataset”

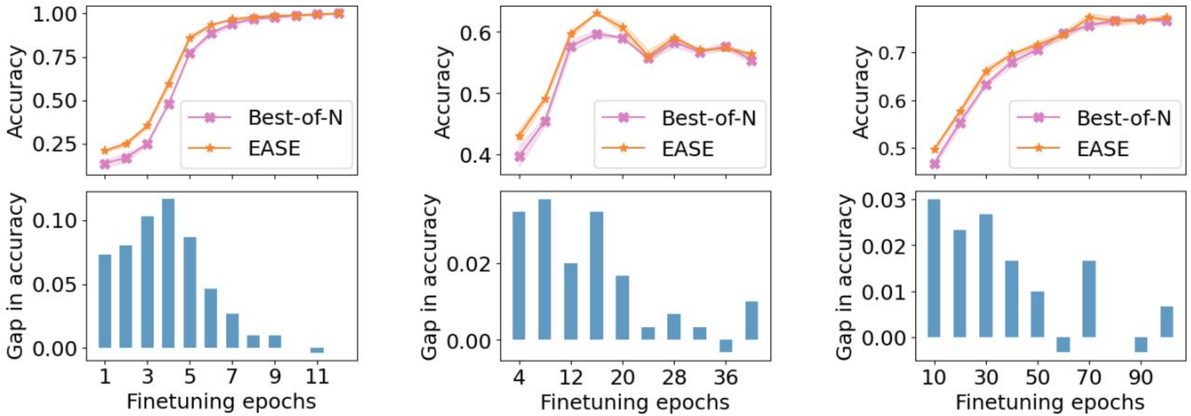


Figure 1: From left to right, the tasks are taxonomy animal, sentence similarity and object counting. The performance gaps between EASE and the Best-of-N baseline diminish as the LLM is finetuned.

03 Experimental results

“Selection of exemplars has larger impact on the performance in unseen tasks for LLM”

Table 2: Average accuracy ± standard error over 3 independent trials achieved by different algorithms on the new families of out-of-distribution tasks.

Type	Task	Noise	DPP	MMD	OT	Cosine	BM25	Active	Inf	Evo	Best-of-N	EASE
Rule-based tasks	LR	0%	31.7±1.4	38.3±2.7	50.0±0.0	71.7±1.4	70.0±0.0	36.7±1.4	56.7±5.9	61.7±1.4	66.7±1.4	**81.7±3.6**
		10%	8.3±1.4	36.7±1.4	48.3±1.4	61.7±1.4	61.7±1.4	0.0±0.0	58.3±3.6	60.0±0.0	65.0±2.4	**73.3±3.6**
		30%	10.0±0.0	28.3±1.4	46.7±1.4	63.3±1.4	60.0±0.0	40.0±2.4	35.0±2.4	53.3±1.4	50.0±0.0	**78.3±1.4**
		50%	0.0±0.0	38.3±1.4	45.0±0.0	65.0±0.0	53.3±1.4	0.0±0.0	53.3±1.4	46.7±1.4	45.0±0.0	**71.7±2.7**
		70%	0.0±0.0	55.0±0.0	38.3±2.7	65.0±0.0	50.0±0.0	26.7±5.4	30.0±4.7	33.3±1.4	33.3±1.4	**66.7±3.6**
		90%	0.0±0.0	21.7±1.4	26.7±1.4	46.7±1.4	3.3±1.4	0.0±0.0	6.7±2.7	8.3±1.4	15.0±0.0	**53.3±2.7**
	LP-variant	0%	48.3±2.7	40.0±2.4	41.7±1.4	65.0±0.0	58.3±1.4	30.0±0.0	61.7±1.4	75.0±2.4	71.7±1.4	**75.0±0.0**
		10%	0.0±0.0	36.7±1.4	40.0±0.0	63.3±2.7	60.0±0.0	36.7±2.7	65.0±2.4	70.0±2.4	73.3±1.4	**75.0±2.4**
		30%	0.0±0.0	48.3±2.7	40.0±2.4	60.0±0.0	55.0±0.0	40.0±7.1	53.3±4.9	65.0±2.4	65.0±0.0	**73.3±1.4**
		50%	0.0±0.0	65.0±0.0	35.0±2.4	63.3±2.7	60.0±0.0	38.3±3.6	48.3±3.6	61.7±1.4	65.0±0.0	**76.7±2.7**
		70%	0.0±0.0	46.7±2.7	35.0±0.0	70.0±0.0	60.0±0.0	25.0±8.2	60.0±4.1	56.7±1.4	56.7±1.4	**75.0±0.0**
		90%	0.0±0.0	35.0±2.4	50.0±0.0	**65.0±2.4**	0.0±0.0	30.0±12.5	50.0±2.4	38.3±1.4	55.0±2.4	63.3±1.4
Re-mapped label tasks	AG News Remap	0%	20.0±2.4	15.0±0.0	26.7±1.4	43.3±1.4	43.3±2.7	5.0±2.4	25.0±4.1	40.0±0.0	40.0±0.0	**53.3±3.6**
		10%	5.0±0.0	15.0±0.0	15.0±0.0	41.7±1.4	38.3±1.4	3.3±1.4	26.7±2.7	36.7±1.4	40.0±0.0	**56.7±2.7**
		30%	10.0±0.0	5.0±0.0	5.0±0.0	40.0±0.0	36.7±1.4	1.7±1.4	10.0±0.0	40.0±0.0	43.3±1.4	**51.7±1.4**
		50%	5.0±0.0	10.0±0.0	5.0±0.0	43.3±1.4	35.0±0.0	3.3±1.4	20.0±4.1	35.0±0.0	35.0±0.0	**56.7±1.4**
		70%	5.0±0.0	25.0±0.0	8.3±1.4	50.0±0.0	35.0±0.0	1.7±1.4	11.7±5.4	38.3±1.4	46.7±1.4	**51.7±1.4**
		90%	5.0±0.0	18.3±1.4	5.0±0.0	40.0±0.0	10.0±0.0	15.0±6.2	35.0±0.0	35.0±0.0	41.7±1.4	**55.0±2.4**
	SST5 Reverse	0%	20.0±0.0	10.0±0.0	13.3±1.4	40.0±0.0	40.0±0.0	15.0±2.4	33.3±5.4	35.0±2.4	40.0±0.0	**50.0±0.0**
		10%	16.7±1.4	10.0±0.0	15.0±0.0	48.3±1.4	40.0±0.0	13.3±2.7	23.3±5.4	33.3±2.7	40.0±0.0	**50.0±0.0**
		30%	23.3±1.4	6.7±1.4	25.0±2.4	40.0±0.0	40.0±0.0	21.7±3.6	26.7±1.4	30.0±0.0	31.7±1.4	**41.7±3.6**
		50%	21.7±1.4	15.0±0.0	15.0±0.0	**43.3±1.4**	33.3±1.4	21.7±1.4	23.3±1.4	28.3±1.4	30.0±0.0	**43.3±1.4**
		70%	25.0±0.0	23.3±1.4	23.3±1.4	40.0±0.0	30.0±0.0	20.0±2.4	25.0±2.4	36.7±1.4	36.7±1.4	**45.0±2.4**
		90%	20.0±0.0	15.0±2.4	20.0±0.0	30.0±0.0	30.0±0.0	13.3±2.7	21.7±1.4	30.0±0.0	30.0±0.0	**31.7±1.4**

03 Experimental results

"Joint optimization of exemplars and instruction improves over only exemplars optimization significantly"

Table 3: Average accuracy ± s.e. for EASE with and without jointly optimized instructions. We removed tasks with 100% accuracy. The full results are in App C, Tab. 6.

	EASE	EASE with instructions	Improvement
antonyms	**90.0**±0.0	85.0±0.0	-5.0 ↓
auto_categorization	30.0±0.0	**46.7**±4.9	16.7 ↑
negation	95.0±0.0	**100.0**±0.0	5.0 ↑
object_counting	73.3±1.4	**75.0**±0.0	1.7 ↑
orthography_starts_with	78.3±1.4	**81.7**±1.4	3.3 ↑
rhymes	**100.0**±0.0	91.7±3.6	-8.3 ↓
second_word_letter	50.0±0.0	**100.0**±0.0	50.0 ↑
sentence_similarity	**56.7**±1.4	**56.7**±1.4	0.0 ○
synonyms	**30.0**±0.0	**30.0**±0.0	0.0 ○
taxonomy_animal	88.3±2.7	**100.0**±0.0	11.7 ↑
translation_en-de	**90.0**±0.0	**90.0**±0.0	0.0 ○
translation_en-es	**100.0**±0.0	**100.0**±0.0	0.0 ○
translation_en-fr	**88.3**±1.4	85.0±0.0	-3.3 ↓
word_sorting	**91.7**±1.4	**91.7**±1.4	0.0 ○
word_unscrambling	78.3±2.7	**80.0**±0.0	1.7 ↑
linear_4_10_noisy	**73.3**±3.6	41.7±9.5	-31.7 ↓
LP-variant (10% noise)	75.0±2.4	**85.0**±2.4	10.0 ↑
AG News Remap (10% noise)	56.7±2.7	**65.0**±0.0	8.3 ↑
SST5 Reverse (10% noise)	**50.0**±0.0	**50.0**±0.0	0.0 ○

03 Experimental results

"Our algorithm can leverage the existing retrieval-based methods to scale to larger exemplar domains"

Table 4: Average accuracy $\pm$ s.e. achieved by EASE and EASE with retrieval for larger exemplar set sizes.

AG News Remap (10% noise)

Size n	EASE	EASE with retrieval
1000	41.7$\pm$1.4	**63.3**$\pm$1.4
10000	55.0$\pm$2.4	**65.0**$\pm$0.0
50000	56.7$\pm$3.6	**63.3**$\pm$1.4
100000	50.0$\pm$2.4	**65.0**$\pm$0.0

SST5 Reverse (10% noise)

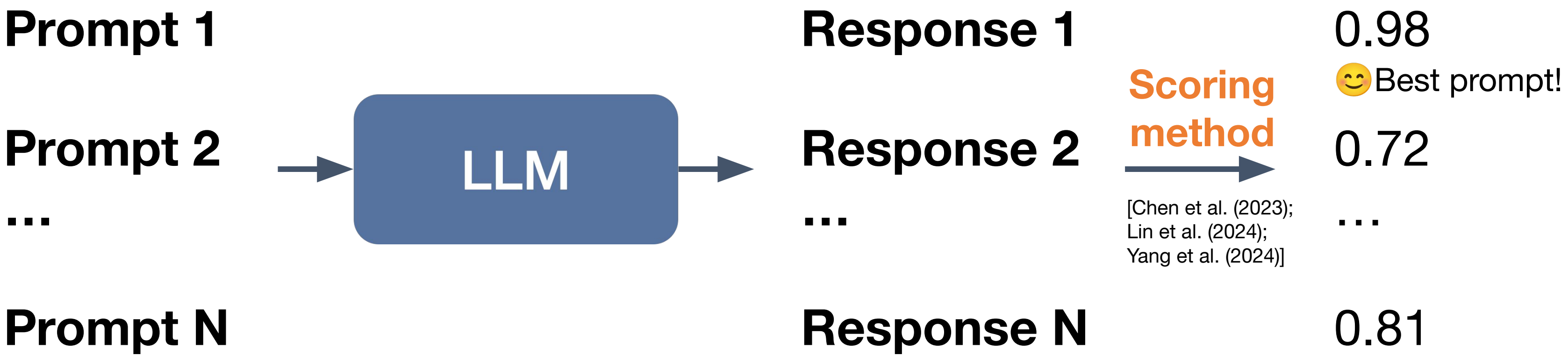
Size n	EASE	EASE with retrieval
1000	46.7$\pm$1.4	**55.0**$\pm$3.5
3000	42.5$\pm$1.8	**51.7**$\pm$1.4
5000	43.3$\pm$1.4	**45.0**$\pm$0.0
7000	43.3$\pm$1.4	**50.0**$\pm$0.0

Prompt Optimization with Human Feedback

Xiaoqiang Lin, Zhongxiang Dai, Arun Verma, See-Kiong Ng, Patrick Jaillet, Bryan Kian Hsiang Low

In ICML 2024, Workshop on Models of Human Feedback for AI Alignment, Oral Presentation

Prompt Optimization with Scoring Functions



Motivations

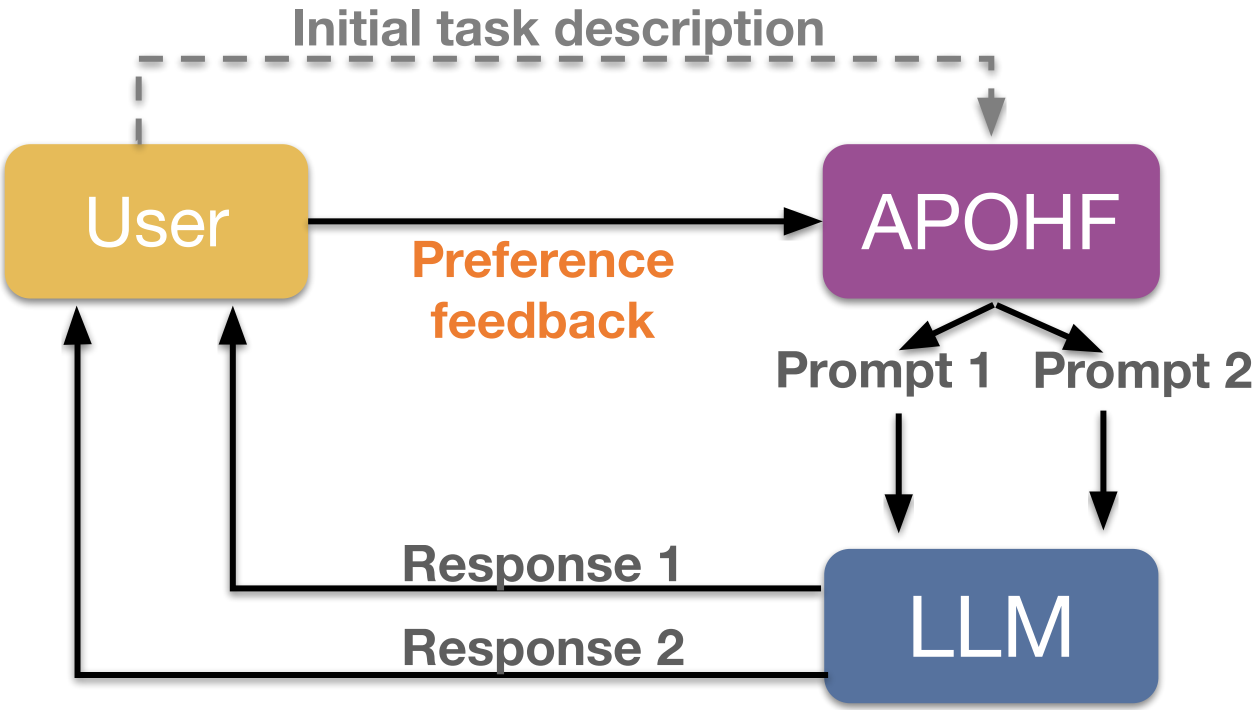
A scoring method may not be available or reliable

- No validation dataset available
- A scorer LLM may not be accurate
- Human is not good at giving a score (Yue et al. 2012)

Human is more reliable at providing preference feedback (Yue et al. 2012)

Can we perform prompt optimization using only human preference feedback?

Prompt Optimization with Human Feedback



Our algorithm - APOHF

- Using the neural network for latent score prediction
  - $h(x;\theta)$ mapping from prompt to latent score
- Preference feedback model - Bradley-Terry-Luce (BTL) model (Hunter et al. 2004)

$$P(\mathrm{x}_1 \succ x_2) = \sigma(h(x_1;\theta) - h(x_2;\theta))$$

- Given the previous feedback $\boldsymbol{D}_{t-1} = \{\boldsymbol{x}_{s,1}, \boldsymbol{x}_{s,2}, \boldsymbol{y}_s\}_{\boldsymbol{s=1} \ldots \boldsymbol{t-1}}$, train the NN ($h$) by minimizing the following loss function:

$$\ell(\theta) = -likelihood\left(y, \sigma\left(h(x_1;\theta) - h(x_2;\theta)\right)\right) + \lambda||\theta||$$

Our algorithm - APOHF

- Selection of first prompt:

$$\mathbf{x}_{t,1} = \arg\max_{x} \boldsymbol{h}(\boldsymbol{x}; \boldsymbol{\theta}_t)$$

- Selection of second prompt:

$$\mathbf{x}_{t,2} = \arg\max_{x} \boxed{\boldsymbol{h}(\boldsymbol{x}; \theta_t)} + \boldsymbol{\nu} \boxed{||\nabla \mathbf{h}(\mathbf{x}; \boldsymbol{\theta}_t) - \nabla \mathbf{h}(\mathbf{x}_{t,1}; \boldsymbol{\theta}_t)||_{V_{t-1}^{-1}}}$$

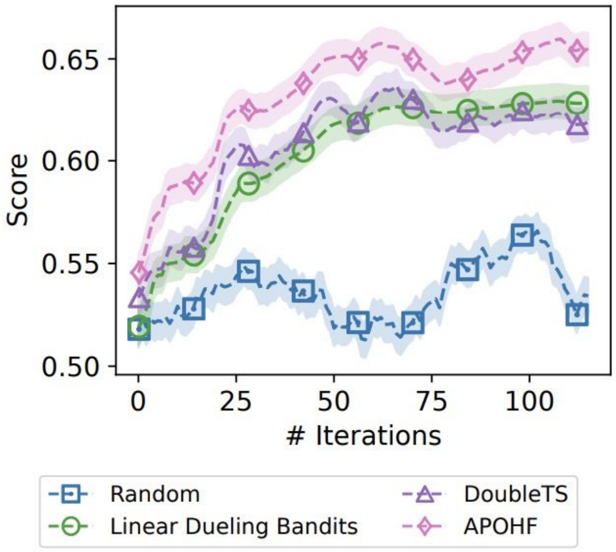
Exploitation: Score prediction

Exploration: Encourages $\mathrm{x}_{t,2}$ to be different from the previously selected prompts and $\mathrm{x}_{t,1}$

(Arun et al. (2024))

Verma, Arun, Zhongxiang Dai, Xiaoqiang Lin, Patrick Jaillet, and Bryan Kian Hsiang Low. “Neural Dueling Bandits.” *ICML 2024 Workshop RLControlTheory*

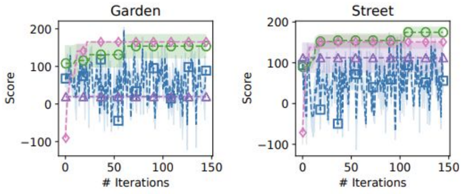
Experiment - Optimization of User Instructions



Task	Iter	Instruction	Score
antonyms	0	add the prefix "un-" to the given words to form their opposites.	0.45
	5	remove the "un-" prefix from each word.	0.45
	10	provide the opposite of the given words.	0.70

Our APOHF consistently and significantly outperforms the other methods.

Experiment - Prompt Optimization for Text-to-Image Generative Models



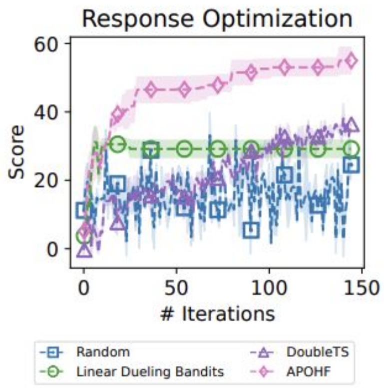
Our APOHF can efficiently produce images which better align with the image the user has in mind.

Experiment - Response Optimization with Human Feedback

The prompt is “Human: What is there to do in Atlantic City?”.

Iter	Response	Score
0	There are many activities to do in Atlantic City! You can visit the famous Atlantic City Boardwalk, try your luck at the casinos, enjoy the beach and water activities, shop at the outlets and specialty stores, dine at top-rated restaurants, enjoy live entertainment and nightlife, visit museums and historical sites, and attend events and festivals throughout the year.	81.96
10	There are plenty of things to do in Atlantic City! You can visit the famous boardwalk, try your luck at the casinos, enjoy a show or concert at one of the many entertainment venues, dine at a variety of restaurants, shop at the outlets and specialty stores, relax on the beach, or take a stroll through the historic district. There’s something for everyone in Atlantic City!	91.46
20	There are many attractions and activities in Atlantic City, including: 1. Casinos: Atlantic City is known for its many casinos, where you can try your luck at slots, poker, blackjack, and more. 2. Boardwalk: ... 3. Beach: .. 4. Steel Pier: ... 5. Shows and concerts: ... 6. Nightlife: ... 7. Dining: ... 8. Shopping: ...	180.14

Our APOHF is able to further refine the response of an LLM to make it more preferable for human users.



Agenda

What is prompt optimization

What are the challenges

What's next

What's next in prompt optimization

1. Current PO methods require optimization for a specific LLM.
2. PO method require expensive optimization for each task.
3. The generation of the prompt domain is vital to the PO performance.
4. Applications for PO: agentic AI, improve reasoning by PO, etc.

Q & A

- Any questions?

THANK YOU

My Other Related Works

[1] Hu, W., Shu, Y., Yu, Z., Wu, Z., Lin, X., Dai, Z., ... & Low, B. K. H. (2024). Localized zeroth-order prompt optimization. NeurIPS 2024 Spotlight.
[2] Zhou, Z., Lin, X., Xu, X., Prakash, A., Rus, D., & Low, B. K. H. (2024). DETAIL: Task DEmonsTration Attribution for Interpretable In-context Learning. NeurIPS 2024.
[3] Xu, X., Wu, Z., Qiao, R., Verma, A., Shu, Y., Wang, J., ... & Low, B. K. H. (2024, November). Position Paper: Data-Centric AI in the Age of Large Language Models. EMNLP findings.
[4] Wang, J., Lin, X., Qiao, R., Foo, C. S., & Low, B. K. H. (2024). Helpful or Harmful Data? Fine-tuning-free Shapley Attribution for Explaining Language Model Predictions. ICML 2024.