Instruction Learning with Large Language Models

Xubin Ren, Yiming Zhang

2023/09/29

Logistics

- Background and Motivation
- Paper presentation:
  - *Multitask Prompted Training Enables Zero-Shot Task Generalization*
  - *SELF-INSTRUCT: Aligning Language Models with Self-Generated Instructions*
  - *How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources*
  - *One Embedder, Any Task: Instruction-Finetuned Text Embeddings*
- Discussion

ChatGPT is a real generalist.

- brain storming
- explain complex concepts
- coding design
- … ….

Question: What makes the large language model so powerful?

Zhao, Wayne Xin, et al. "A survey of large language models." arXiv preprint arXiv:2303.18223 (2023).

What gives the robust capabilities of LLMs in handling a various of downstream tasks?

- Pretrained on an extensive corpus sourced from the vast expanses of the internet.

- fine-tunes models like GPT-3 and GPT-4 using human-labeled data

Pretraining
(GPT 3: 499 Bilion Tokens)

Fine-tuning
(InstructGPT*: 44K examples)

ten million times smaller

Ouyang, Long, et al. "Training language models to follow instructions with human feedback." Advances in Neural Information Processing Systems 35 (2022).

One model, only one task

<u>“Instance-level generalization within one task.”</u>

Sentiment Analysis Model:

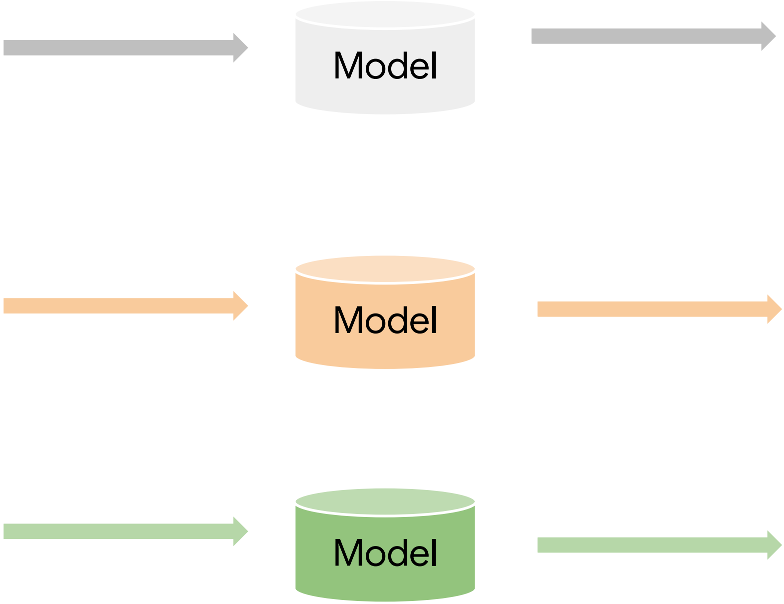
What a lovely day.

Question-Answering Model:

Can you recommend any song?

Machine Translation Mode:

You're so smart.



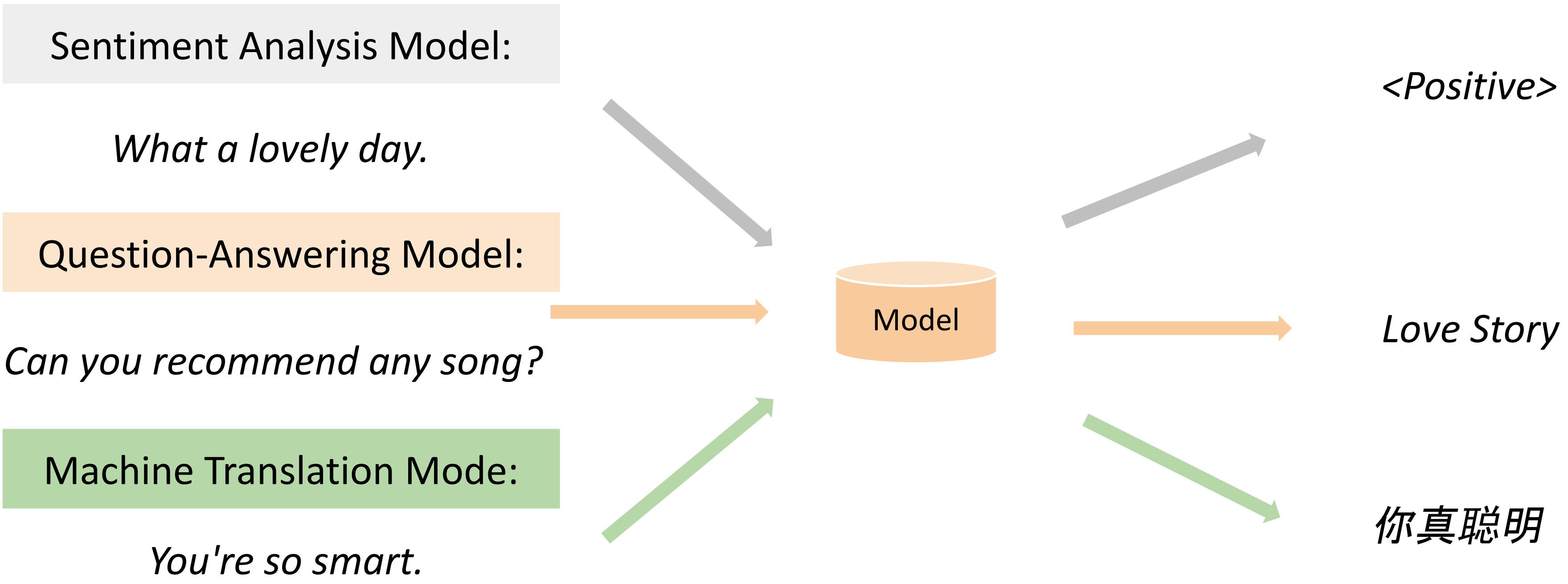
<Positive>

Love Story

你真聰明

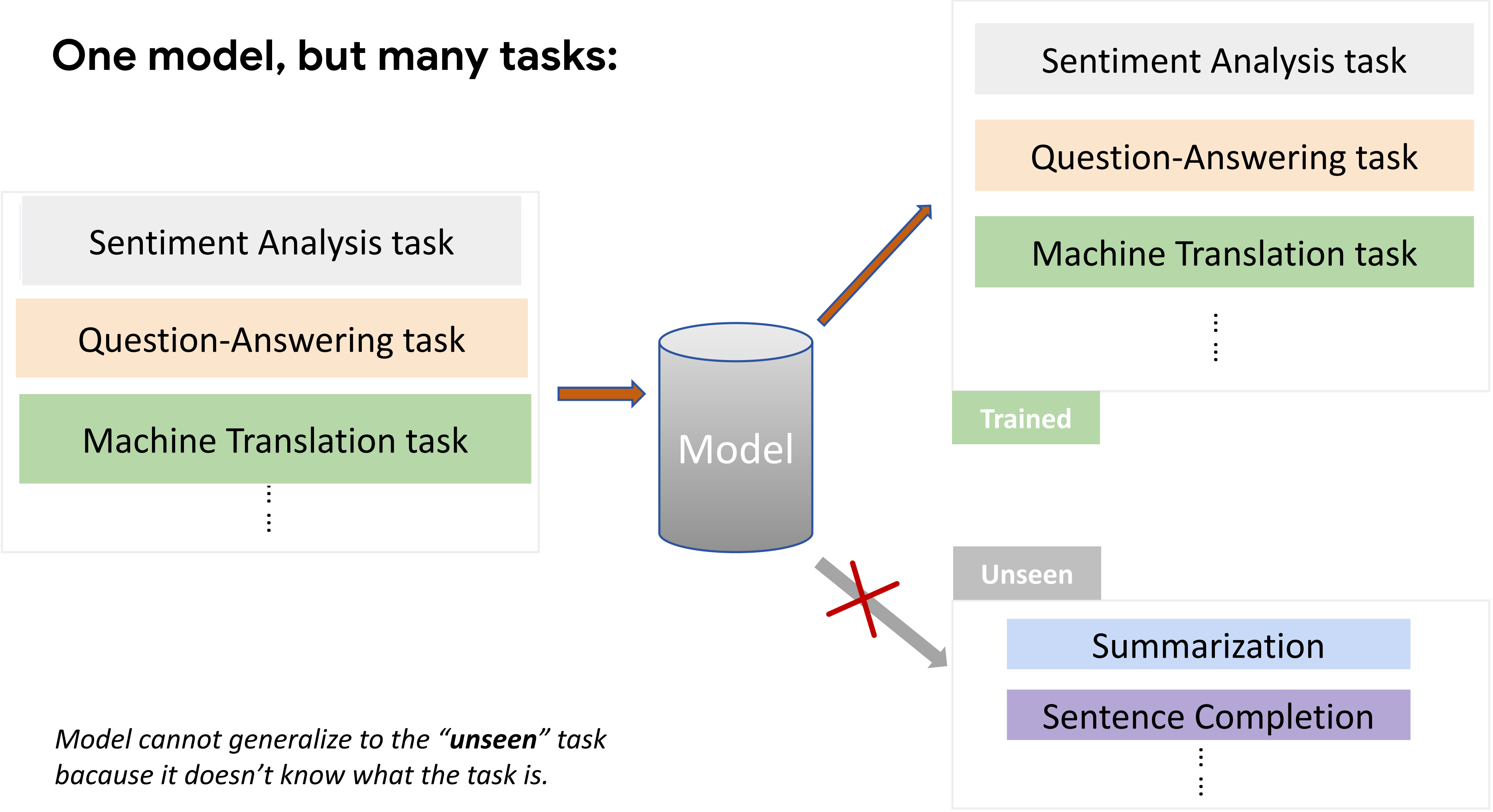
Otter, D. W., et al. "A survey of the usages of deep learning for natural language processing." IEEE transactions on neural networks and learning systems 32.2 (2020): 604-624.

One model, but many tasks:



McCann, Bryan, et al. "The natural language decathlon: Multitask learning as question answering." arXiv preprint arXiv:1806.08730 (2018).

One model, but many tasks:



*Model cannot generalize to the "**unseen**" task bacause it doesn't know what the task is.*

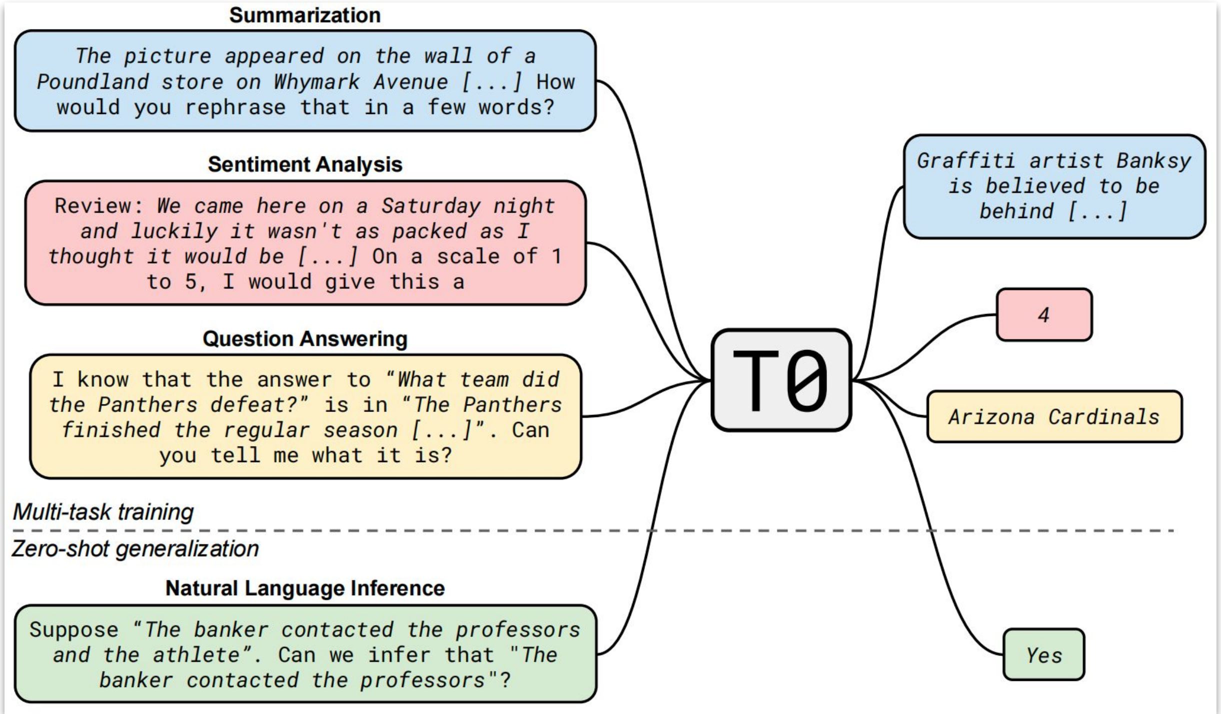
Paper list:

- Multitask Prompted Training Enables Zero-Shot Task Generalization
- SELF-INSTRUCT: Aligning Language Models with Self-Generated Instructions
- How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources
- One Embedder, Any Task: Instruction-Finetuned Text Embeddings

Paper 1: Multitask Prompted Training Enables Zero-Shot Task Generalization

T0, a model with zero-shot generalization ability.

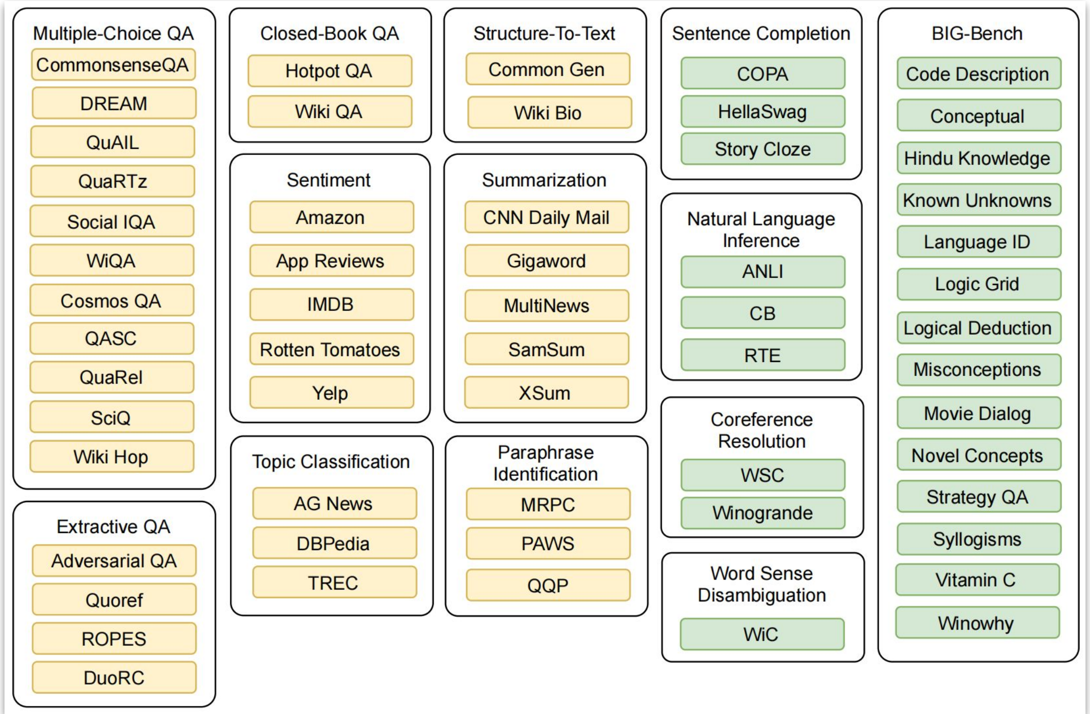
- An encoder-decoder based model.
- T0 could provide reasonable answer in unseen task, like natural language inference in this picture.



Sanh, Victor, et al. "Multitask prompted training enables zero-shot task generalization." arXiv preprint arXiv:2110.08207 (2021).

Experiment Setup

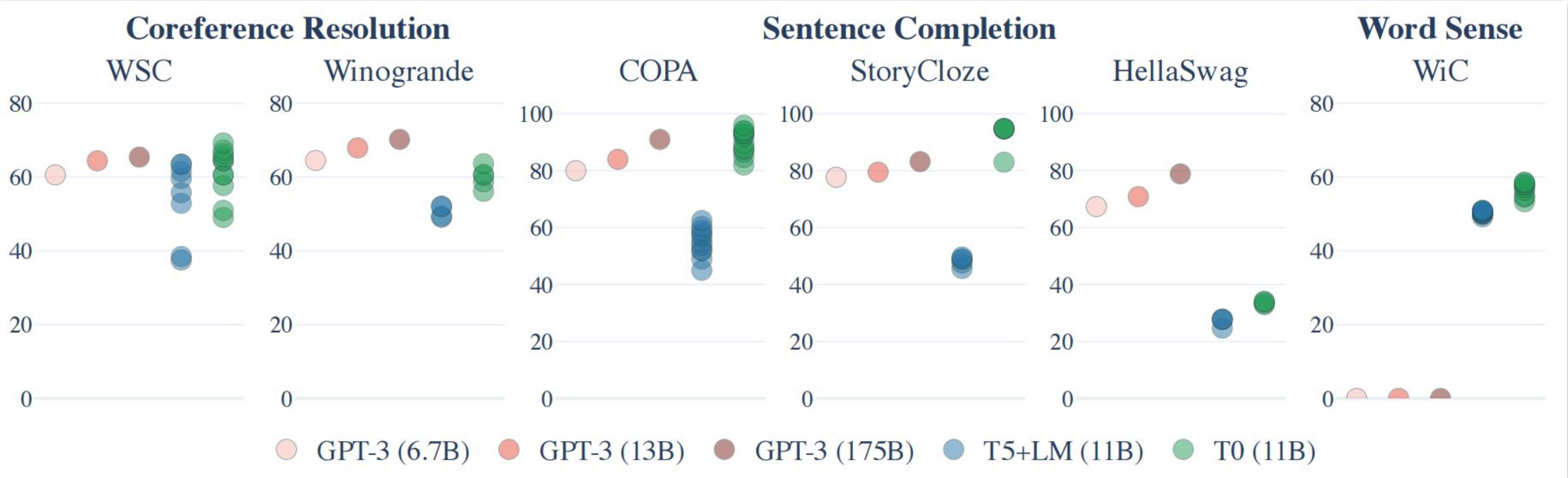
- T0 is trained on the **multitask** mixture detailed in Yellow tasks (8 tasks).
- Evaluate **zero-shot generalization** on the Green tasks



Sanh, Victor, et al. "Multitask prompted training enables zero-shot task generalization." arXiv preprint arXiv:2110.08207 (2021).

Compare T0 with GPT3 and T5+LM

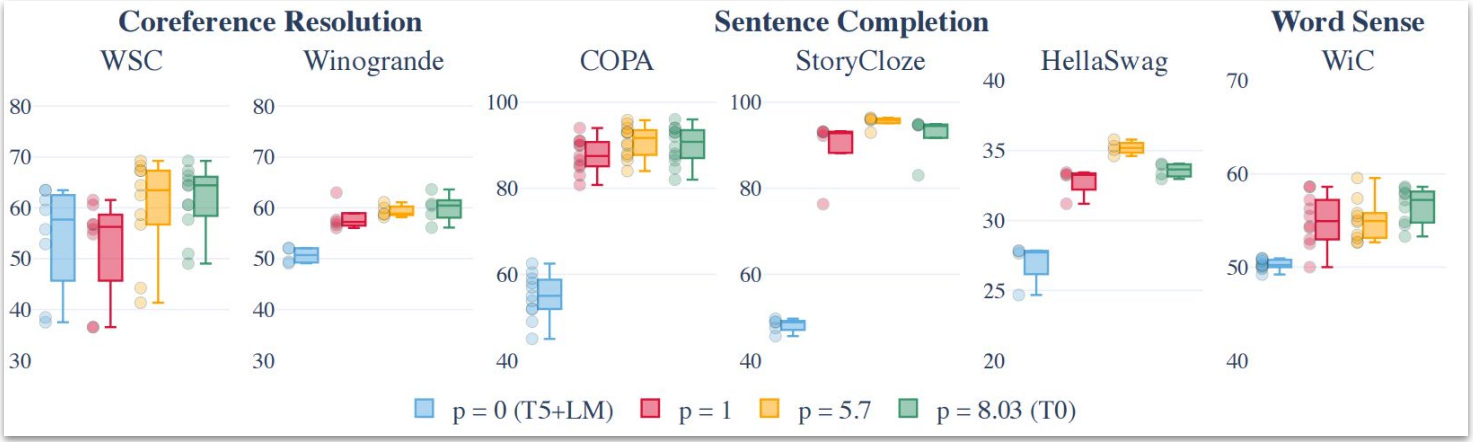
- T0 achieves a general good performance in **unseen tasks**



Sanh, Victor, et al. "Multitask prompted training enables zero-shot task generalization." arXiv preprint arXiv:2110.08207 (2021).

Prompt Robustness

- More prompts leads to better performance
- Generally lower interquartile range



Sanh, Victor, et al. "Multitask prompted training enables zero-shot task generalization." arXiv preprint arXiv:2110.08207 (2021).

Takeways

1. Instruction Learning could significantly **improve the performance of zero-shot task generalization**.

2. Adding **more number of prompts** will improve Zero-Shot generalization performance.

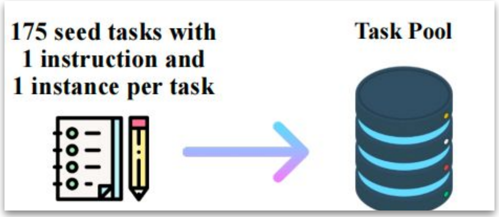
3. Thinking: In this era of prompting, perhaps more attention should be directed towards the efficient design of prompts, such as automating instruction discovery.

Sanh, Victor, et al. "Multitask prompted training enables zero-shot task generalization." arXiv preprint arXiv:2110.08207 (2021).

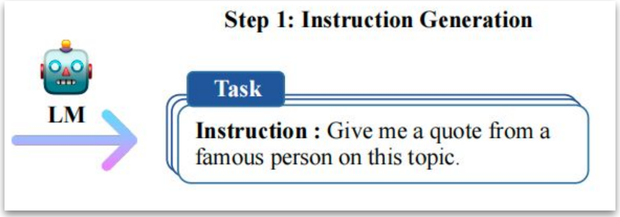
Paper 2. SELF-INSTRUCT: Aligning Language Models with Self-Generated Instructions

Pipeline of Instruction Generation

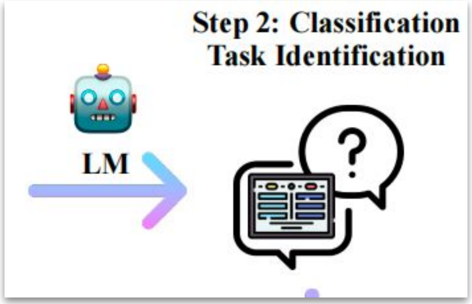
1. Manually design instructions. Put in the task pool.



2. Generate the instruction



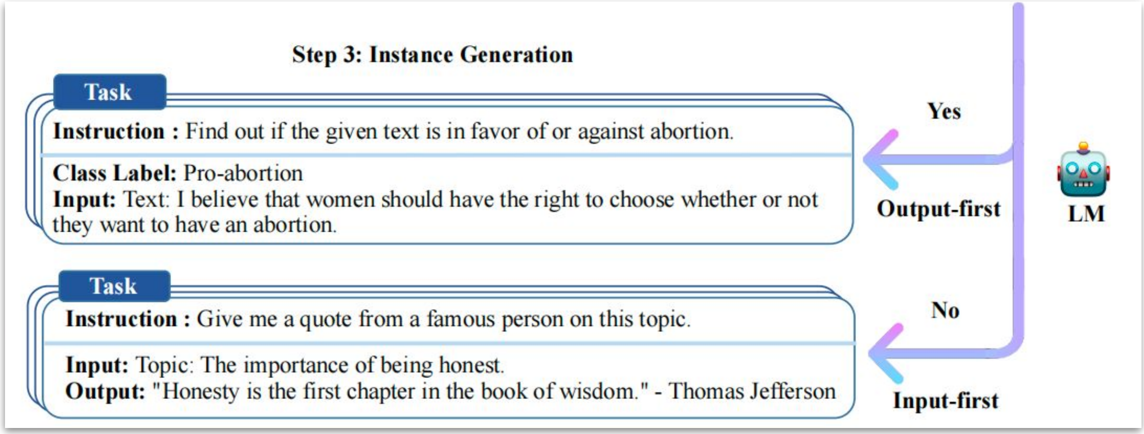
3. Manually design instructions. Put in the task pool.



Wang, Yizhong, et al. "Self-instruct: Aligning language model with self generated instructions." arXiv preprint arXiv:2212.10560 (2022).

Pipeline of Instruction Generation

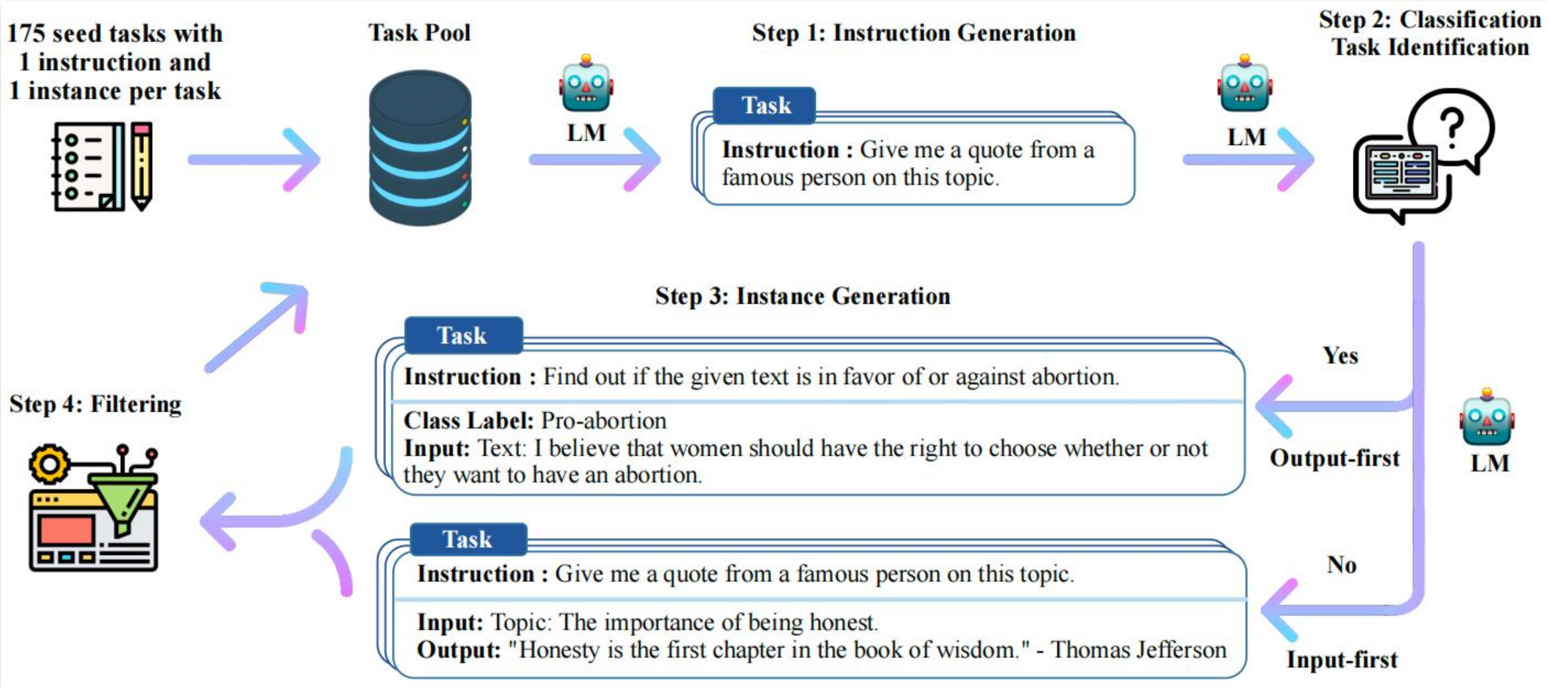
4. Instance Generation



5. Filtering

Wang, Yizhong, et al. "Self-instruct: Aligning language model with self generated instructions." arXiv preprint arXiv:2212.10560 (2022).

Pipeline of Instruction Generation



Wang, Yizhong, et al. "Self-instruct: Aligning language model with self generated instructions." arXiv preprint arXiv:2212.10560 (2022).

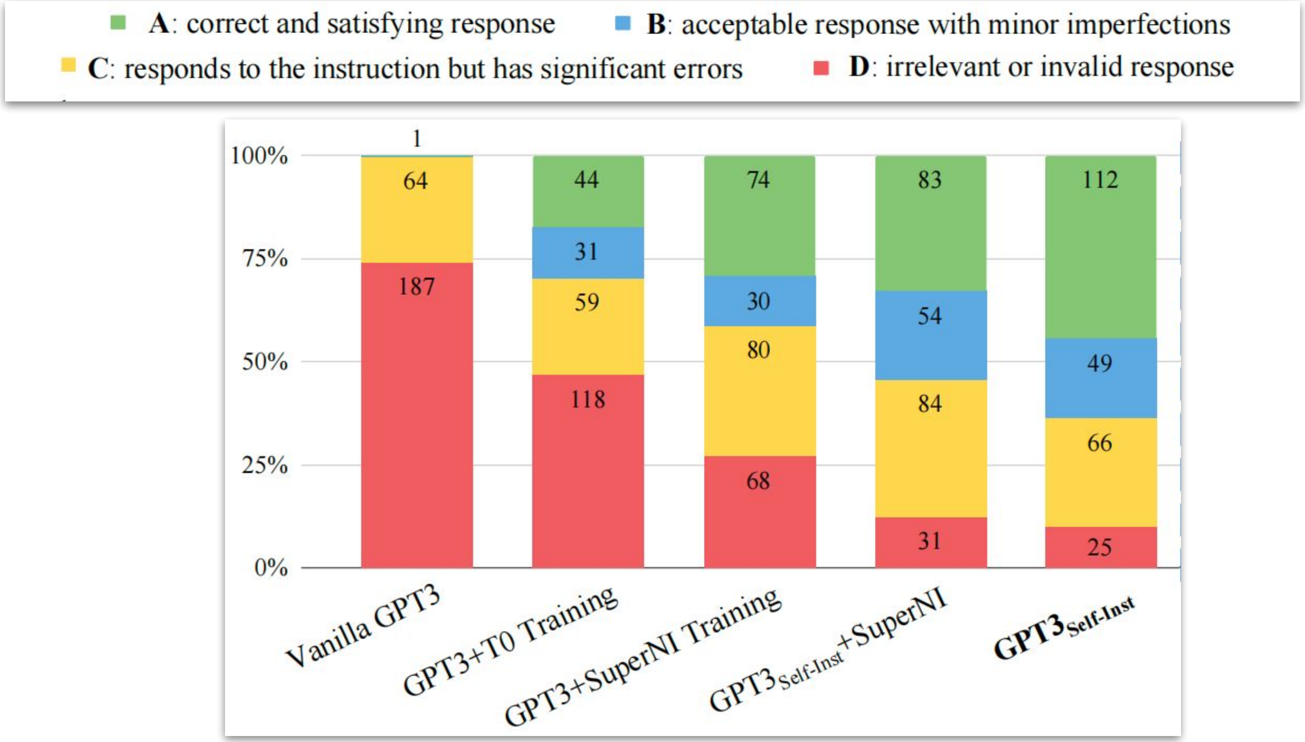
Evaluation results on unseen tasks from SUPERNI

1. Self-instruct **boosts** GPT3 by 33.1%
2. **Nearly matches** the performance of InstructGPT.
3. Complementary **improvement** to the existing human-labeled training set.

Model	# Params	ROUGE-L
Vanilla LMs		
T5-LM	11B	25.7
GPT3	175B	6.8
Instruction-tuned w/o SUPERNI		
T0	11B	33.1
GPT3 + T0 Training	175B	37.9
$\text{GPT3}_{\text{SELF-INST}}$ (Ours)	175B	39.9
InstructGPT_{001}	175B	**40.8**
Instruction-tuned w/ SUPERNI		
Tk-INSTRUCT	11B	46.0
GPT3 + SUPERNI Training	175B	49.5
$\text{GPT3}_{\text{SELF-INST}}$ + SUPERNI Training (Ours)	175B	**51.6**

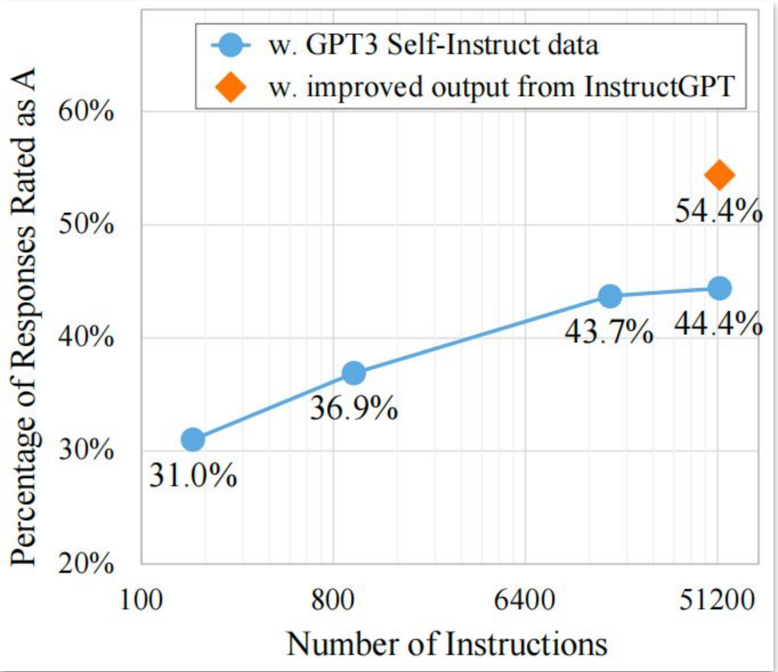
Wang, Yizhong, et al. "Self-instruct: Aligning language model with self generated instructions." arXiv preprint arXiv:2212.10560 (2022).

Evaluation based on Human Experts



Wang, Yizhong, et al. "Self-instruct: Aligning language model with self generated instructions." arXiv preprint arXiv:2212.10560 (2022).

Impact of the Quality of Instructions



Improving the **instruction quality** (using instructGPT) can significantly boost the performance

Wang, Yizhong, et al. "Self-instruct: Aligning language model with self generated instructions." arXiv preprint arXiv:2212.10560 (2022).

Takeways

- You **don't need a huge amount of labeled data** to get good initial instructionfollowing ability.
- LLMs themselves know many tasks/skills.
- One aspect of instruction tuning is to elicit these skills (i.e., alignment).
- Self-instruct is applicable to a broad range of LLMs, both vanilla and instruction-tuned.
- More broadly, humans & AI should collaborate in building datasets.

Wang, Yizhong, et al. "Self-instruct: Aligning language model with self generated instructions." arXiv preprint arXiv:2212.10560 (2022).

Paper 3. How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources

Various Instruction Dataset

Table 1: Instruction datasets investigated in this work. CoT and FLAN V2 are sampled to 100K to match the sizes of other datasets. We report the average number of rounds ($\bar{N}_{rounds}$), average length of prompts ($\bar{L}_{prompt}$), average length of completion ($\bar{L}_{completion}$).

Datasets	**Sourced from**	**# Instances**	$\bar{N}_{rounds}$	$\bar{L}_{prompt}$	$\bar{L}_{completion}$
SuperNI [42]	NLP datasets + Human-written Instructions	96,913	1.0	291.1	38.7
CoT [44]	NLP datasets + Human-written CoTs	100,000	1.0	266.0	53.2
Flan V2 [26]	NLP datasets + Human-written Instructions	100,000	1.0	355.7	31.2
Dolly [12]	Human-written from scratch	15,011	1.0	118.1	91.3
Open Assistant 1 [23]	Human-written from scratch	34,795	1.6	34.8	212.5
Self-instruct [41]	Generated w/ vanilla GPT3 LM	82,439	1.0	41.5	29.3
Unnatural Instructions [21]	Generated w/ Davinci-002	68,478	1.0	107.8	23.6
Alpaca [38]	Generated w/ Davinci-003	52,002	1.0	27.8	64.6
Code-Alpaca [6]	Generated w/ Davinci-003	20,022	1.0	35.6	67.8
GPT4-Alpaca [31]	Generated w/ Davinci-003 + GPT4	52,002	1.0	28.0	161.8
Baize [46]	Generated w/ ChatGPT	210,311	3.1	17.6	52.8
ShareGPT[3]	User prompts + outputs from various models	168,864	3.2	71.0	357.8

Wang, Yizhong, et al. "How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources." arXiv preprint arXiv:2306.04751 (2023).

Various Instruction Dataset

An example, ShareGPT

- Collect from **users' interactions with GPT**
- Users' input prompts **contain instructions**
- A **fusion** of human-written data and GPT-generated data

<|user|>
Explain the fault-tolerance of the reaction control system on the Space Shuttle.

<|assistant|>
The reaction control system (RCS) on the Space Shuttle was designed to be fault-tolerant, meaning it was able to continue functioning even if one or more of its components failed. The RCS consisted of two sets of ... </s>

<|user|>
Did the RCS have any on-orbit failures?

<|assistant|>
There were several instances where the reaction control system (RCS) on the Space Shuttle experienced failures or malfunctions during on-orbit missions. These ... </s>

Wang, Yizhong, et al. "How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources." arXiv preprint arXiv:2306.04751 (2023).

Dataset Selection and Performance Analysis

Should we combine all the datasets or use a specific dataset that serves as a comprehensive source?

- **Not a single best** instruction tuning dataset across all tasks
- Achieve the best overall (average) performance when **combining all the data**

	MMLU (factuality)	GSM (reasoning)	BBH (reasoning)	TydiQA (multilinguality)	Codex-Eval (coding)	AlpacaFarm (open-ended)	Average
	EM (0-shot)	EM (8-shot, CoT)	EM (3-shot, CoT)	F1 (1-shot, GP)	P@10 (0-shot)	Win % vs Davinci-003	
Vanilla LLaMa 13B	42.5	14.0	36.9	47.4	26.6	-	-
+SuperNI	49.8	4.0	2.8	51.4	13.1	5.0	21.0
+CoT	44.5	**39.5**	39.0	**52.2**	23.3	4.7	33.9
+Flan V2	**50.7**	21.0	39.2	47.5	16.2	5.3	30.0
+Dolly	45.3	17.0	26.0	46.8	31.4	18.3	30.8
+Open Assistant 1	43.1	16.0	38.5	38.3	31.8	55.2	37.1
+Self-instruct	30.3	9.0	29.6	40.4	13.4	7.3	21.7
+Unnatural Instructions	46.2	7.5	32.8	39.3	24.8	10.8	26.9
+Alpaca	45.1	8.0	34.5	32.8	27.6	33.2	30.2
+Code-Alpaca	42.6	12.0	36.6	41.3	34.5	21.3	31.4
+GPT4-Alpaca	47.0	14.0	38.3	24.4	32.5	63.6	36.6
+Baize	43.5	8.5	36.7	33.9	27.3	33.9	30.6
+ShareGPT	49.2	16.0	40.1	30.1	31.6	**69.1**	39.3
+ Human data mix	50.4	36.5	39.4	49.8	23.7	38.5	39.7
+Human+GPT data mix	49.2	36.5	**42.8**	46.1	**35.0**	57.2	**44.5**

Wang, Yizhong, et al. "How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources." arXiv preprint arXiv:2306.04751 (2023).

Impact of the Base Model

The better the quality of the base model, **the better performance** it will achieve.

Table 4: Performance of different base models after training on the Human+GPT data mixture.

	MMLU (factuality)	GSM (reasoning)	BBH (reasoning)	TydiQA (multilinguality)	Codex-Eval (coding)	AlpacaFarm (open-ended)	Average
	EM (0-shot)	EM (8-shot, CoT)	EM (3-shot, CoT)	F1 (1-shot, GP)	P@10 (0-shot)	Win % vs Davinci-003	
Pythia 6.9B	34.6	15.5	27.8	33.4	21.4	9.3	23.7
OPT 6.7B	34.9	15.5	27.9	27.2	7.9	14.5	21.3
LLaMA7B	**44.5**	**27.0**	**39.2**	**45.7**	**27.8**	**48.6**	**38.8**

Wang, Yizhong, et al. "How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources." arXiv preprint arXiv:2306.04751 (2023).

Put it Together!

All the Insturction data + open-source **state-of-the-art** LLMs

	MMLU (factuality)	GSM (reasoning)	BBH (reasoning)	TydiQA (multilinguality)	Codex-Eval (coding)	AlpacaFarm (open-ended)	Average
	EM (0-shot)	EM (8-shot, CoT)	EM (3-shot, CoT)	F1 (1-shot, GP)	P@10 (0-shot)	Win % vs Davinci-003	
Vanilla LLaMa models ↓							
LLaMa 7B	31.0	9.0	33.3	39.1	18.3	-	-
LLaMa 13B	42.5	14.0	36.9	47.4	26.6	-	-
LLaMa 30B	54.1	33.5	48.5	57.5	42.9	-	-
LLaMa 65B	58.7	50.5	57.1	57.4	42.9	-	-
65B models trained on alternate data mixtures ↓							
ShareGPT 65B	61.5 (+2.8)	42.0 (-8.5)	52.1 (-5.0)	33.5 (-23.9)	53.5 (+10.6)	72.8	52.6
Human mix. 65B	60.7 (+2.0)	57.5 (+7.0)	52.7 (-4.4)	58.5 (+1.1)	43.2 (+0.3)	47.4	53.3
🐫 models trained on our final Human+GPT data mixture ↓							
TÜLU 🐫 7B	44.5 (+13.5)	27.0 (+18.0)	39.2 (+5.9)	45.7 (+6.6)	27.8 (+9.5)	48.3	38.8
TÜLU 🐫 13B	49.2 (+6.7)	36.5 (+22.5)	42.8 (+5.9)	46.1 (-1.3)	35.0 (+8.4)	53.9	44.5
TÜLU 🐫 30B	57.7 (+3.6)	51.0 (+17.5)	48.7 (+0.2)	58.2 (+0.7)	46.0 (+3.1)	63.5	54.1
TÜLU 🐫 65B	59.2 (+0.5)	60.0 (+9.5)	53.5 (-3.6)	51.8 (-5.6)	45.9 (+3.0)	62.7	55.7
Proprietary models ↓							
ChatGPT	67.9	76.0	66.1	51.9	88.4	84.4	72.2
GPT-4	82.4	92.5	88.0	70.8	94.1	91.6	86.8

Wang, Yizhong, et al. "How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources." arXiv preprint arXiv:2306.04751 (2023).

Other Evaluation Metrics

Are there other evaluation metrics? (e.g, <u>model-based evaluation</u> or human-based evaluation)

Model-based Evaluation

- Utilize a LLMs like GPT-4 to **determine which is better**
- Tuning with **ShareGPT** achieves the best results
- The model prefer the **long and diverse** answers

Training Dataset ↓	7B	13B	30B	65B
SuperNI	5.7	6.2	-	-
CoT	4.2	5.6	-	-
Flan V2	4.6	5.5	-	-
Dolly	12.7	16.2	-	-
Open Assistant 1	47.8	53.5	-	-
Self-instruct (original)	7.5	6.8	-	-
Unnatural Instructions	8.2	10.9	-	-
Alpaca	21.1	28.7	-	-
Code-Alpaca	17.5	19.4	-	-
GPT4-Alpaca	57.0	61.1	-	-
Baize	23.5	28.7	-	-
ShareGPT	**58.3**	**68.9**	**70.2**	**72.8**
Human mix.	29.4	36.3	44.6	46.5
TÜLU 🐫	48.3	53.9	63.5	62.7

Table 6: Win-rate (%) of LLAMA models of varying sizes fine-tuned on the given dataset against Davinci-003 using AlpacaFarm [16].

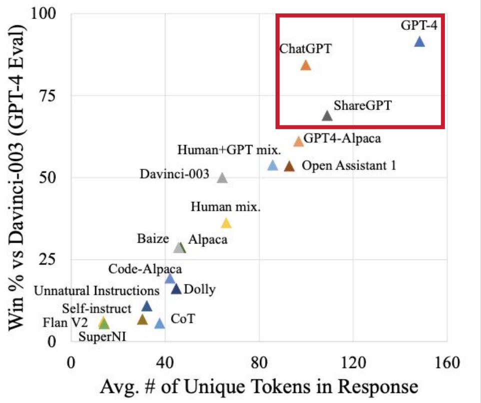


Figure 2: Win-rate scores of 13B models (trained on different datasets) given by GPT-4 strongly correlate with the average numbers of unique tokens in the model responses (Pearson $r = 0.96$).

Wang, Yizhong, et al. "How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources." arXiv preprint arXiv:2306.04751 (2023).

Other Evaluation Metrics

Are there other evaluation metrics? (e.g, model-based evaluation or human-based evaluation)

Human-based Evaluation

- **Consistent** with prior findings
- **Larger** models show greater improvements

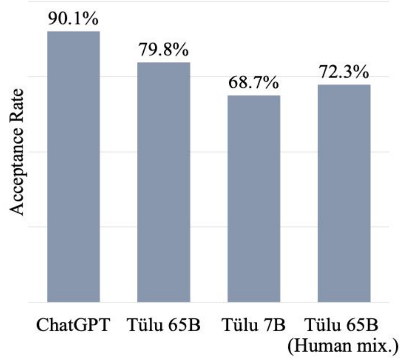


Figure 3: Human acceptance rates for four evaluated models.

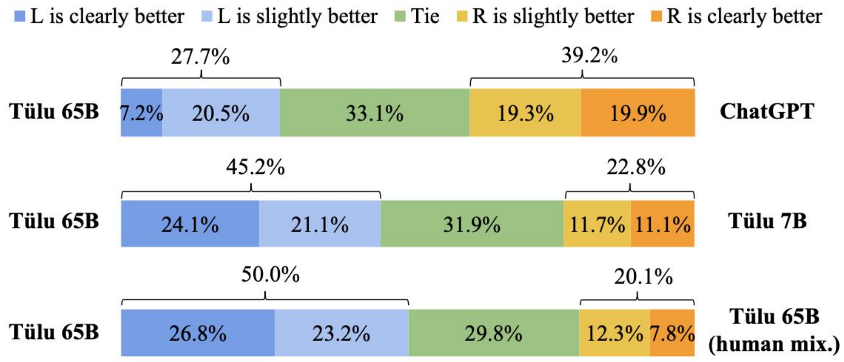


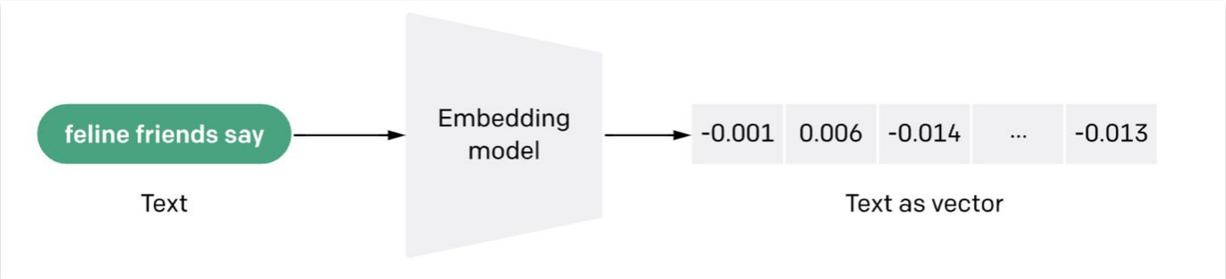
Figure 4: Human preference rates for three comparison pairs of models.

Wang, Yizhong, et al. "How Far Can Camels Go? Exploring the State of Instruction Tuning on Open Resources." arXiv preprint arXiv:2306.04751 (2023).

Paper 4. One Embedder, Any Task: Instruction-Finetuned Text Embeddings

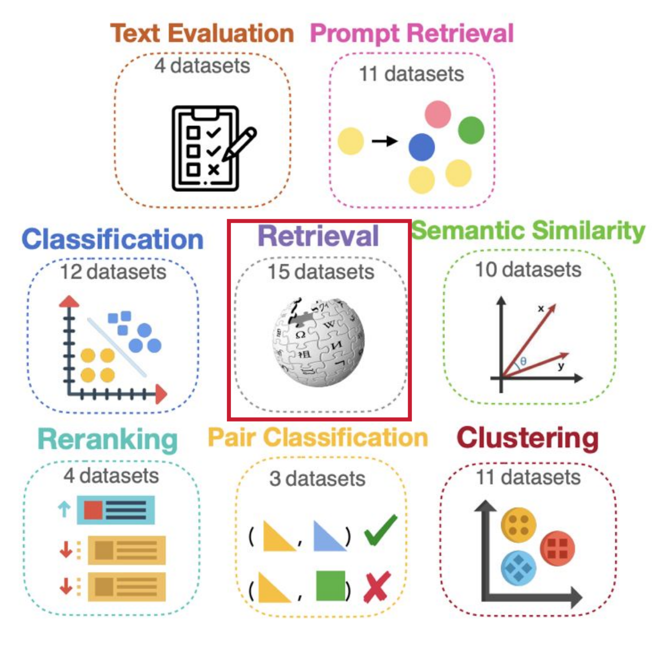
What is Text Embedding

Text embeddings are **numerical representations of concepts**



Su, Hongjin, et al. "One embedder, any task: Instruction-finetuned text embeddings." ACL 2023.

Applications of Text Embedding



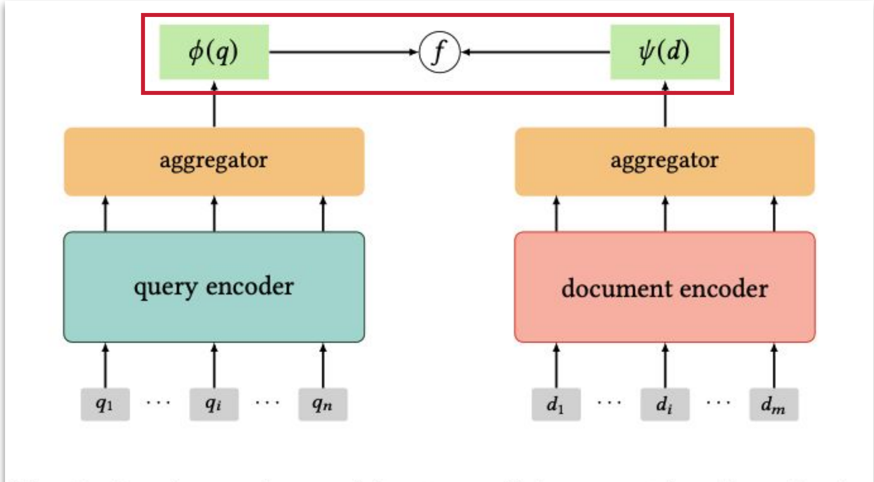
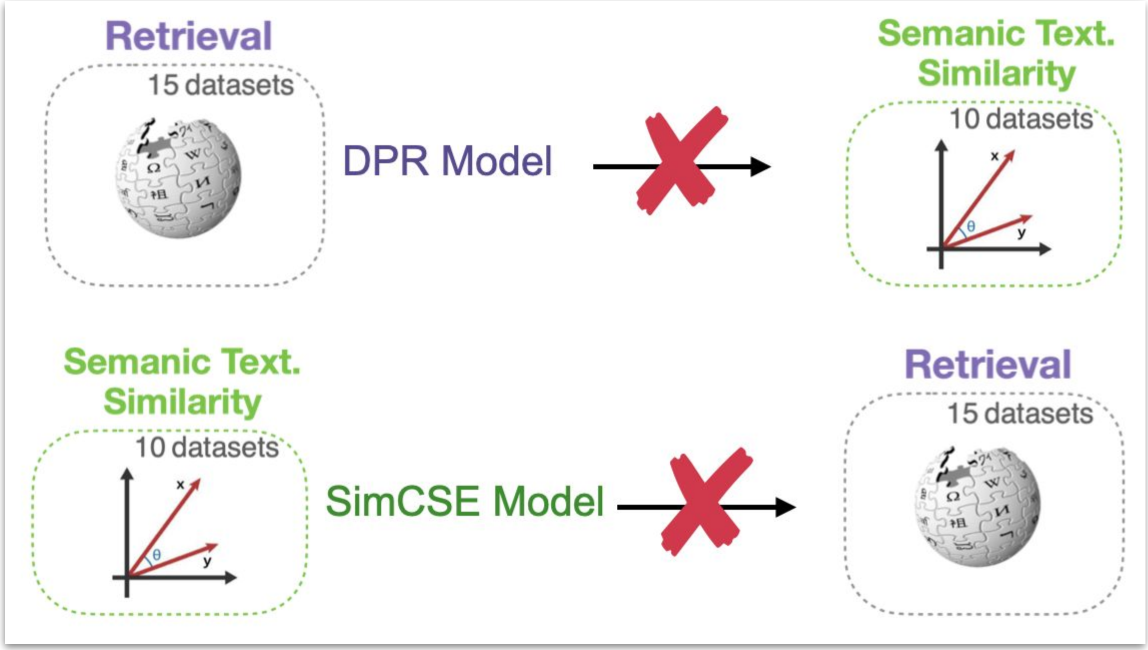


Fig. 2. Dual-encoder architecture of dense retrieval methods.

Su, Hongjin, et al. "One embedder, any task: Instruction-finetuned text embeddings." ACL 2023.
Guo, Jiafeng, et al. "Semantic models for the first-stage retrieval: A comprehensive review." *ACM Transactions on Information Systems (TOIS)* 40.4 (2022): 1-42.

Model Transferability Challenges

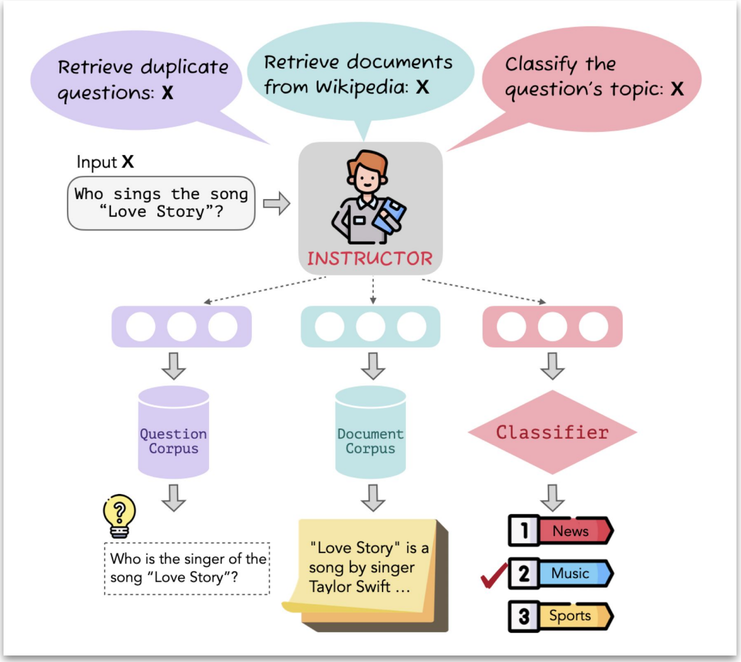
One Embedder trained with specific task can’t be transferred to a new task



Su, Hongjin, et al. "One embedder, any task: Instruction-finetuned text embeddings." ACL 2023.

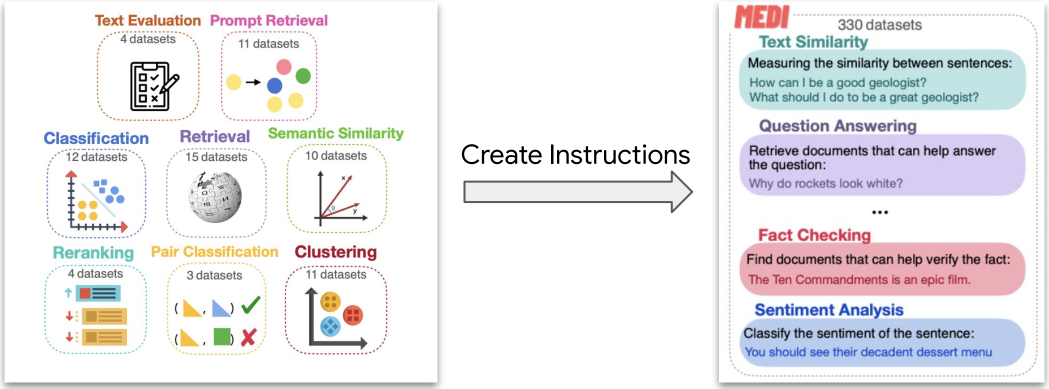
Apply Instruction in Text Embedding!

Text embeddings adjusted to different downstream applications using **task and domain descriptions**



Su, Hongjin, et al. "One embedder, any task: Instruction-finetuned text embeddings." ACL 2023.

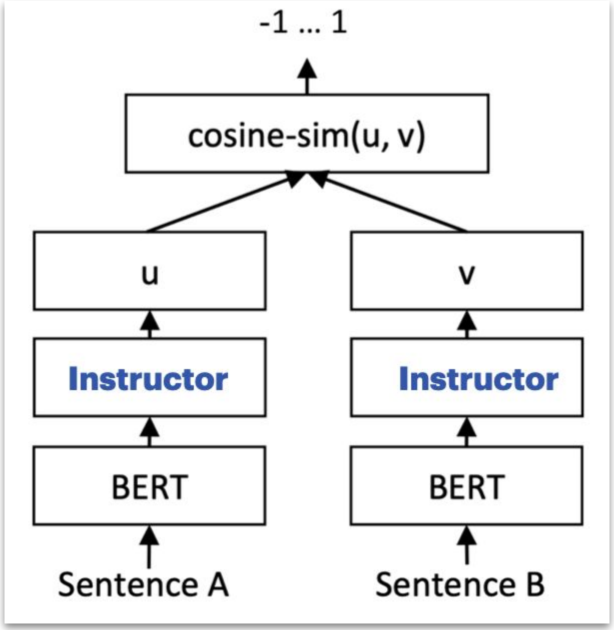
MEDI: Multitask Embedding Data with Instructions



Task type	# of Datasets	Task	Instruction
Retrieval	15	Natural Question (BEIR)	***Query instruction:*** Represent the Wikipedia question for retrieving supporting documents:, ***Doc instruction:*** Represent the Wikipedia document for retrieval:
Reranking	4	MindSmallReranking	***Query instruction:*** Represent the News query for retrieving articles: ***Doc instruction:*** Represent the News article for retrieval:
Clustering	11	MedrxivClusteringS2S	Represent the Medicine statement for retrieval:

Su, Hongjin, et al. "One embedder, any task: Instruction-finetuned text embeddings." ACL 2023.

Architecture and Training Method



1. Create Pos/Neg Pairs

$$(x, I_x, y, I_y)$$

2. Calculate Similarity Score

$$s(x, y) = \cos(\mathbf{E}_I(I_x \oplus x), \mathbf{E}_I(I_y \oplus y))$$

3. Contrastive Training

$$\mathcal{L} = \frac{e^{s(x,y^+)/\gamma}}{\sum_{y \in \mathcal{B}} e^{s(x,y)/\gamma}},$$

Su, Hongjin, et al. "One embedder, any task: Instruction-finetuned text embeddings." ACL 2023.

Overall Comparison

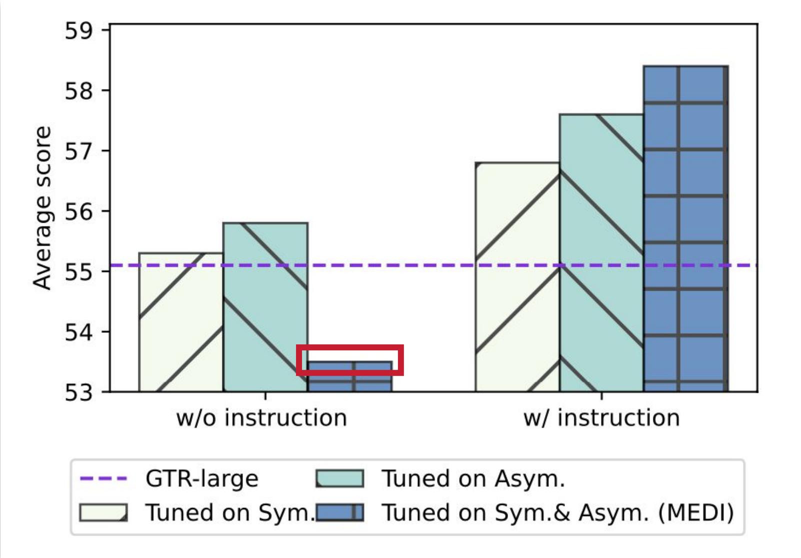
Achieving SOTA performance on various benchmarks **with Instruction**

Benchmark	MTEB								Billboard	Prompt	Avg.
Task category	Retri.	Rerank	Cluster	Pair.	Class.	STS	Sum.	Avg.	Text Eval.	Retri.	
# datasets	15	4	11	3	12	10	1	56	3	11	70
Small Models for reference (<500M)											
SimCSE (110M)	21.9	47.5	33.4	73.7	67.3	79.1	23.3	48.7	29.4	58.3	48.2
coCondenser (110M)	33.0	51.8	37.6	81.7	64.7	76.5	29.5	52.4	31.5	59.6	51.8
Contriever (110M)	41.9	53.1	41.1	82.5	66.7	76.5	30.4	56.0	29.0	57.3	53.2
GTR-Large (335M)	47.4	55.4	41.6	85.3	67.1	78.2	29.5	58.3	31.2	59.8	55.1
INSTRUCTOR (335M)	**47.6**	**57.5**	**45.3**	**85.9**	**73.9**	**83.2**	**31.8**	**61.6**	**36.9**	**63.2**	**58.4**
Relative gain (%)	+0.4	+4.5	+8.9	+0.7	+10.1	+6.4	+7.8	+5.7	+18.3	+5.7	+5.9
Large Models for reference(≥500M)											
Sent-T5-XXL (4.8B)	42.2	56.4	43.7	85.1	**73.4**	82.6	30.1	59.5	33.9	61.5	56.5
GTR-XXL (4.8B)	48.1	56.7	42.4	86.1	67.4	78.4	30.6	58.9	32.0	60.8	55.8
SGPT-NLI (5.8B)	32.3	52.3	37.0	77.0	70.1	80.5	30.4	53.7	29.6	57.9	51.9
GTR-XL (1.5B)	48.0	56.0	41.5	86.1	67.1	77.8	30.2	58.4	32.0	60.4	55.5
INSTRUCTOR-XL (1.5B)	**49.3**	**57.3**	**44.7**	**86.6**	73.2	**83.1**	**32.0**	**61.8**	**34.1**	**68.6**	**58.8**
Relative gain (%)	+2.7	+2.3	+7.7	+0.6	+9.1	+6.9	+6.0	+5.8	+6.6	+13.6	+5.9

Su, Hongjin, et al. "One embedder, any task: Instruction-finetuned text embeddings." ACL 2023.

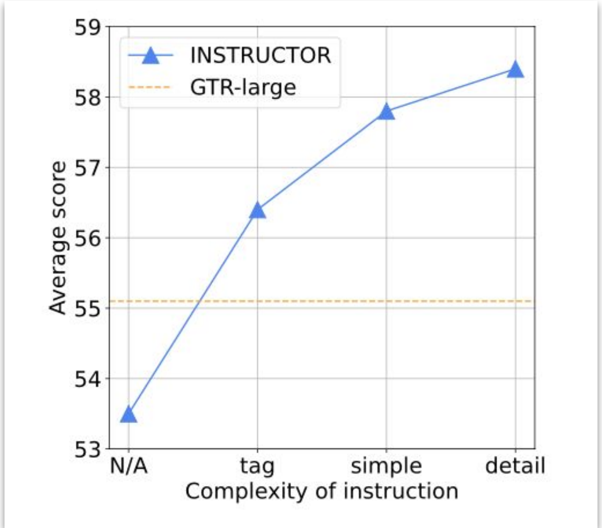
The Necessity of Instructions

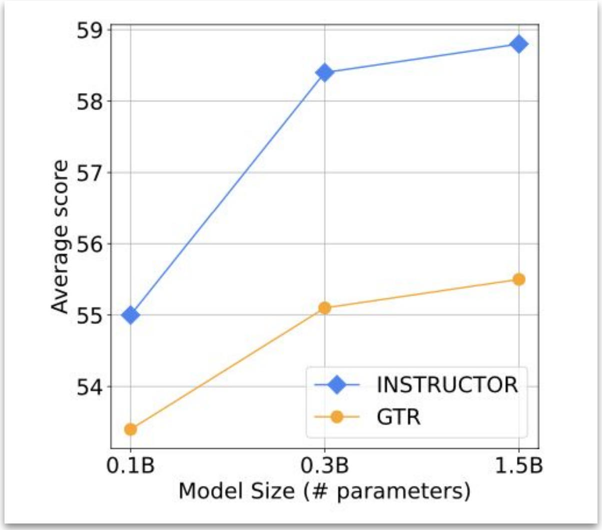
Without instructions, the model **cannot be trained with the whole dataset**.



Su, Hongjin, et al. "One embedder, any task: Instruction-finetuned text embeddings." ACL 2023.

Complexity of Instuctions & Model Size





Su, Hongjin, et al. "One embedder, any task: Instruction-finetuned text embeddings." ACL 2023.

Questions to Discuss

- Can we directly provide instructions to LLMs instead of tuning them?
- Have tasks outside of NLP utilized instruction tuning with LLMs to address problems?
- Are there other forms or approaches that can incorporate the idea of instruction tuning, similar to Instructor?

Thank You.