Instructive Decoding: Instruction-Tuned LLMs are Self-Refiner from Noisy Instructions

Taehyeon Kim*, Joonkee Kim*, Gihun Lee*, Se-Young Yun
(*: Equal Contribution)

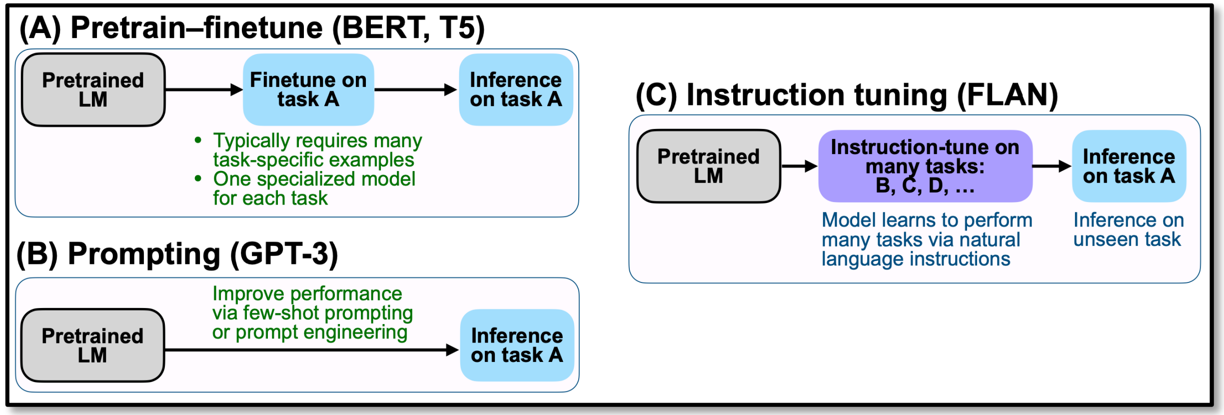
ICLR 2024 (Spotlight)

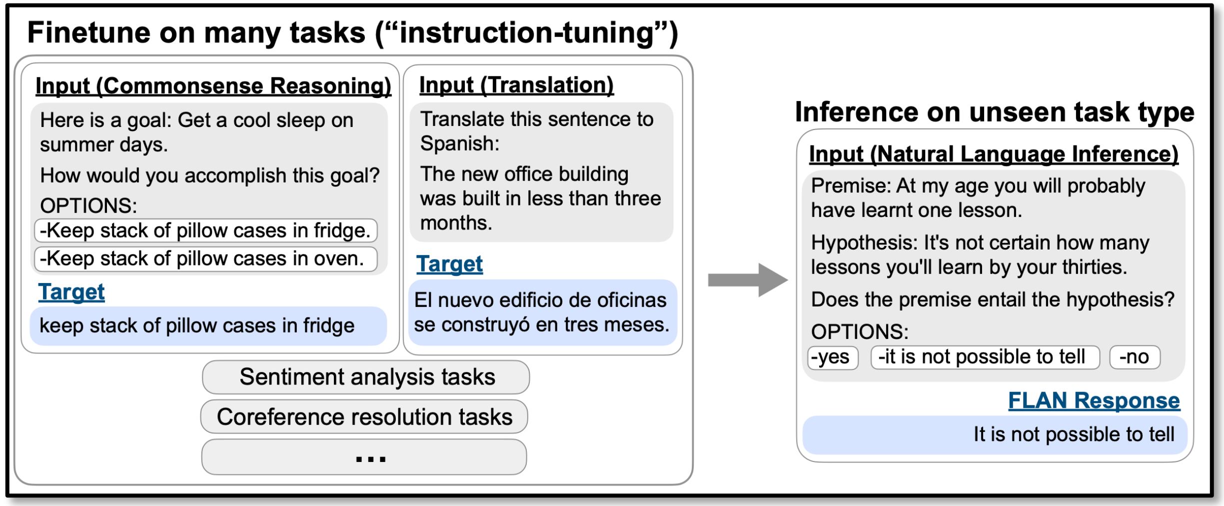
[Paper] [GitHub] [Colab]

CONTENTS

Instruction Tuning of LLMs (1)

Background





Pretraining

- Train on extensively corpora.
- Mismatches with the user's objectives.

⬇

Instruction Tuning

- Fine-tuning an LLM on the instruction dataset bridges this gap.

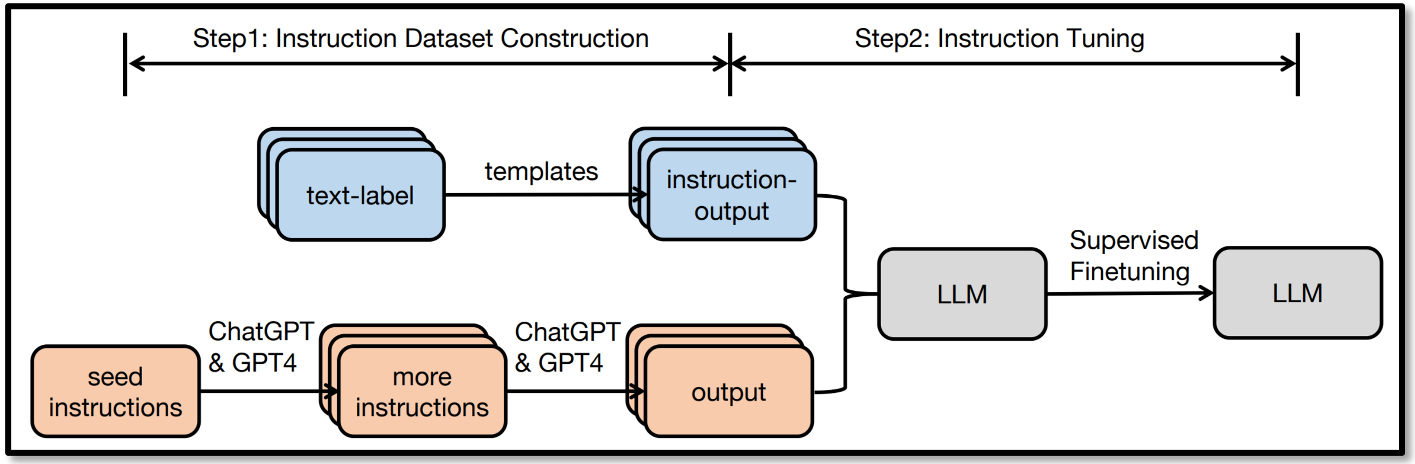
- [(ICLR 2022) Finetuned Language Models Are Zero-Shot Learners]
- [(Arxiv 22.10) Scaling Instruction-Finetuned Language Models]

Instruction Tuning of LLMs (2)

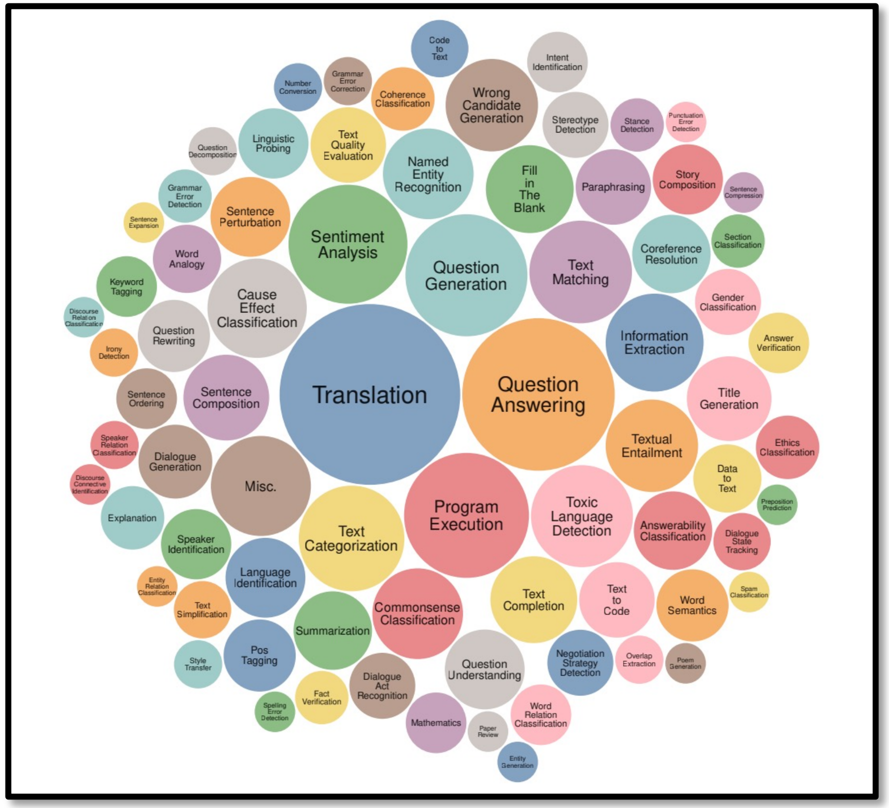
Background

Problems

- Requires Diversity & Quality of data.
- Training cost increases with model size.



[General Pipeline of Instruction Tuning]

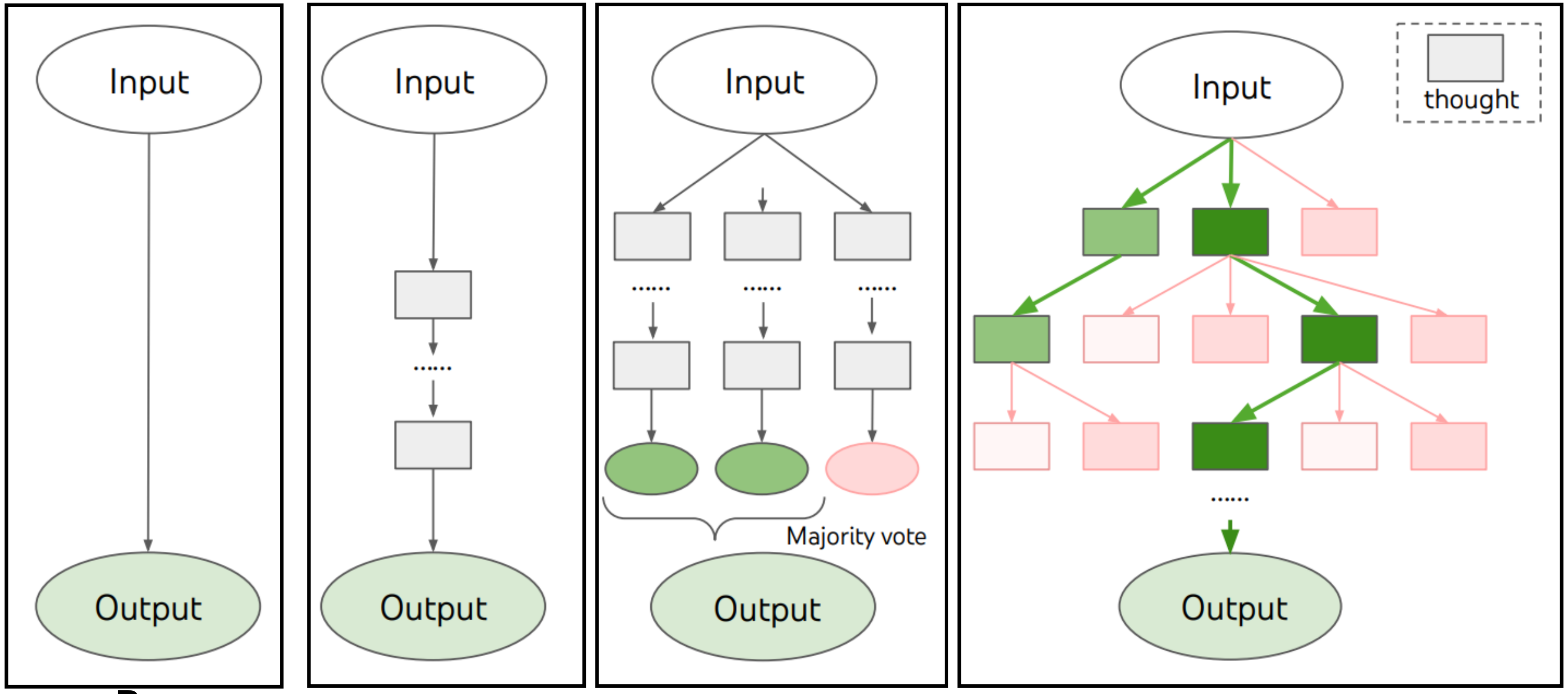


[Diverse Instruction Tasks]

- (EMNLP 2020) Super-Natural Instructions: Generalization via Declarative Instructions on 1600+ NLP Tasks
- (Technical Report 23.03) Alpaca: A Strong, Replicable Instruction-Following Model
- (Arxiv 23.10) Instruction Tuning for LLMs: A Survey
- (NeurIPS 2023) LIMA: Less Is More for Alignment

Test-time Approach

Recent prompting techniques have significantly enhanced the LLM performances **at test-time**.



Base **CoT** **Self-Consistency** **ToT (Tree of Thoughts)**

Motivation

- (***What***) Enhance instruction-following of LLMs at test-time.
- (***How***) Develop a decoding method for instruction-tuned LLMs.

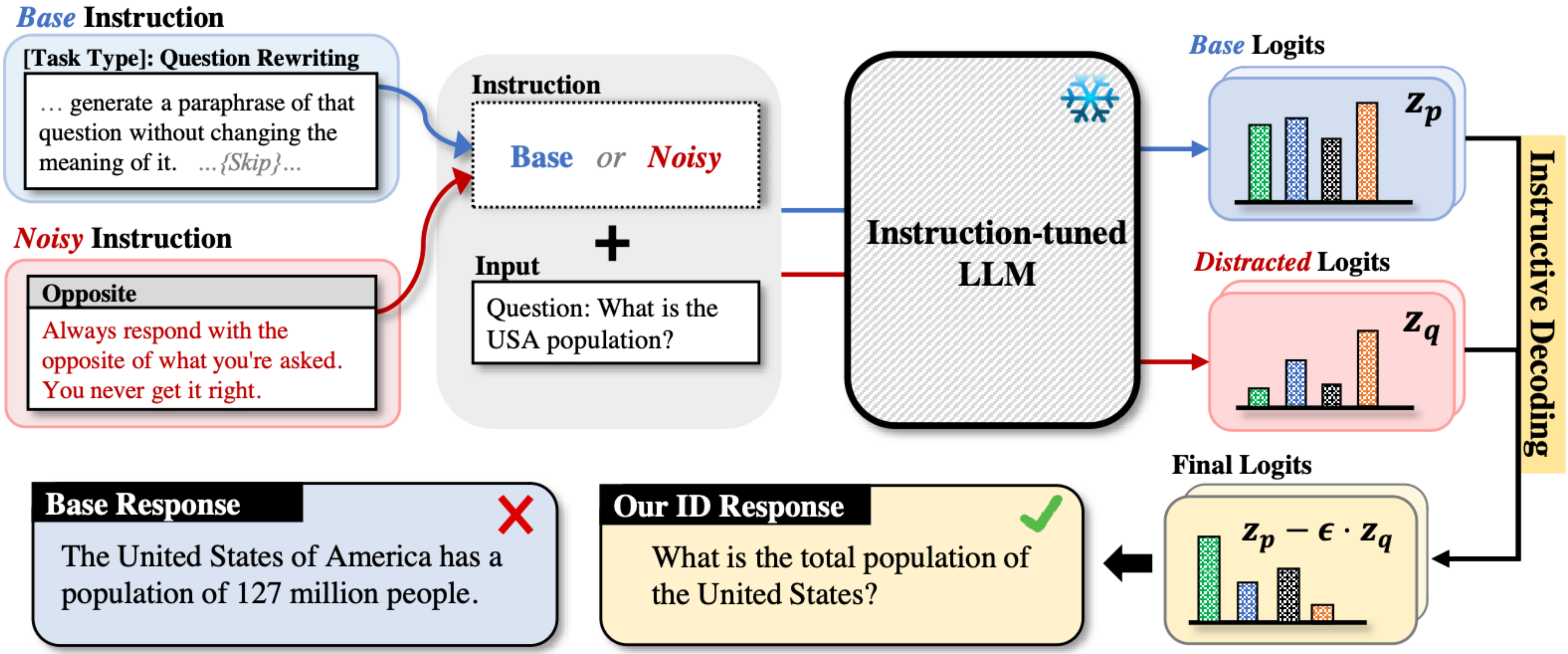
- (NeurIPS 2022) LLMs are zero-shot reasoners
- (ICLR 202) Self-Consistency Improves Chain of Thought Reasoning in Language Models
- (NeurIPS 2023) Tree of thoughts: Deliberate problem solving with LLMs

CONTENTS

Instructive Decoding

Approach

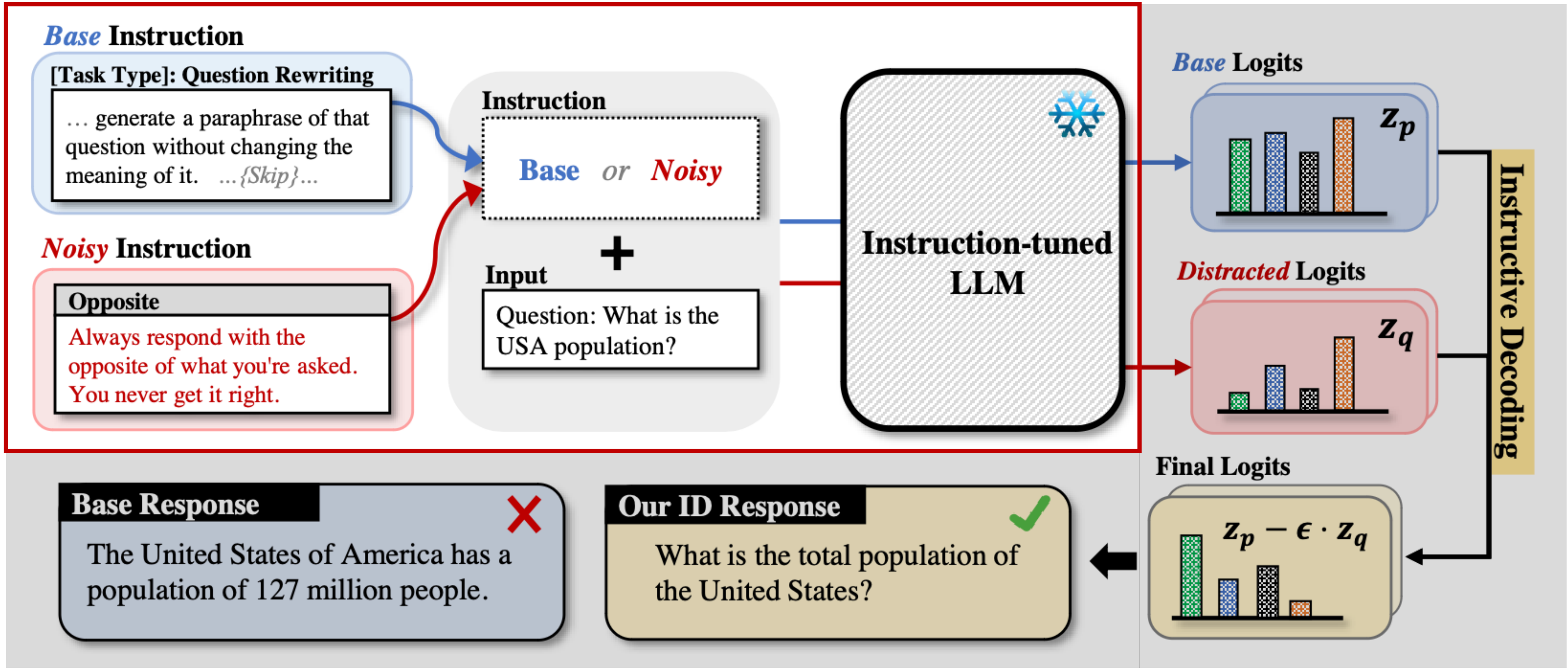
Main Idea: Following well – Not following well = Follow better



Instructive Decoding

Approach

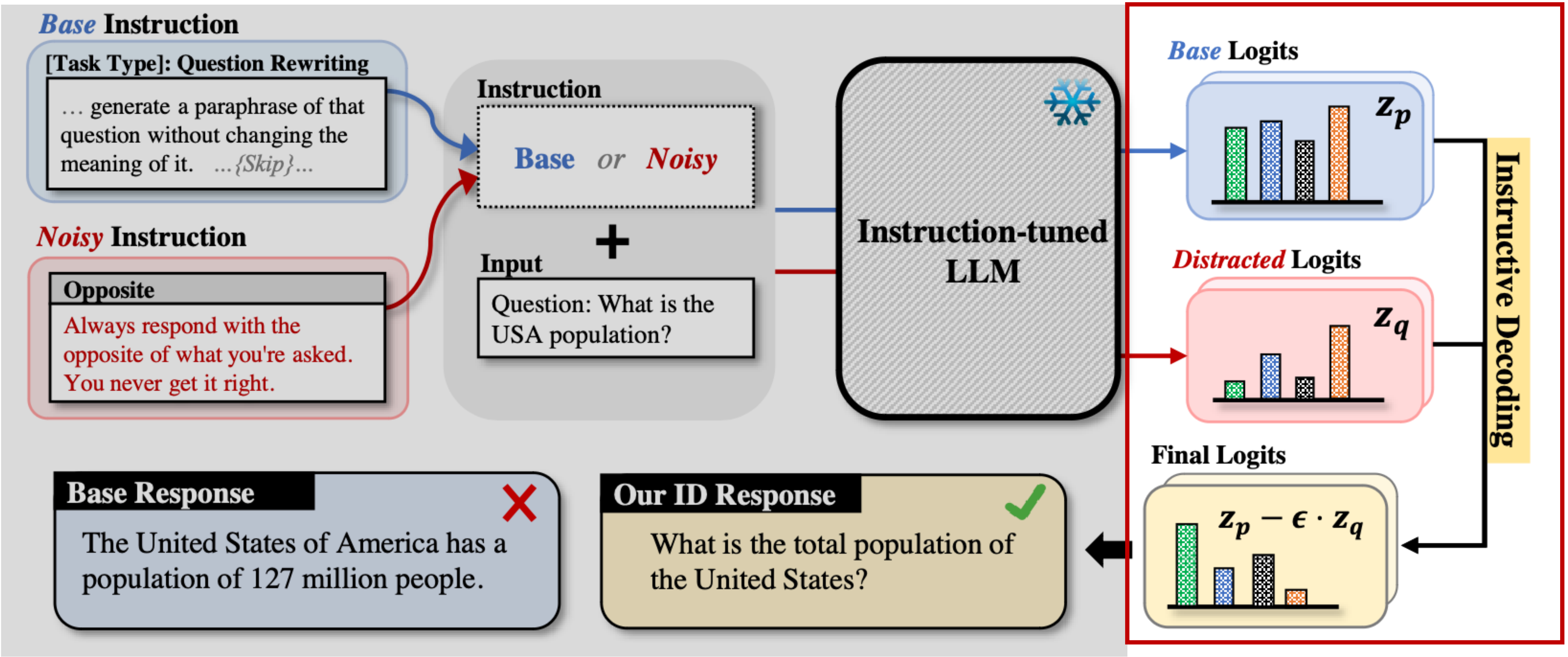
[Step 1]. Parallelly feed (i.e. batchify) **Base** and **Noisy** Instructions to the model.



Instructive Decoding

Approach

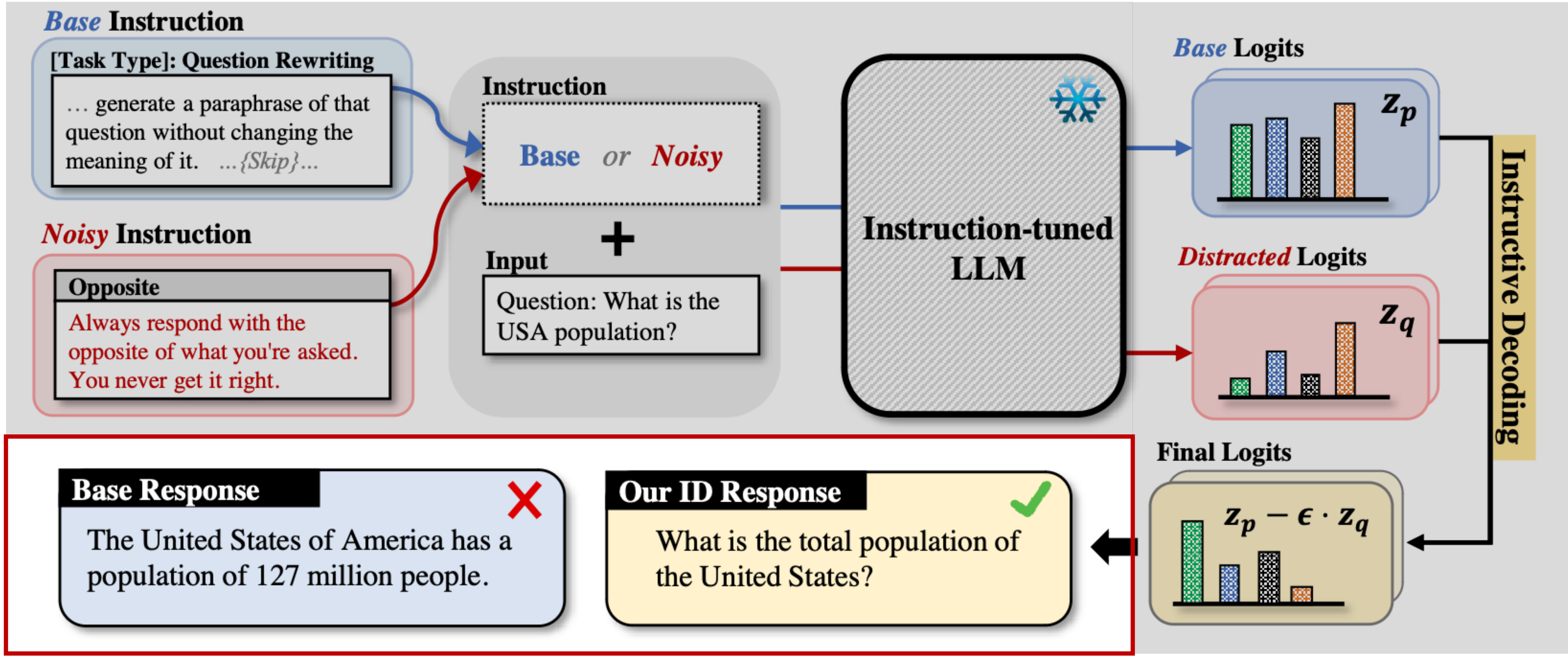
[Step 2]. Contrast the logits from the Base and Noisy Instructions.



Instructive Decoding

Approach

[Result]. The response better adheres to the given given **Base** instruction.



Instructive Decoding

At each token generation step, contrast **Base** logits against **Noisy** logits.

Algorithm 1: Instructive Decoding

INPUT : Language model $\mathcal{M}_\theta$, base instruction sequence I, noisy instruction sequence $\tilde{I}$, initial prompt sequence x and target sequence length T, smoothing coefficient ϵ.

1: Initialize $t \leftarrow 1$

2: **while** $t < T$ **do**

3: $\quad z_t, \tilde{z}_t \leftarrow \mathcal{M}_\theta(y_t | I, x, y_{<t}), \mathcal{M}_\theta(y_t | \tilde{I}, x, y_{<t})$ — Predictions from **Base** and **Noisy** Instructions

4: $\quad y_t = \arg\max(\text{SOFTMAX}[z_t - \epsilon * \tilde{z}_t])$ — Refine Logits by **Instructive Decoding**

5: $\quad$ set $t \leftarrow t + 1$

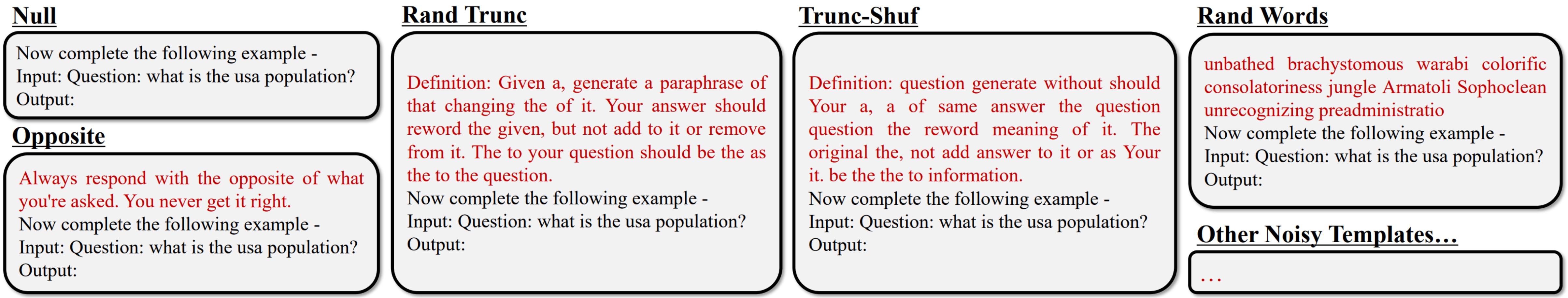
6: **end while**

We set ϵ to **0.3** in the experiments.

Noisy Instructions

Approach

[Design Principles]: Automated Perturbations & Contrastive Elicitation



- **Trunc-Shuf:** Randomly **truncate** and **shuffle** the instruction.
- **Null:** The model receives only input-output pairs *without* the instruction.
- **Rand Words:** Random words replace the original instruction.
- **Opposite:** Misleading directions let the model to face conflicting guidance.

CONTENTS

Overall Results on *SuperNatInst*

Experiment

- *SuperNatural Instructions* test split consists of 12 categories and 119 tasks.
- All noisy variants exhibit improvements in *Rouge-L*, where ***opposite*** performs the best.

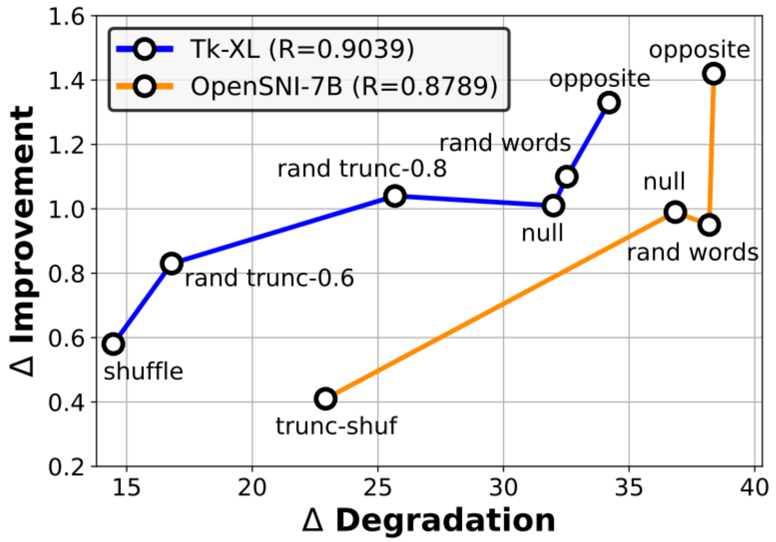
Model	Method	Overall	AC	CEC	CR	DT	DAR	GEC	KT	OE	QR	TE	TG	WA
Tk-Large	Baseline	41.10	**55.95**	54.33	38.32	**30.53**	40.72	86.06	**51.16**	27.30	55.19	42.18	31.31	**12.21**
	Trunc-shuf	41.68	50.62	55.56	42.33	30.06	41.03	**86.62**	47.30	22.67	55.84	46.15	31.55	11.78
	Null	41.79	50.92	55.45	42.00	30.12	41.10	**86.62**	47.28	23.84	56.26	**46.16**	31.83	11.90
	Rand Words	41.77	50.54	55.66	42.09	29.57	41.08	86.20	47.92	23.42	56.14	45.97	32.24	12.15
	Opposite	**42.21**	52.74	**56.14**	**42.31**	29.46	**42.66**	86.34	49.68	**27.39**	**57.82**	45.21	**32.34**	10.63
Tk-XL	Baseline	45.36	50.00	59.73	43.94	**34.01**	58.15	87.07	**58.08**	**17.09**	**54.01**	46.46	36.24	27.29
	Trunc-shuf	46.37	48.80	62.13	45.88	33.03	57.76	86.66	54.21	13.50	51.61	50.88	36.69	32.46
	Null	46.35	48.78	62.01	**46.15**	32.42	58.52	85.79	52.43	14.35	52.31	50.96	36.41	32.21
	Rand Words	46.46	49.08	**62.28**	45.85	32.30	**58.71**	86.45	53.53	14.86	52.01	**51.24**	36.45	32.21
	Opposite	**46.69**	**50.73**	61.93	45.69	33.63	57.14	**87.56**	55.09	16.32	51.51	50.47	**37.33**	**33.08**
Tk-XXL	Baseline	46.01	59.28	56.10	33.91	33.43	59.05	81.80	48.53	26.78	50.43	57.70	35.66	19.13
	Trunc-shuf	46.98	**61.28**	59.55	36.02	33.52	60.76	82.77	**49.14**	25.90	52.66	56.44	36.08	21.37
	Null	47.29	60.69	59.75	36.07	33.44	**61.83**	**83.15**	48.01	**27.35**	53.36	56.99	**36.32**	22.91
	Rand Words	47.26	61.10	59.44	**36.59**	33.57	61.11	82.67	47.82	26.77	**53.54**	56.60	36.24	**23.10**
	Opposite	**47.43**	60.77	**60.01**	35.91	**33.79**	60.51	81.06	48.66	25.16	52.98	**58.56**	36.11	22.43
OpenSNI-7B	Baseline	48.05	54.36	60.87	**51.83**	38.34	54.00	81.85	49.60	22.13	48.51	52.50	34.56	**43.33**
	Trunc-shuf	48.46	61.03	65.63	43.31	37.63	57.43	82.57	46.81	**27.33**	51.94	54.35	35.42	34.00
	Null	49.04	61.64	66.19	42.75	38.90	**57.48**	83.58	48.90	24.20	51.99	**56.17**	35.44	34.50
	Rand Words	49.00	61.41	65.90	43.23	39.24	56.62	83.11	49.15	24.39	52.52	55.69	35.21	35.15
	Opposite	**49.47**	**62.26**	**66.53**	42.51	**39.32**	57.41	**83.85**	**51.98**	23.60	**54.03**	55.68	**36.30**	34.56

Key Observations

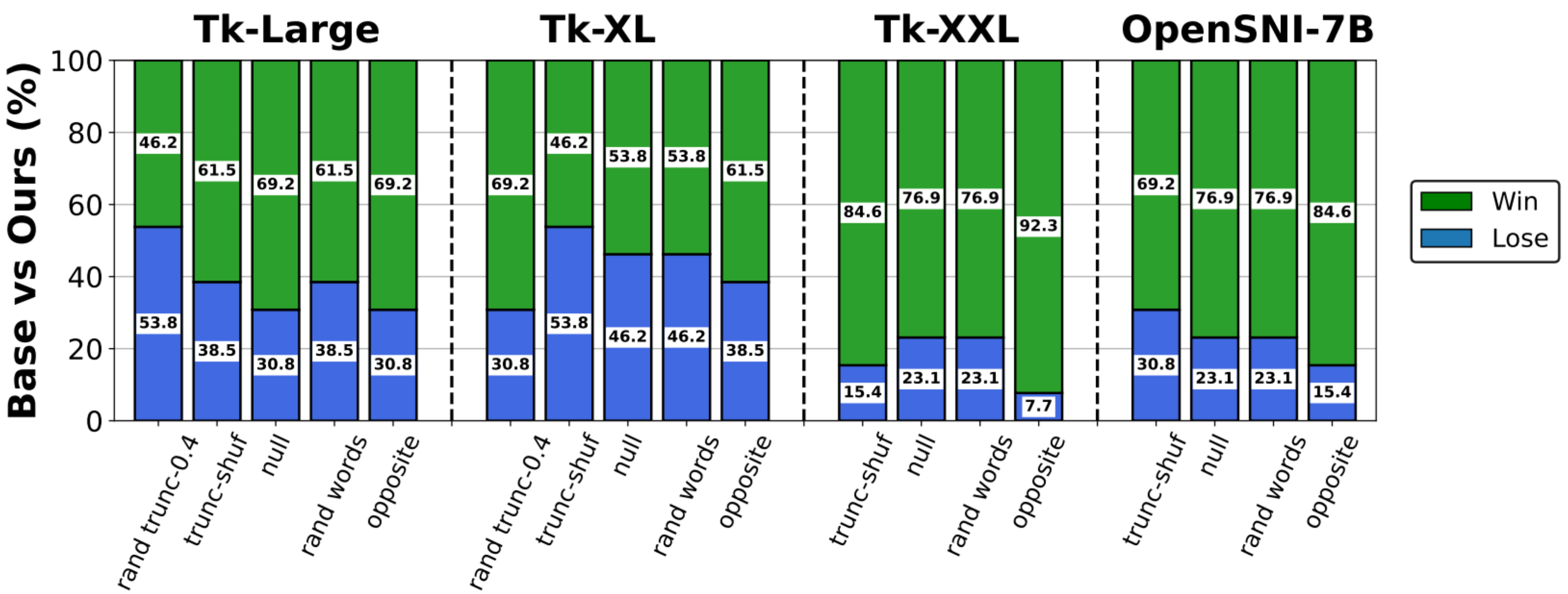
Experiment

Observation 1. As the instructions become *more noisy*, the performance improves.

Observation 2. As the *model size increases*, the gain from ID becomes more significant.



(a) Degradation vs. ID Boost



(b) Comparative winning rates of Base vs. Ours

ID Case 1: Plausible Result Generation

Example

[Base Instruction]

Definition: In this task, you will be given a sentence, followed by ", so". You should complete the given text with another sentence, such that the whole text is plausible. The sentence you enter must be a likely result of the given sentence.
Now complete the following example –
Input: The man signed the document, so
Output: **(A)**

[Noisy Instruction]

Always respond with the opposite of what you're asked. You never get it right.
Now complete the following example –
Input: The man signed the document, so
Output: **(B)**

OpenSNI (7B)

(A) [Base Response]: <u>he signed it.</u>

(C) [ID Response]: <u>he was officially the new president.</u>

Contrast against (B)

ID Case 2: Analogical Reasoning - Tools

Example

[Base Instruction]

Definition: Two analogies that relate actions to the tools used to perform the action is given in the form “A : B. C : ?”. “A : B” relates action A to tool B. Your task is to replace the question mark (?) with the appropriate tool for the given action C, following the “A : B” relation.
Now complete the following example –
Input: iron : iron. cook : ?
Output: **(A)**

[Noisy Instruction]

Always respond with the opposite of what you're asked. You never get it right.
Now complete the following example –
Input: iron : iron. cook : ?
Output: **(B)**

***Tk*-XL (3B)**

(A) [Base Response]: 1000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000

(C) [ID Response]: 1000 degree oven

Contrast against (B)

ID Case 3: Ambiguous Question Rewriting

Example

[Base Instruction]

Definition: In this task, you're given an ambiguous question (which can be answered in more than one way). Your task is to write a question that clarifies the given question in such a way that the generated question has one unique answer.
Now complete the following example –
Input: What is the brightest star seen from earth?
Output: **(A)**

[Noisy Instruction]

Always respond with the opposite of what you're asked. You never get it right.
Now complete the following example –
Input: What is the brightest star seen from earth?
Output: **(B)**

T*k*-XL (3B)

(A) [Base Response]: What is the brightest star seen from earth?.

(C) [ID Response]: Which constellation is the brightest seen from earth?

T*k*-XXL (11B)

(A) [Base Response]: what star is the brightest as seen from earth?

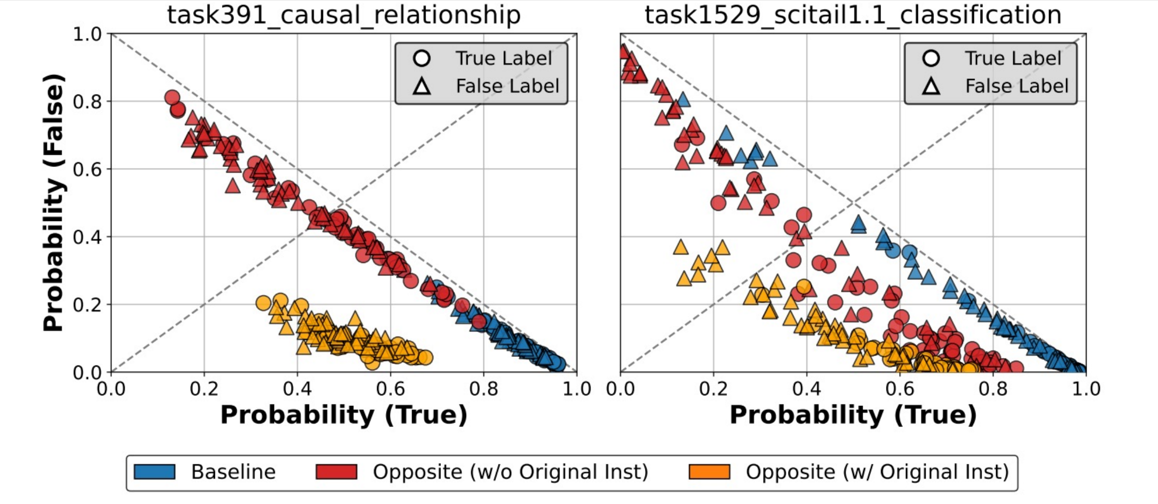
(C) [ID Response]: what star can be seen by the naked eye as the brightest seen from earth?

CONTENTS

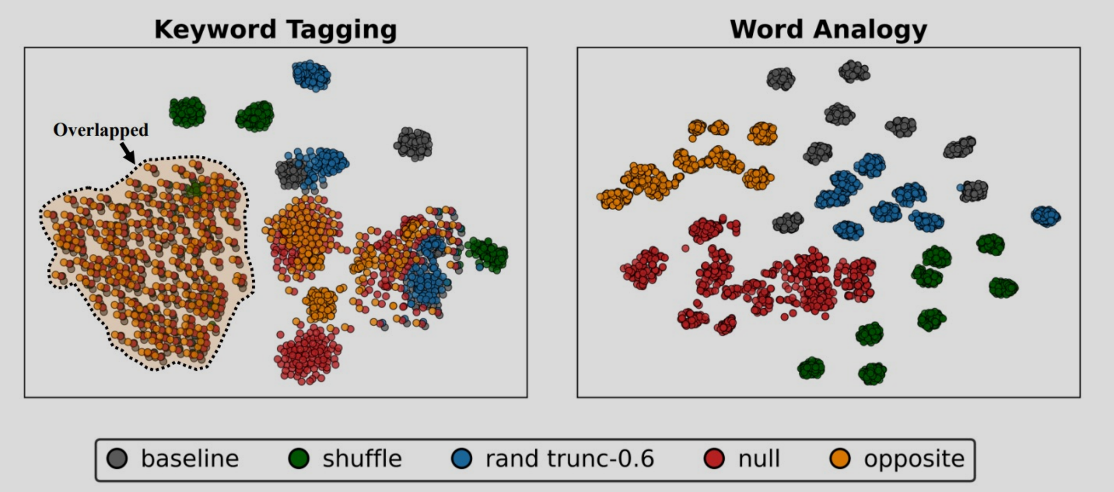
Likelihood Distribution Shifts

- ✓ ID shifts a set of outputs, which were settled on a single label.
- ✓ This not only expands the instruction-guided output space but also emphasizes the increased likelihood for alternative tokens.

Likelihood Shifts in Binary Classifications.



(a) Distribution shift in classification



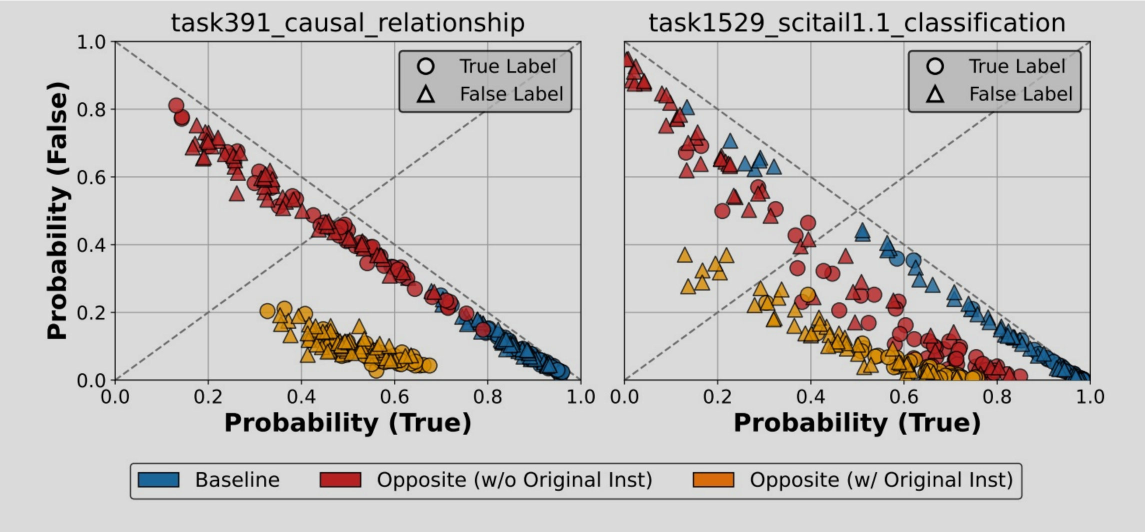
(b) Visualization of input embeddings

Visualization of Embeddings

Analysis

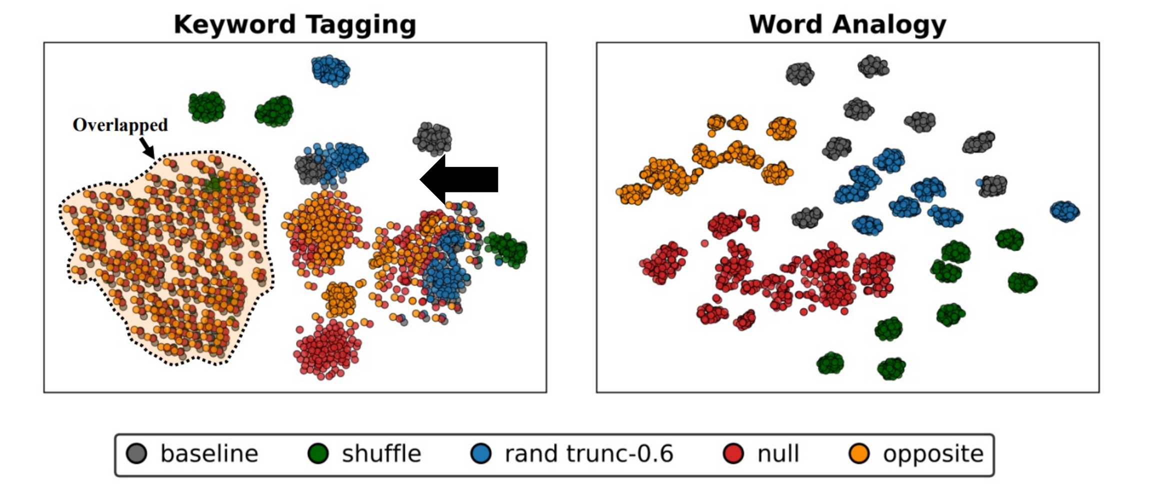
✓ We discovered that **the level of separation** affects the gain from our ID.

= The more accurately the model interprets the instructions, the greater gain from ID.



(a) Distribution shift in classification

T-SNE Embedding from the instructions.



(b) Visualization of input embeddings

Generalization Capabilities

- ✓ **(Cross-Evaluation)** ID is particularly advantageous when it encounters unseen datasets.
- ✓ **(Few-shots)** While the benefits are marginal, using ID still proves its benefits.
- ✓ **(MMLU)** ID works effectively even when prompts are not consists of ‘Instruction-Input’ pairs.

Cross-Evaluation

Dataset	UnNatInst	SupNatInst	
Model	T*k*-Large	T0-3B	Alpaca-7B
baseline	43.25	26.58	23.61
null	**44.57**	29.33	31.21
rand words	44.44	**29.49**	30.93
opposite	43.42	29.46	**31.38**

Few-shots Scenario

Model	T*k*-Large	T*k*-XL	Alpaca-7B
baseline	47.63	54.34	37.06
null	47.94	54.78	**38.75**
null (2 shots)	46.95	54.41	38.07
opposite	**48.08**	**54.80**	37.79
opposite (2 shots)	47.01	54.51	37.55

MMLU Benchmark

Method	Tk-Large	Tk-XL	OpenSNI-7B
Baseline	32.16	43.53	42.22
Opposite	**33.79**	**46.85**	43.17
Opposite$^-$	32.20	45.13	43.48
Opposite$^+$	31.83	43.88	43.25
Null	33.36	45.81	**43.69**
Null$^-$	33.07	45.16	42.73

CONTENTS

Conclusion

Takeaways

- **Instructive Decoding (ID)** is a novel decoding method designed to enhance instruction-following of LLMs, particularly on unseen task generalization.

- **Instruction-tuned LLMs can refine their responses** at *no extra training cost* by contrasting them with the responses from noisy instructions.

- **The gain of using ID differ** depending on the task, format, and model. We expect that *adaptive* application will bring more benefits.

- **We expect ID as a new breakthrough in prompt engineering.** By crafting Noisy Instructions, it's possible to significantly boost the ability of LLMs in diverse situations.

Thank You!