Multi-Task Instruction Tuning

John Zoscak
Department of Computer Science, University of Virginia Charlottesville, VA
jmz9sad@virginia.edu

Wendy Zheng
Department of Computer Science, University of Virginia Charlottesville, VA
ncd9cf@virginia.edu

Yinhan He
Department of Electrical and Computer Engineering, University of Virginia Charlottesville, VA
nee7ne@virginia.edu

Introduction

- Optimization of task understanding for NLP models by feeding the models natural language
- Creating data input in definite (templated) forms which help the models understand natural language prompts / instructions
- Instruction and Prompt engineering

Overview

- *Finetuned Language Models are Zero-Shot Learners*
- *Multitask Prompted Training Enables Zero Shot Task Generalization*
- *Cross-Task Generalization via Natural Language Crowdsourcing Instructions*
- *Super-Natural Instructions: Generalization via Declarative Instructions on +1600 NLP Tasks*

Finetuned language models are zero-shot learners

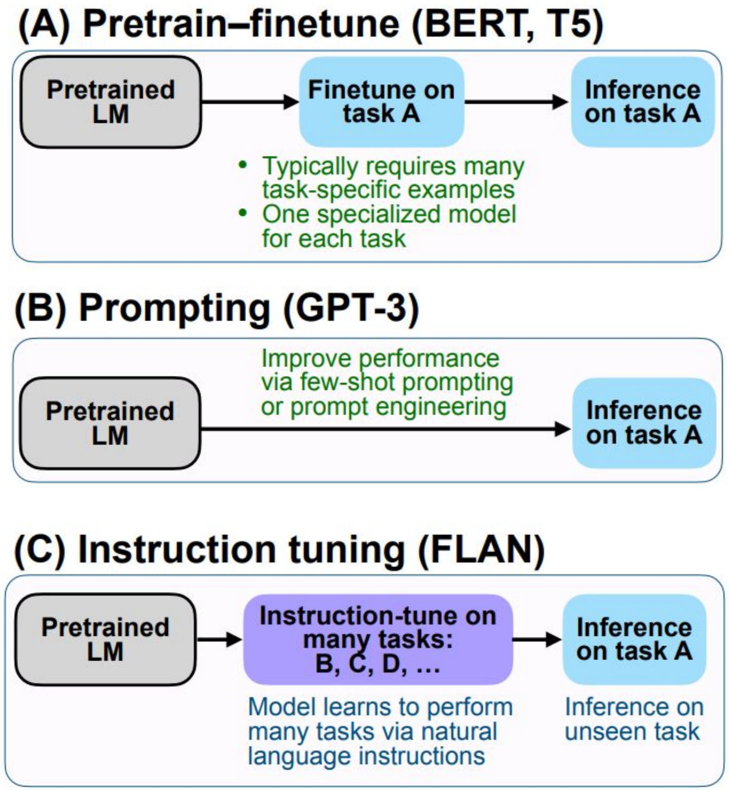
Jason Wei∗ , Maarten Bosma∗ , Vincent Y. Zhao∗ ,
Kelvin Guu∗ , Adams Wei Yu, Brian Lester, Nan Du,
Andrew M. Dai, and Quoc V. Le

Motivation

- **Observation:** Models like GPT-3 do not perform well on zero-shot learning compared to one-shot/few-shot learning.
  - Seen in tasks such as reading comprehension, question answering, and natural language inference.
- **Hypothesis:** Possible reasoning: Without few-shot exemplars, it is harder for models to perform well on prompts that are not similar to the format of the pretraining data.
- **Goal:** Test multi-task learning with instructions as a method for doing zero-shot learning.

Instruction Tuning

- Is a form of fine-tuning using instructions as opposed to task methods.
- Reformatting prompts from datasets into natural-language instructions.
- The instruction tuned decoder-only model is named FLAN (Finetuned Language Net).

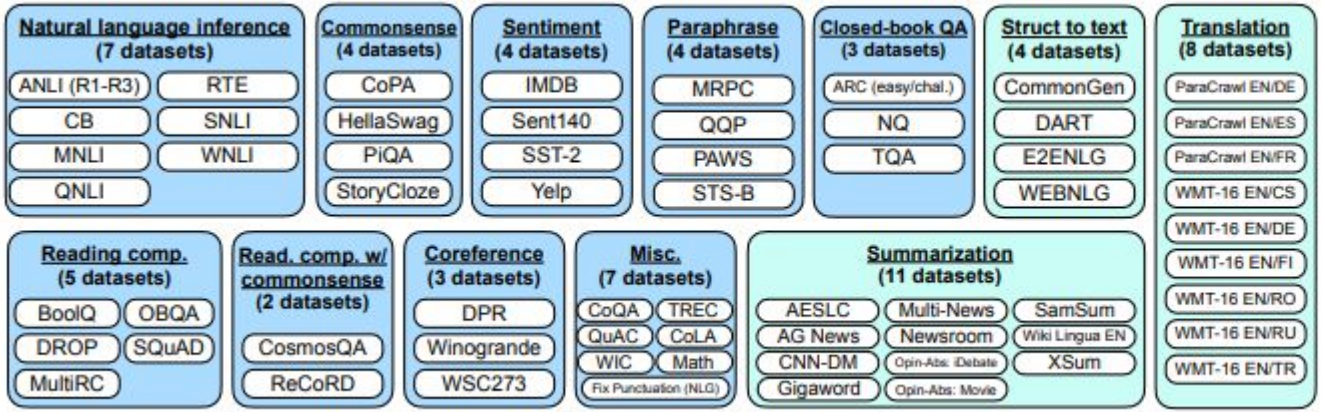

(A) Pretrain–finetune (BERT, T5)
Pretrained LM
Finetune on task A
Inference on task A
• Typically requires many task-specific examples
• One specialized model for each task
(B) Prompting (GPT-3)
Pretrained LM
Improve performance via few-shot prompting or prompt engineering
Inference on task A
(C) Instruction tuning (FLAN)
Pretrained LM
Instruction-tune on many tasks: B, C, D, …
Inference on task A
Model learns to perform many tasks via natural language instructions
Inference on unseen task

Reformatting data into instruction-form

- Taking data from an existing dataset and converting it into an instruction by following an instruction template.
- Instructions take the form of natural language.
  - Combines appealing aspects of both the pretrain–finetune and prompting paradigms by using supervision via finetuning to improve language model's responses to inference-time text interactions.
  - May improve model response to pure natural language prompts

Datasets

- Each dataset is categorized into a task cluster.
- NLU tasks are in blue, NLG tasks are in teal.



Natural language inference (7 datasets)
ANLI (R1-R3), RTE, CB, SNLI, MNLI, WNLI, QNLI

Commonsense (4 datasets)
CoPA, HellaSwag, PiQA, StoryCloze

Sentiment (4 datasets)
IMDB, Sent140, SST-2, Yelp

Paraphrase (4 datasets)
MRPC, QQP, PAWS, STS-B

Closed-book QA (3 datasets)
ARC (easy/chal.), NQ, TQA

Struct to text (4 datasets)
CommonGen, DART, E2ENLG, WEBNLG

Translation (8 datasets)
ParaCrawl EN/DE, ParaCrawl EN/ES, ParaCrawl EN/FR, WMT-16 EN/CS, WMT-16 EN/DE, WMT-16 EN/FI, WMT-16 EN/RO, WMT-16 EN/RU, WMT-16 EN/TR

Reading comp. (5 datasets)
BoolQ, OBQA, DROP, SQuAD, MultiRC

Read. comp. w/ commonsense (2 datasets)
CosmosQA, ReCoRD

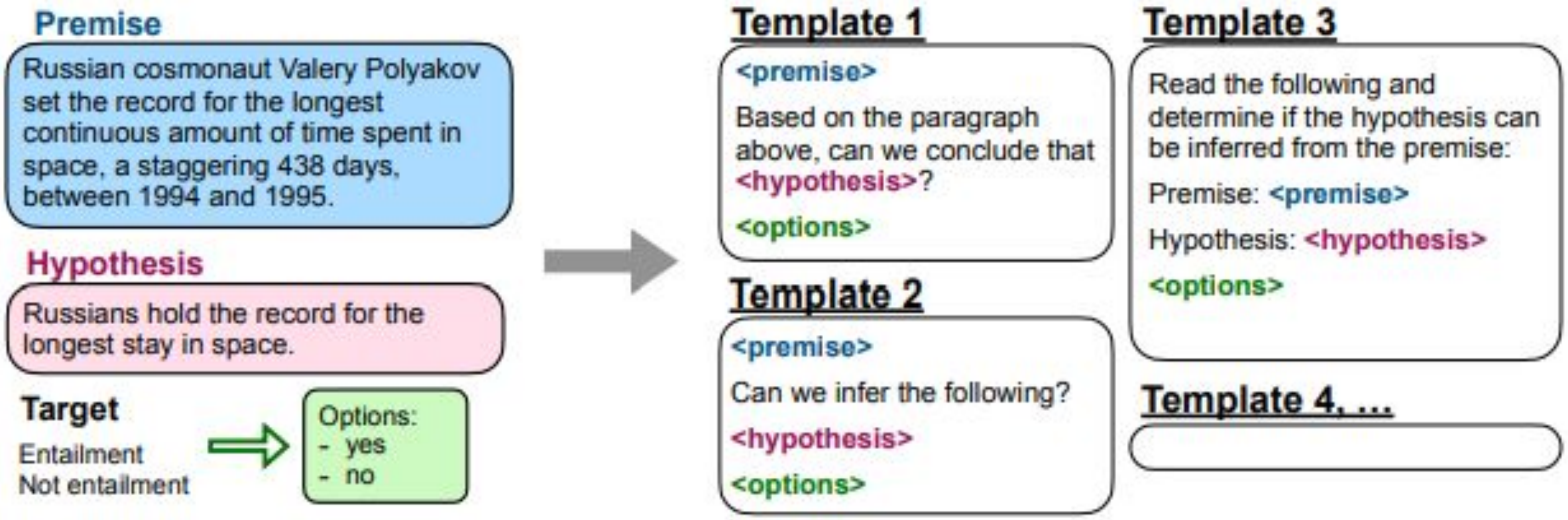
Coreference (3 datasets)
DPR, Winogrande, WSC273

Misc. (7 datasets)
CoQA, TREC, QuAC, CoLA, WIC, Math, Fix Punctuation (NLG)

Summarization (11 datasets)
AESLC, Multi-News, SamSum, AG News, Newsroom, Wiki Lingua EN, CNN-DM, Opin-Abs: iDebate, XSum, Gigaword, Opin-Abs: Movie

Reformatting data into instruction-form

Example formatting data for Natural Language Inference:



Premise

Russian cosmonaut Valery Polyakov set the record for the longest continuous amount of time spent in space, a staggering 438 days, between 1994 and 1995.

Hypothesis

Russians hold the record for the longest stay in space.

Target

Entailment
Not entailment

Options:
- yes
- no

Template 1

<premise>

Based on the paragraph above, can we conclude that <hypothesis>?

<options>

Template 2

<premise>

Can we infer the following?

<hypothesis>

<options>

Template 3

Read the following and determine if the hypothesis can be inferred from the premise:

Premise: <premise>

Hypothesis: <hypothesis>

<options>

Template 4, ...

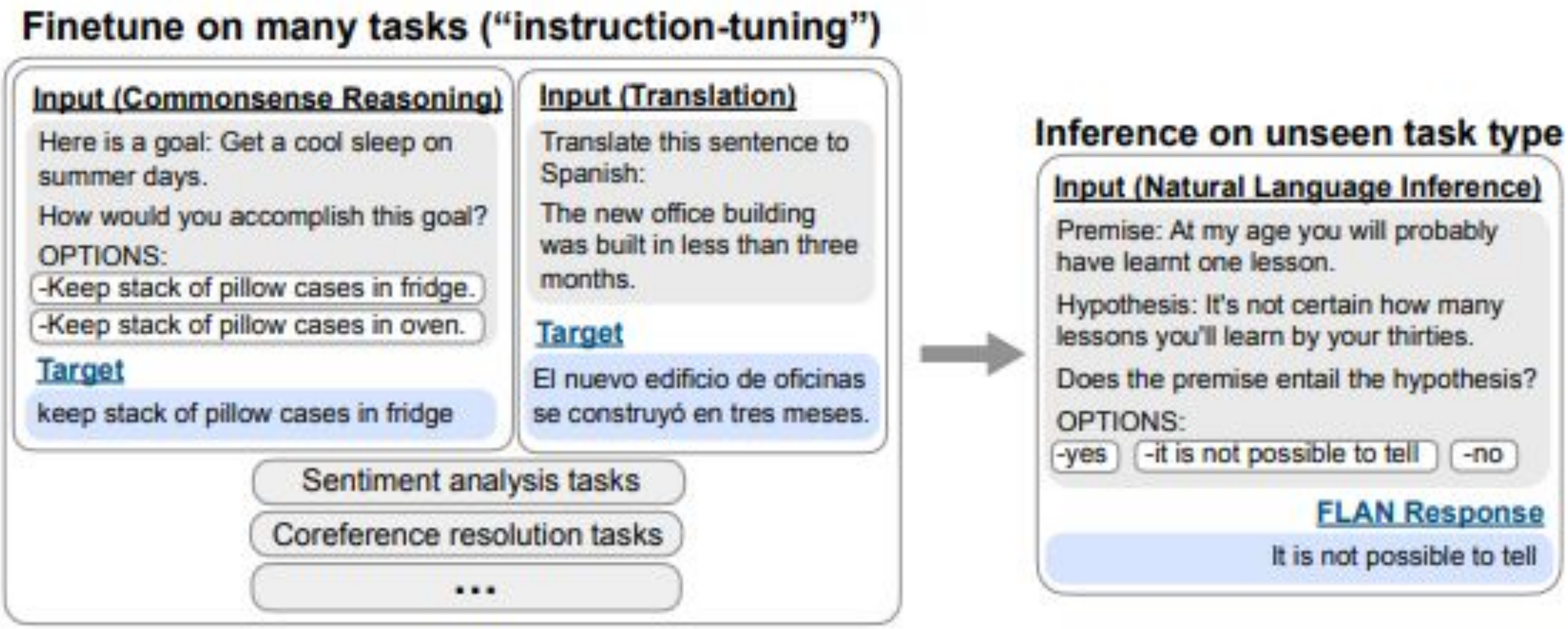
Reformatting each dataset

- For each dataset, we manually compose ten unique templates that use natural language instructions to describe the task for that dataset.
- For each dataset, there are at least three templates that “turned the task around.”

Evaluating zero-shot performance

- A dataset of a certain task is only considered unseen if no other dataset in it’s task cluster was seen during instruction-tuning.
- To evaluate FLAN zero-shot performance, we train several FLAN models on all task-clusters, holding out each cluster for each model.
- Test zero-shot performance on unseen task-clusters for each model.

Evaluating zero-shot performance (continued)



Finetune on many tasks ("instruction-tuning")

Input (Commonsense Reasoning)

Here is a goal: Get a cool sleep on summer days.

How would you accomplish this goal?

OPTIONS:

-Keep stack of pillow cases in fridge.

-Keep stack of pillow cases in oven.

Target

keep stack of pillow cases in fridge

Input (Translation)

Translate this sentence to Spanish:

The new office building was built in less than three months.

Target

El nuevo edificio de oficinas se construyó en tres meses.

Sentiment analysis tasks

Coreference resolution tasks

...

Inference on unseen task type

Input (Natural Language Inference)

Premise: At my age you will probably have learnt one lesson.

Hypothesis: It's not certain how many lessons you'll learn by your thirties.

Does the premise entail the hypothesis?

OPTIONS:

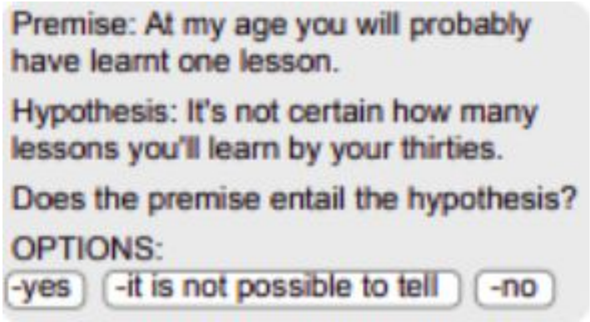
-yes -it is not possible to tell -no

FLAN Response

It is not possible to tell

Evaluating zero-shot performance (continued)

- When evaluating performance on classification tasks, there may be multiple ways of saying “yes” or “no,” reducing the respective probabilities.
- An options token is added to classification instructions to make evaluation more accurate.



Premise: At my age you will probably have learnt one lesson.

Hypothesis: It's not certain how many lessons you'll learn by your thirties.

Does the premise entail the hypothesis?

OPTIONS:

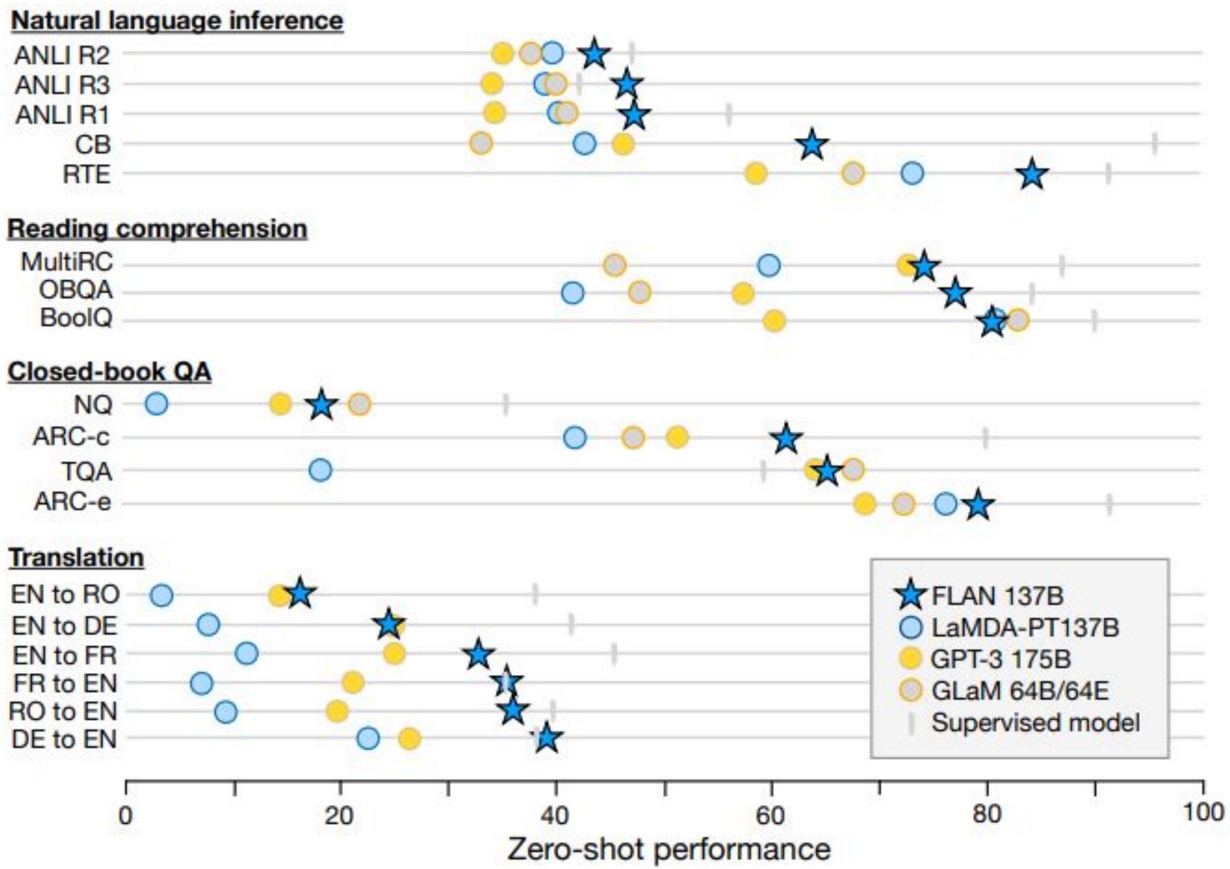
-yes -it is not possible to tell -no

Model Architecture

- Using LaMDA-PT architecture:
  - Pretrained, dense left-to-right, decoder only transformer language model
- FLAN is instruction tuned LaMDA-PT:
  - Size of datasets is balanced.
  - Random-sampling from each dataset
  - Instruction tuning takes 60 hours on a TPU-v3

Results

- Instruction tuning significantly improves LaMDA-PT on most datasets:
  - LaMDA-PT control test used the same prompts as GPT-3.
- Instruction tuning is very effective on tasks naturally verbalized as instructions and is less effective on tasks directly formulated as language modeling
  - Commonsense reasoning, coreference resolution tasks

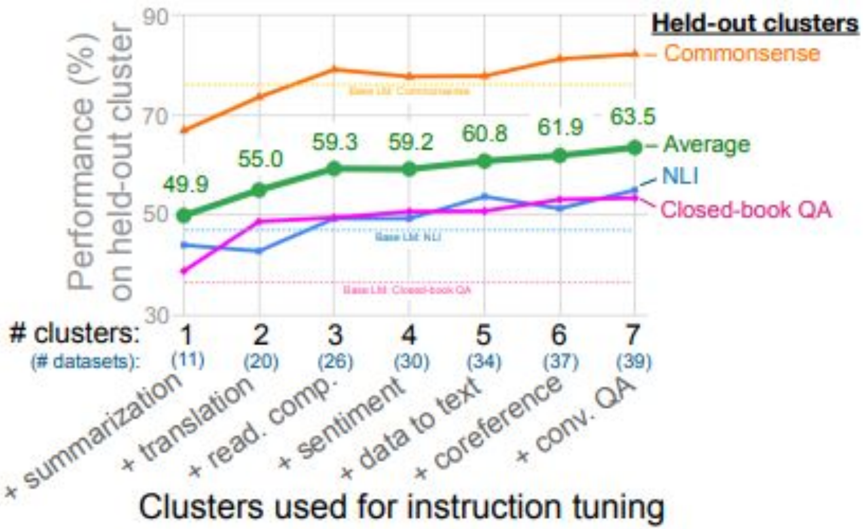

Natural language inference
ANLI R2
ANLI R3
ANLI R1
CB
RTE
Reading comprehension
MultiRC
OBQA
BoolQ
Closed-book QA
NQ
ARC-c
TQA
ARC-e
Translation
EN to RO
EN to DE
EN to FR
FR to EN
RO to EN
DE to EN
FLAN 137B
LaMDA-PT137B
GPT-3 175B
GLaM 64B/64E
Supervised model
0
20
40
60
80
100
Zero-shot performance

Additional Results

- FLAN does not improve performance for many language modeling tasks:
  - **Commonsense reasoning**: Applying real-world commonsense reasoning to multiple choice questions
  - **Coreference resolution tasks**: Identifying what in a prompt refers to the same entity
- When the downstream task is the same as the original language modeling pre-training objective, formatting the prompt as a pure natural language instruction prompt is not helpful.

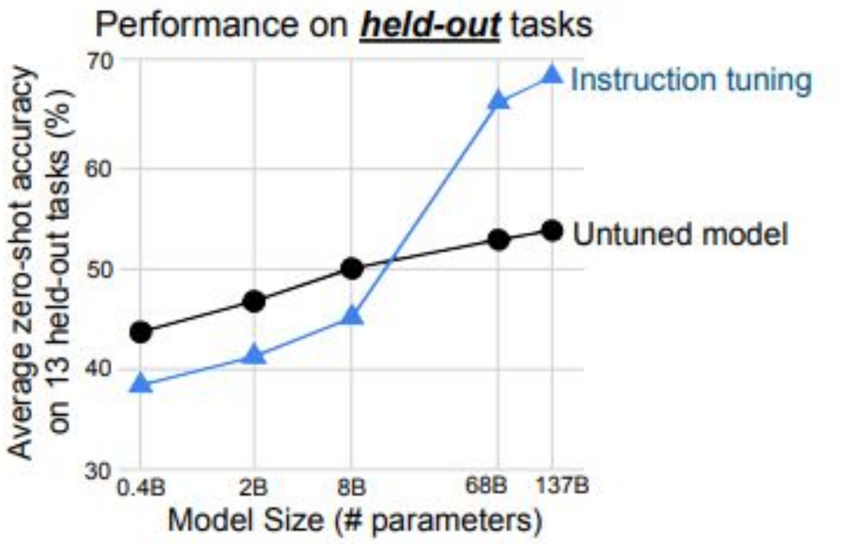
Ablation Studies

- Performance of FLAN as we add more task-clusters in instruction-tuning:



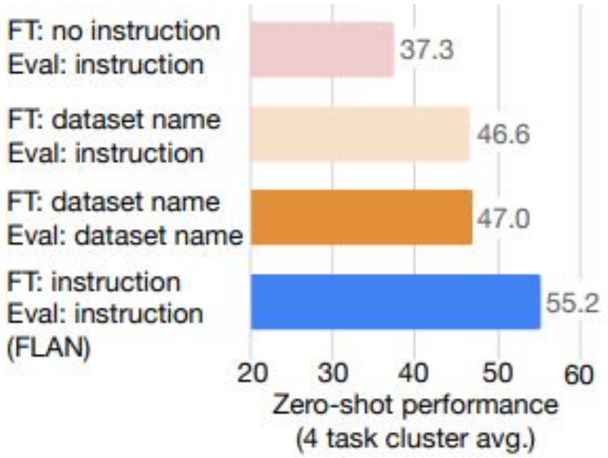
Ablation studies (continued)

- Scaling
  - Instruction tuning improves zero-shot performance on models on the order of >60B parameters
  - Instruction tuning hurts performance for held out tasks on smaller models <10B
  - This may be because smaller models are saturated more easily



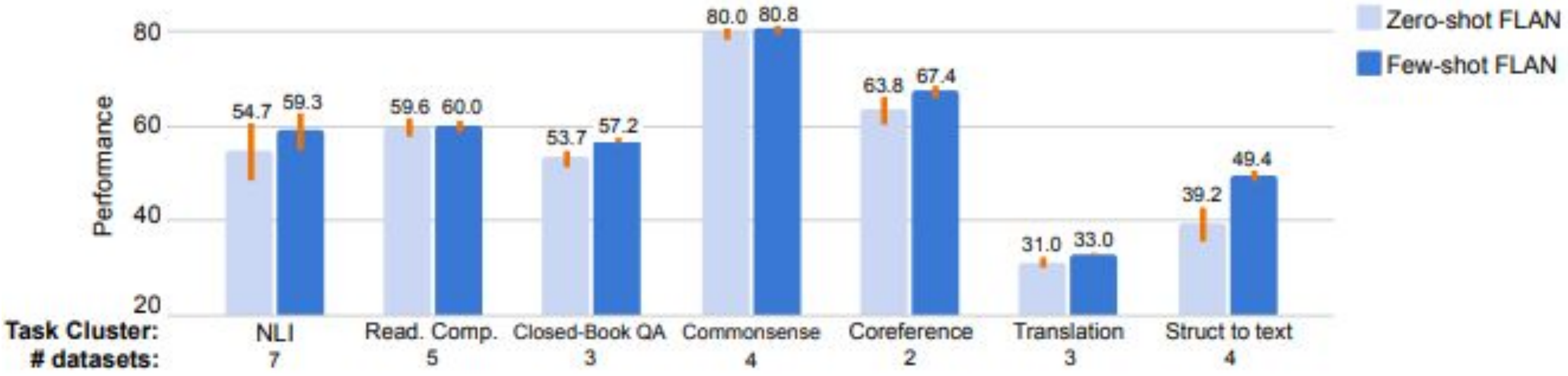
Ablation studies (continued)

- Multi-task tuning without instruction templates
- Dataset name
    - [Translation: WMT’14 to French] The dog runs.)
- Ablation configurations performed substantially worse than FLAN



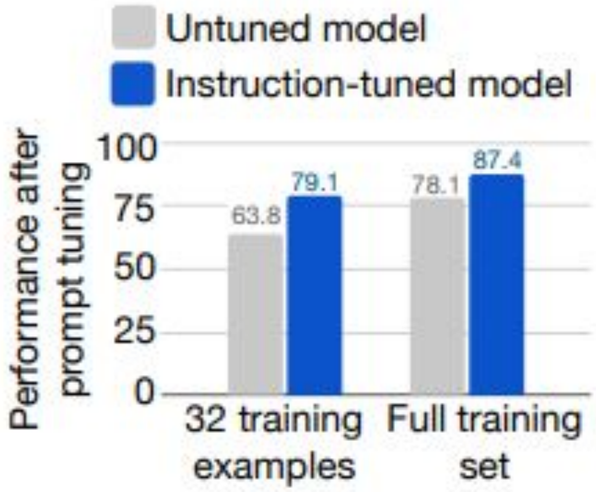
Few-shot Instruction training on FLAN

- Concatenation of few-shot instruction exemplars
- Randomly sampling 16 exemplars from training set



Prompt-Tuning FLAN

- Prompts are similar to instructions, but are produced particularly for guiding the behavior of a model
- FLAN was trained on continuous prompts for tasks that were not seen during instruction tuning.
- Prompt tuning on FLAN resulted in better improvement than on LaMDA-PT



Limitations

- Subjectivity in assigning tasks to clusters
- FLAN was trained using mainly short instructions (1 sentence)
- Instructions in the pre-training data of LaMDA-PT
  - Post-hoc analysis found that the results were not substantially impacted by this
- FLAN is 136B parameters, making it particularly costly to use

Conclusions

- FLAN is favorable on a number of benchmarks against GPT-3
- Instruction tuning may be effective for improving the zero-shot performance of large-scale models and for interpreting natural language queries more effectively
- Possible implications:
  - Instruction tuning for zero-shot performance enhancement may have implications for the arguments about general LLM model development vs. speciality produced LLM models

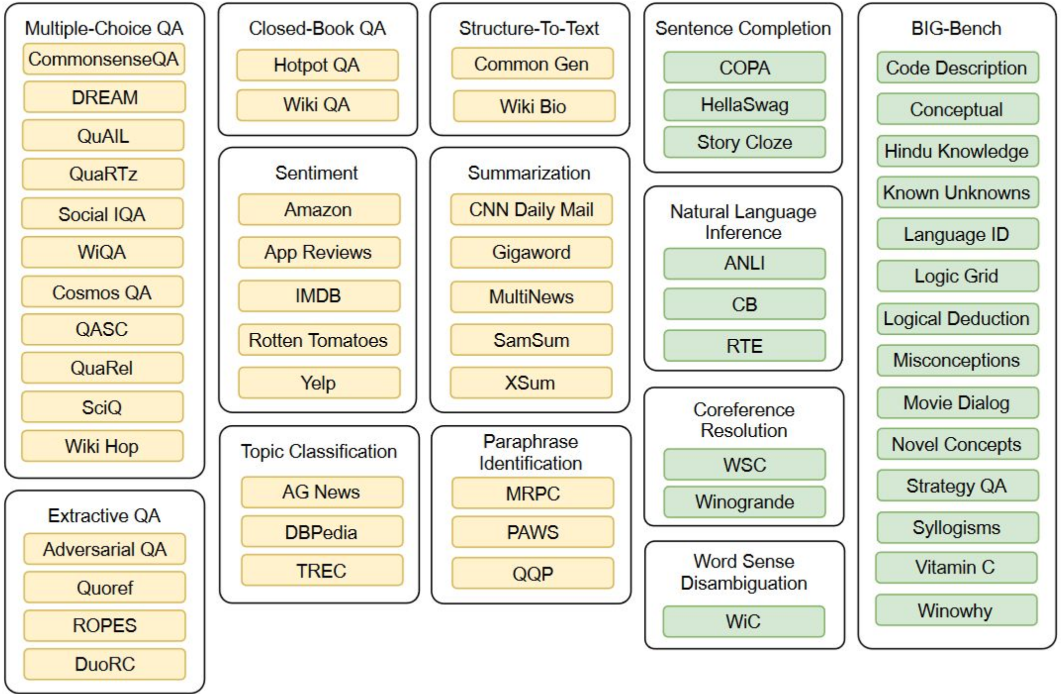
Multitask Prompted Training Enables Zero-Shot Task Generalization

Victor Sanh*, Albert Webson*, Colin Raffel*, Stephen H. Bach*, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Tali Bers, Stella Biderman, Leo Gao, Thomas Wolf, Alexander M. Rush

Motivation

- **Observation**: LLMs perform relatively well on unseen tasks not explicitly trained to perform.
- **Hypothesis**: This is caused by an implicit process of multitask learning.
- **Goal**: Train LMs using explicit multitask learning.

Datasets and Held-out Tasks



Multiple-Choice QA
- CommonsenseQA
- DREAM
- QuAIL
- QuaRTz
- Social IQA
- WiQA
- Cosmos QA
- QASC
- QuaRel
- SciQ
- Wiki Hop

Extractive QA
- Adversarial QA
- Quoref
- ROPES
- DuoRC

Closed-Book QA
- Hotpot QA
- Wiki QA

Sentiment
- Amazon
- App Reviews
- IMDB
- Rotten Tomatoes
- Yelp

Topic Classification
- AG News
- DBPedia
- TREC

Structure-To-Text
- Common Gen
- Wiki Bio

Summarization
- CNN Daily Mail
- Gigaword
- MultiNews
- SamSum
- XSum

Paraphrase Identification
- MRPC
- PAWS
- QQP

Sentence Completion
- COPA
- HellaSwag
- Story Cloze

Natural Language Inference
- ANLI
- CB
- RTE

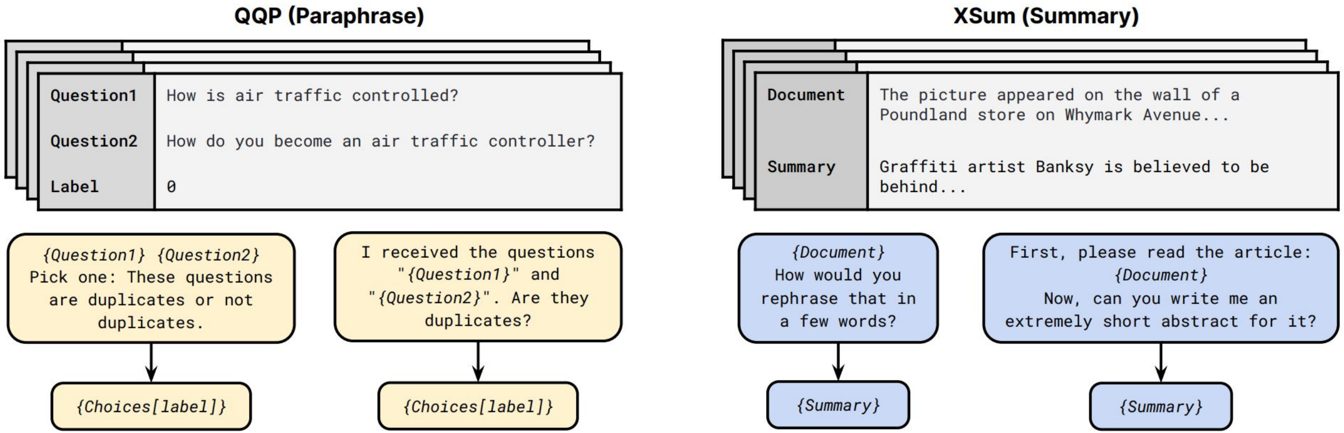
Coreference Resolution
- WSC
- Winogrande

Word Sense Disambiguation
- WiC

BIG-Bench
- Code Description
- Conceptual
- Hindu Knowledge
- Known Unknowns
- Language ID
- Logic Grid
- Logical Deduction
- Misconceptions
- Movie Dialog
- Novel Concepts
- Strategy QA
- Syllogisms
- Vitamin C
- Winowhy

12 tasks and 62 datasets

Prompt Generation: PromptSource and P3



QQP (Paraphrase)

Question1	How is air traffic controlled?
Question2	How do you become an air traffic controller?
Label	0

{Question1} {Question2} Pick one: These questions are duplicates or not duplicates.

→ *{Choices[label]}*

I received the questions "*{Question1}*" and "*{Question2}*". Are they duplicates?

→ *{Choices[label]}*

XSum (Summary)

Document	The picture appeared on the wall of a Poundland store on Whymark Avenue...
Summary	Graffiti artist Banksy is believed to be behind...

{Document} How would you rephrase that in a few words?

→ *{Summary}*

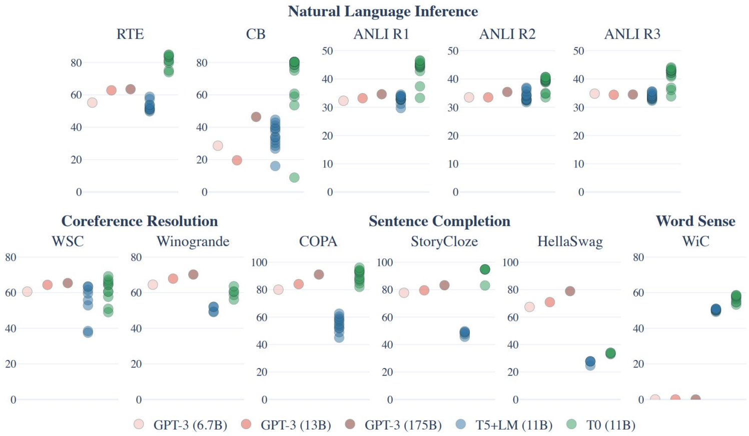
First, please read the article: *{Document}* Now, can you write me an extremely short abstract for it?

→ *{Summary}*

Experimental Setup

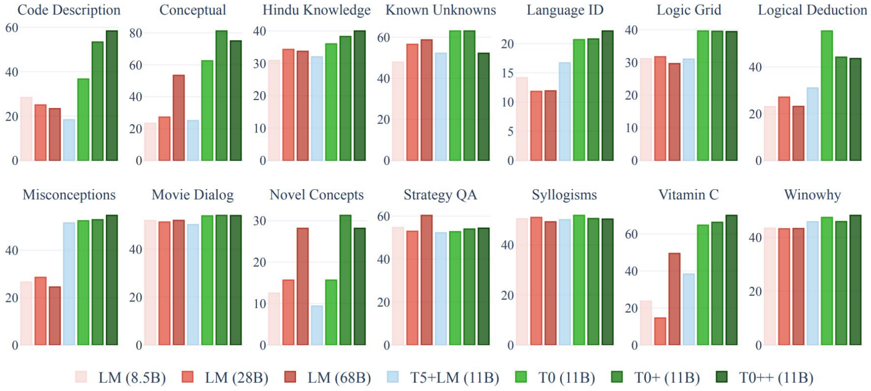
- **Pretrained model**: LM-adapted T5 (T5+LM)
- **Training**: T0, T0+, T0++
  - T0 is the trained version of T5+LM
  - T0+ includes GPT-3's evaluation datasets
  - T0++ includes GPT-3's datasets and SuperGLUE
- **Evaluation**:
  - Accuracy for performance metric
  - Log-likelihood for tasks with multiple choices
  - Median performance and IQR to measure robustness

Results: Does multitask prompted training improve generalization to held-out tasks?

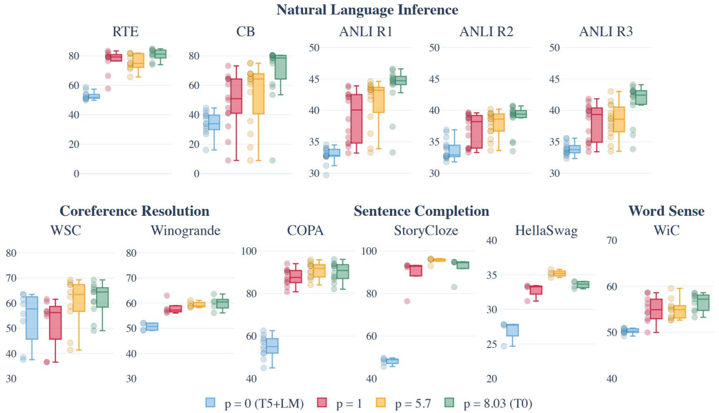


Model Performance on 4 Held-out Tasks

Model Performance on BIG-bench datasets

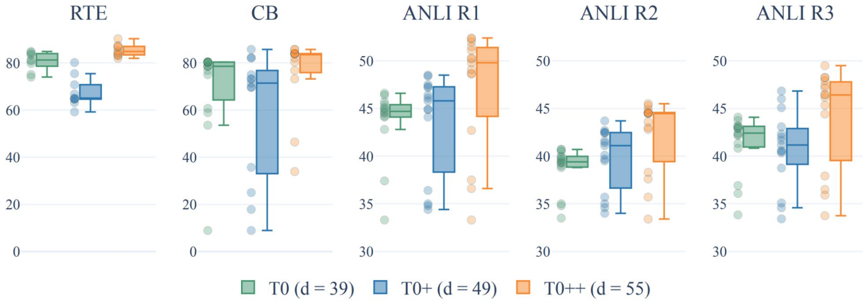


Results: Does training on a wider range of prompts improve robustness to prompt wording?



Ablation Study 1: Effect of More Prompts per Dataset

Ablation Study 2: Effect of Prompts from More Datasets



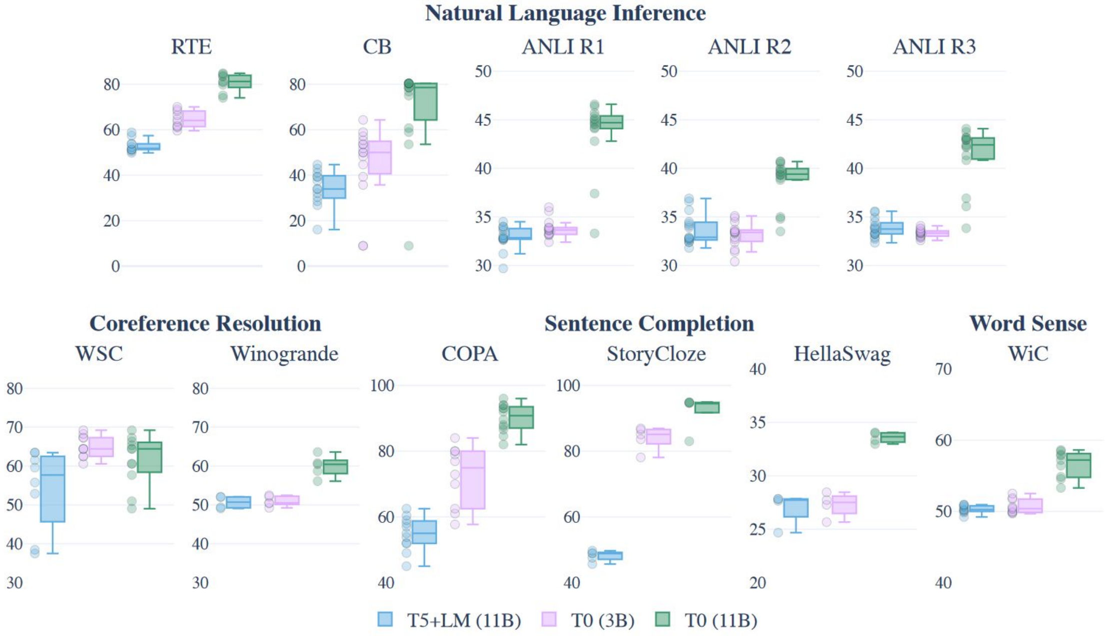
Results: Does training on a wider range of prompts improve robustness to prompt wording? (Cont.)

- T0 vs GPT-3
  - Median: 52.96%
  - IQR: 1.28

Discussion: FLAN (Wei et al., 2021)

- Main difference
  - Pretrained model
  - Evaluation with 1 held out task vs multiple tasks
- T0 and T0++ generally had better performance with 10x less parameters
- Underperformance on Winogrande and HellaSwag
  - HellaSwag: improved from 33.65% to 57.93%
  - Winogrande: no significant change

Discussion (Cont.)



Effect of pretrained model size

Discussion (Cont.)

- Possible reasons for result differences
  - Masked language modeling objective
  - Diverse prompts

Limitations

- Model Scaling
  - How does multitask learning affect large scale models?
- Task Taxonomy
  - Organizing by format vs by content
- Contamination analysis of pretraining corpus
  - HellaSwag - 9.12%
  - ANLI - 33.7% (premises) and 0.6% (hypotheses)
  - RTE - 11.0% (premises) and 5.2% (hypotheses)

Conclusion

- P3 dataset
- T0 (11B) and model variants has better zero shot ability than large scale models
  - GPT-3 (175B)
  - FLAN(137B)

Cross-Task Generalization via Natural Language Crowdsourcing Instructions

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Hannaneh Hajishirzi

University of Virginia | ENGINEERING
Department of Computer Science

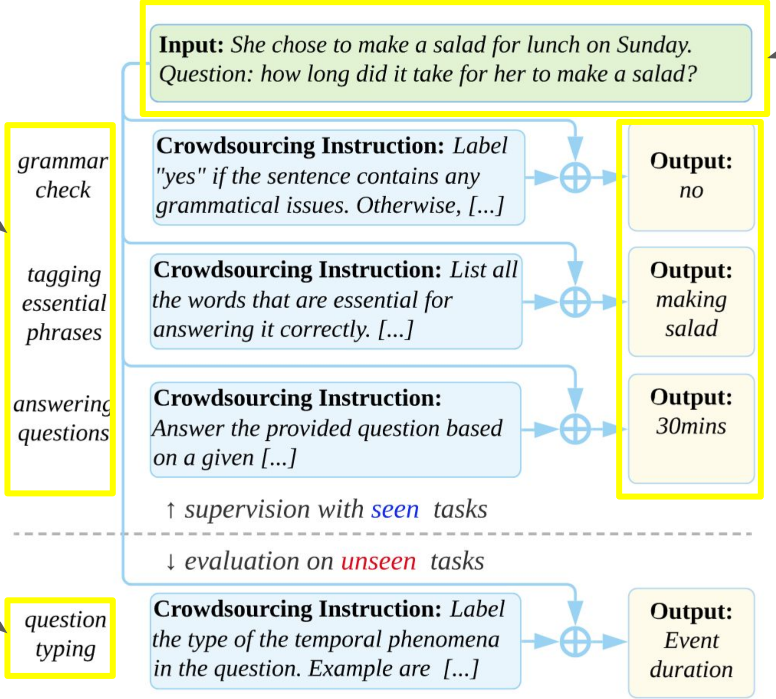
Motivation

- **Observation**: Conventional supervised models learned on individual datasets *struggle* with *generalization across tasks* (e.g., a question-answering system ↛ classification tasks).
- **Hypothesis**: Pre-trained LM can learn multiple *seen* tasks well in one training procedure, it possibly has ability to generalize to *unseen* tasks.
- **Goal**: Build a model that learns a new task by *understanding the human-readable instructions* that define it.

Motivation

Input text, shared across tasks

Different *seen* tasks



Input: *She chose to make a salad for lunch on Sunday. Question: how long did it take for her to make a salad?*

Crowdsourcing Instruction: *Label "yes" if the sentence contains any grammatical issues. Otherwise, [...]*

Output: *no*

Crowdsourcing Instruction: *List all the words that are essential for answering it correctly. [...]*

Output: *making salad*

Crowdsourcing Instruction: *Answer the provided question based on a given [...]*

Output: *30mins*

↑ *supervision with seen tasks*

↓ *evaluation on unseen tasks*

Crowdsourcing Instruction: *Label the type of the temporal phenomena in the question. Example are [...]*

Output: *Event duration*

Output by LM

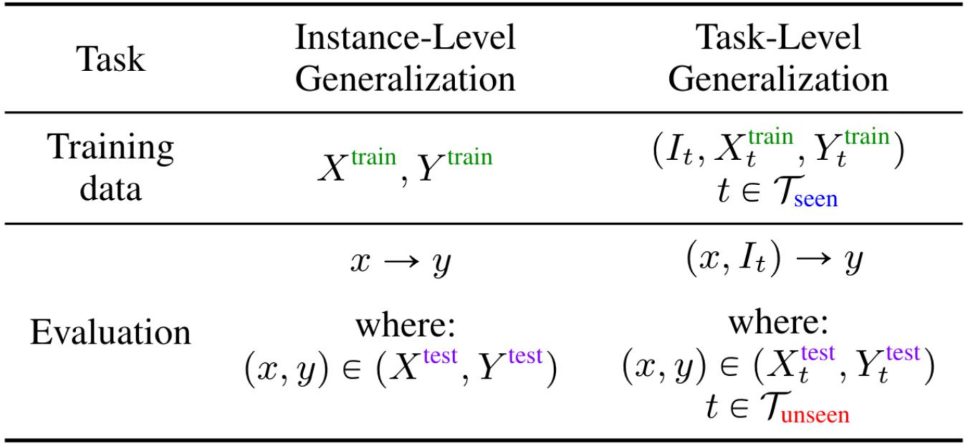
unseen task to generalize

Evaluate if training on unseen tasks *benefits the LM answering*

Definition of Cross-task models

Cross-task models: learn a model M that *at inference* obtains the output y given the *input x* and the *task instruction I_t*:

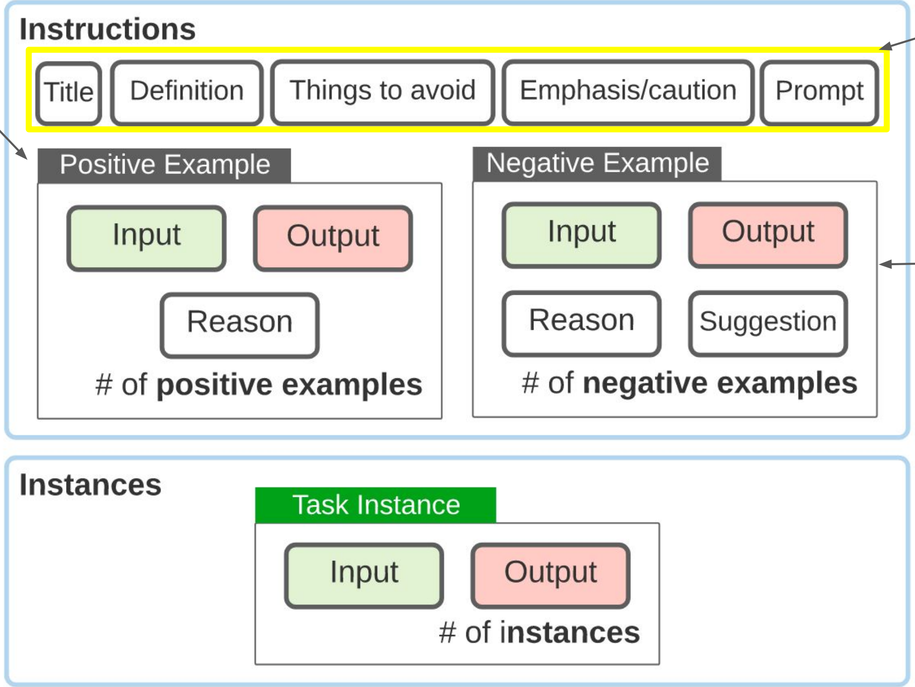
$$M(x) = y \text{ for } (x, y) \in (X_t^{\text{train}}, Y_t^{\text{train}})$$



Task	Instance-Level Generalization	Task-Level Generalization
Training data	$X^{\text{train}}, Y^{\text{train}}$	$(I_t, X_t^{\text{train}}, Y_t^{\text{train}})$ $t \in \mathcal{T}_{\text{seen}}$
Evaluation	$x \rightarrow y$ where: $(x, y) \in (X^{\text{test}}, Y^{\text{test}})$	$(x, I_t) \rightarrow y$ where: $(x, y) \in (X_t^{\text{test}}, Y_t^{\text{test}})$ $t \in \mathcal{T}_{\text{unseen}}$

Proposed Dataset Schema: NATURAL INSTRUCTIONS

An example of desired instance of the task.



Description of a NLP task.

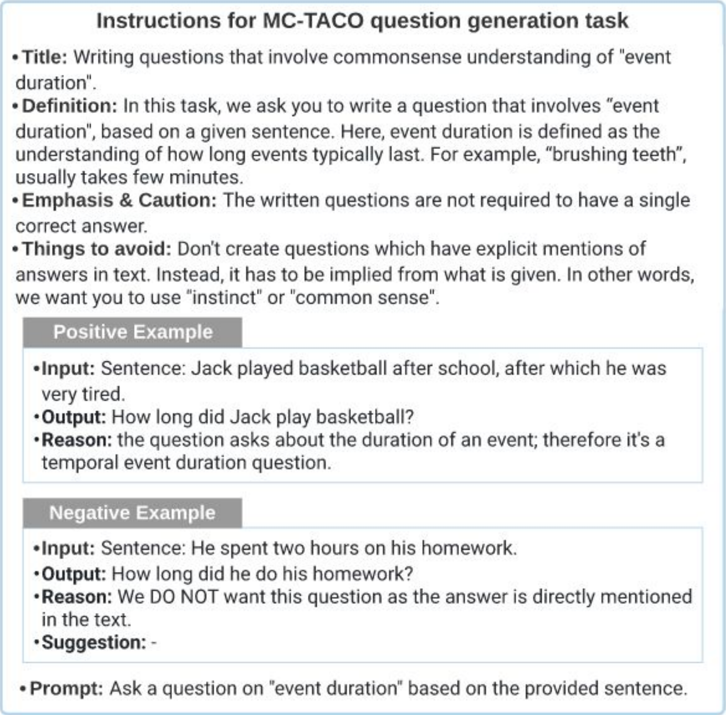
An example of undesired instance of the task.

Many instances of the task.

Proposed Dataset Schema: NATURAL INSTRUCTIONS

- **TITLE** provides a *high-level description* of a task and its associated skill.
- **PROMPT** is a single sentence *command* for the instructions.
- **DEFINITION** provides the *core detailed instructions* for a task.
- **THINGS TO AVOID** contain instructions regarding undesirable *annotations that must be avoided*
- **EMPHASIS AND CAUTION** are highlighted statements to be *emphasized* or warned against.
- **POSITIVE EXAMPLES** contain inputs/outputs similar to the input given to a worker/system and its *expected output*.
- **NEGATIVE EXAMPLES** contain *unexpected* inputs/outputs.

Proposed Dataset Example: NATURAL INSTRUCTIONS



Instructions for MC-TACO question generation task

- **Title:** Writing questions that involve commonsense understanding of "event duration".
- **Definition:** In this task, we ask you to write a question that involves “event duration", based on a given sentence. Here, event duration is defined as the understanding of how long events typically last. For example, “brushing teeth”, usually takes few minutes.
- **Emphasis & Caution:** The written questions are not required to have a single correct answer.
- **Things to avoid:** Don't create questions which have explicit mentions of answers in text. Instead, it has to be implied from what is given. In other words, we want you to use "instinct" or "common sense".

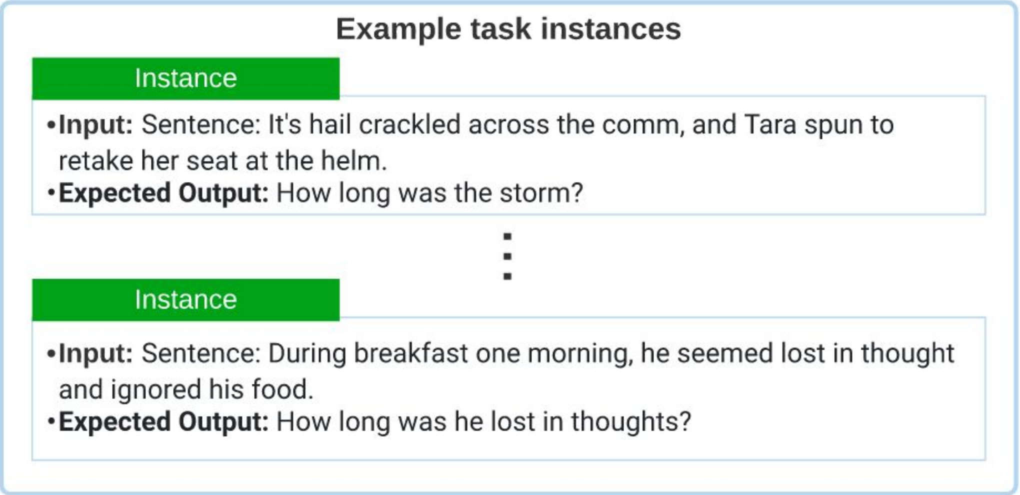
Positive Example

- **Input:** Sentence: Jack played basketball after school, after which he was very tired.
- **Output:** How long did Jack play basketball?
- **Reason:** the question asks about the duration of an event; therefore it's a temporal event duration question.

Negative Example

- **Input:** Sentence: He spent two hours on his homework.
- **Output:** How long did he do his homework?
- **Reason:** We DO NOT want this question as the answer is directly mentioned in the text.
- **Suggestion:** -

- **Prompt:** Ask a question on "event duration" based on the provided sentence.



Example task instances

Instance

- **Input:** Sentence: It's hail crackled across the comm, and Tara spun to retake her seat at the helm.
- **Expected Output:** How long was the storm?

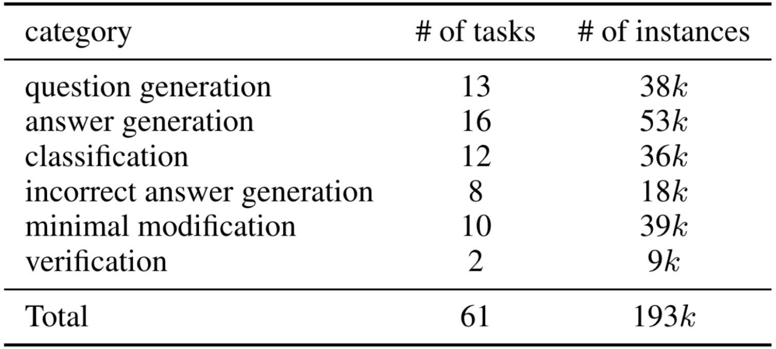
⋮

Instance

- **Input:** Sentence: During breakfast one morning, he seemed lost in thought and ignored his food.
- **Expected Output:** How long was he lost in thoughts?

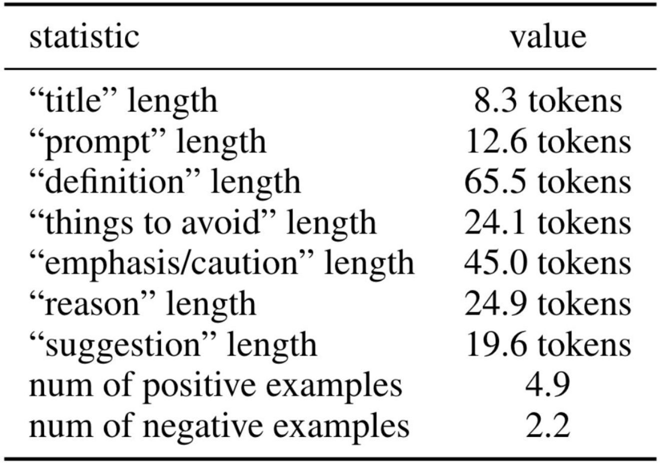
Proposed Dataset Statistics: NATURAL INSTRUCTIONS

Dataset is collected from *crowdsourcing NLP datasets* and *mapped to the schema* with certain procedures.



category	# of tasks	# of instances
question generation	13	38k
answer generation	16	53k
classification	12	36k
incorrect answer generation	8	18k
minimal modification	10	39k
verification	2	9k
Total	61	193k

Table 2: Task categories and their statistics.

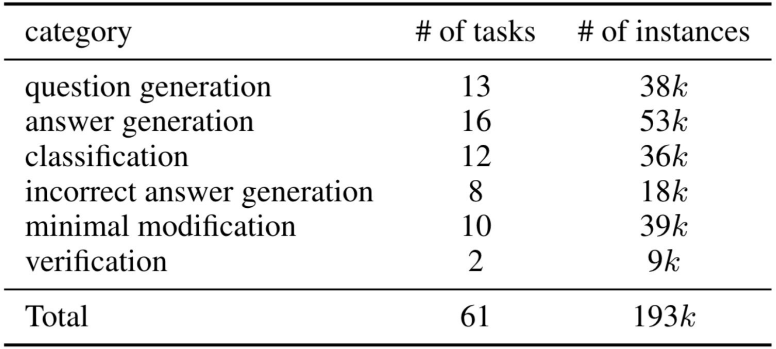


statistic	value
“title” length	8.3 tokens
“prompt” length	12.6 tokens
“definition” length	65.5 tokens
“things to avoid” length	24.1 tokens
“emphasis/caution” length	45.0 tokens
“reason” length	24.9 tokens
“suggestion” length	19.6 tokens
num of positive examples	4.9
num of negative examples	2.2

Table 3: Statistics of NATURAL INSTRUCTIONS

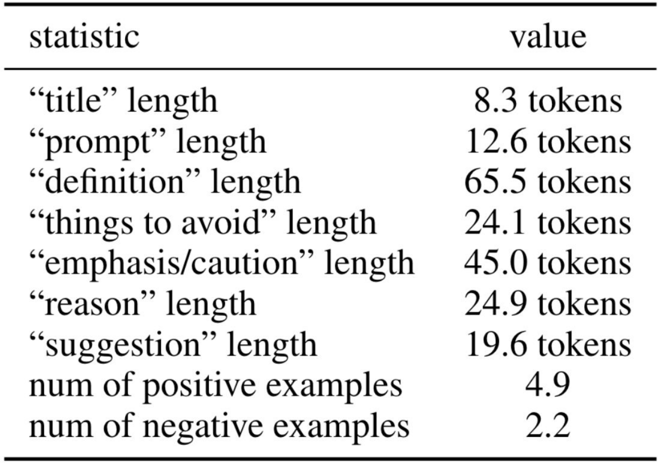
Proposed Dataset Statistics: NATURAL INSTRUCTIONS

Dataset is collected from *crowdsourcing NLP datasets* and *mapped to the schema.* with certain procedures.



category	# of tasks	# of instances
question generation	13	38k
answer generation	16	53k
classification	12	36k
incorrect answer generation	8	18k
minimal modification	10	39k
verification	2	9k
Total	61	193k

Table 2: Task categories and their statistics.



statistic	value
"title" length	8.3 tokens
"prompt" length	12.6 tokens
"definition" length	65.5 tokens
"things to avoid" length	24.1 tokens
"emphasis/caution" length	45.0 tokens
"reason" length	24.9 tokens
"suggestion" length	19.6 tokens
num of positive examples	4.9
num of negative examples	2.2

Table 3: Statistics of NATURAL INSTRUCTIONS

Experiment Setup

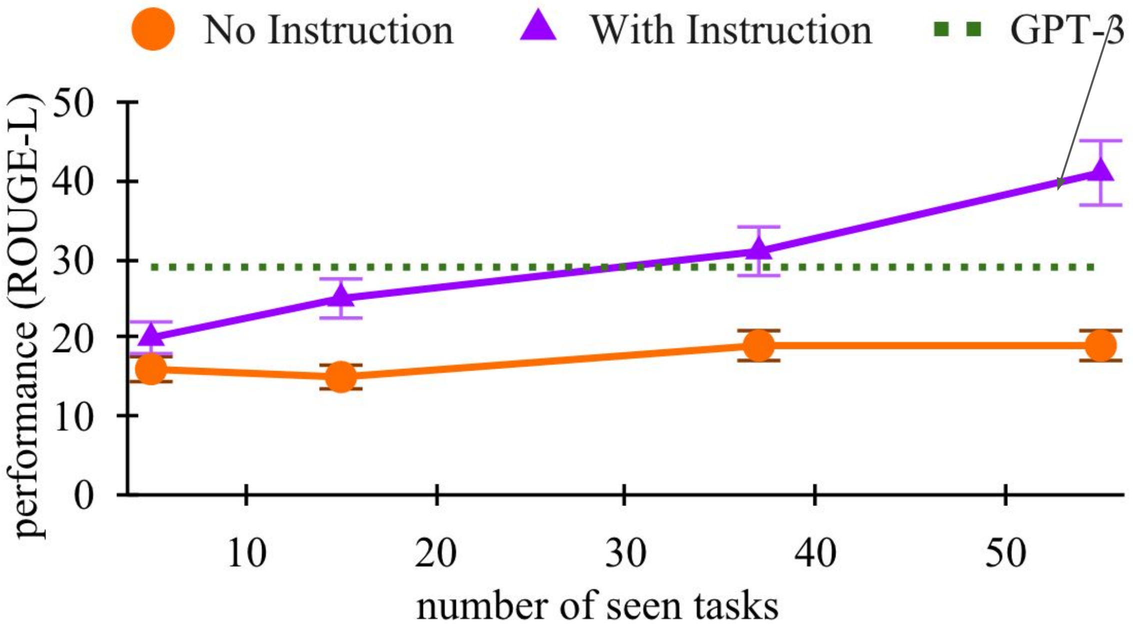
- **Random split**: *Two tasks from each task category* are randomly selected for evaluation, and the rest of the tasks are used for training, i.e., *12 unseens* tasks and *49 tasks in seen* tasks.
- **Leave-one-out generalization**:
  - **Leave-one-category:** evaluates how well a model *generalizes to a task category* if it is trained on others – no task of that category is in seen tasks.
  - **Leave-one-dataset:** evaluates how well a model can *generalize to all tasks in one dataset* if it is trained on all other tasks – no task of that dataset is in seen tasks.
  - **Leave-one-task:** evaluates how well a model can *learn a single task* by training on all other tasks.

Models and Evaluation metric

- **Models**:
  - **BART**: A encoder-decoder model (140M), we *finetune it with seen tasks and evaluate on unseen tasks*.
  - **GPT3**: A autoregressive LM (175B), we use it *off-the-shelf* on the evaluation in unseen tasks.
- **Evaluation**:
  - **ROUGE-L**: the average ratio of the length of *largest common sequence* (LCS) of the LM-generated answers and the ground-truth answers to the length of ground-truth answers (GTA), i.e., avg(len(LCS)/len(GTA)).

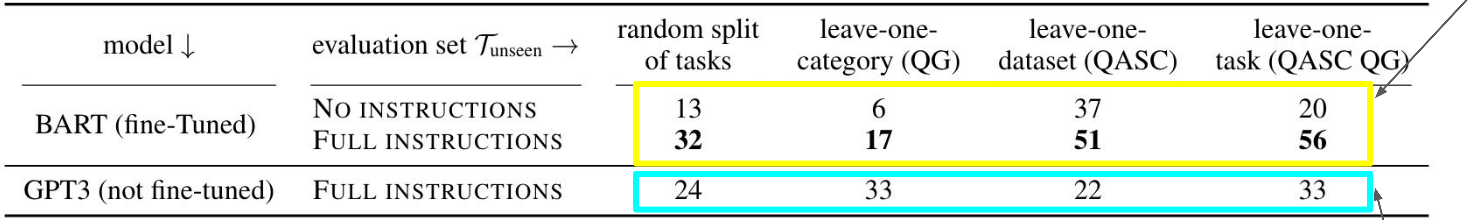
Analysis with increasing #seen tasks

Increasing #seen tasks in finetuning is *beneficial*
if evaluated with full instruction.



Analysis under various evaluation splits

Fine-tuned BART shows improved performance when provided with instructions.

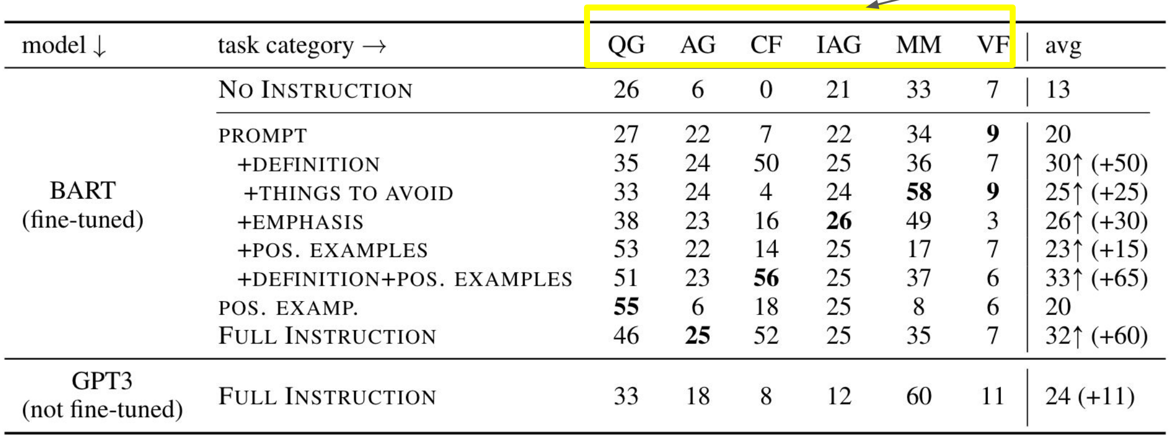


model ↓	evaluation set $\mathcal{T}_{\text{unseen}}$ →	random split of tasks	leave-one-category (QG)	leave-one-dataset (QASC)	leave-one-task (QASC QG)
BART (fine-Tuned)	NO INSTRUCTIONS	13	6	37	20
	FULL INSTRUCTIONS	**32**	**17**	**51**	**56**
GPT3 (not fine-tuned)	FULL INSTRUCTIONS	24	33	22	33

Fine-tuned BART achieves better GPT3, a much larger model.

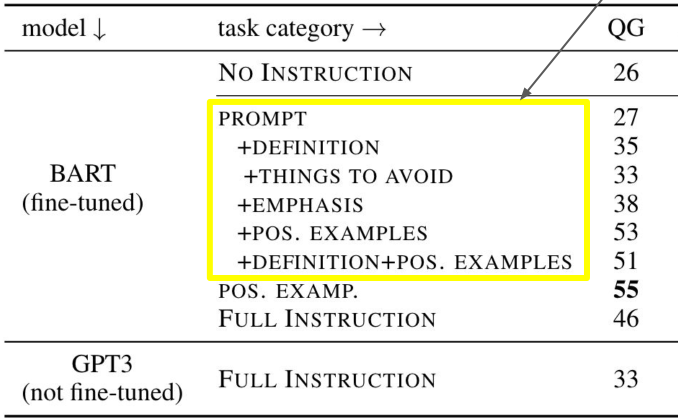
Analysis under different task categories

Task Categories: *QG*: Question Generation, *AG*: Answer Generation, *CF*: Classification, *IAG*: Incorrect Answer Generation, *MM*: Minimal Text Modification, *VF*: Verification.

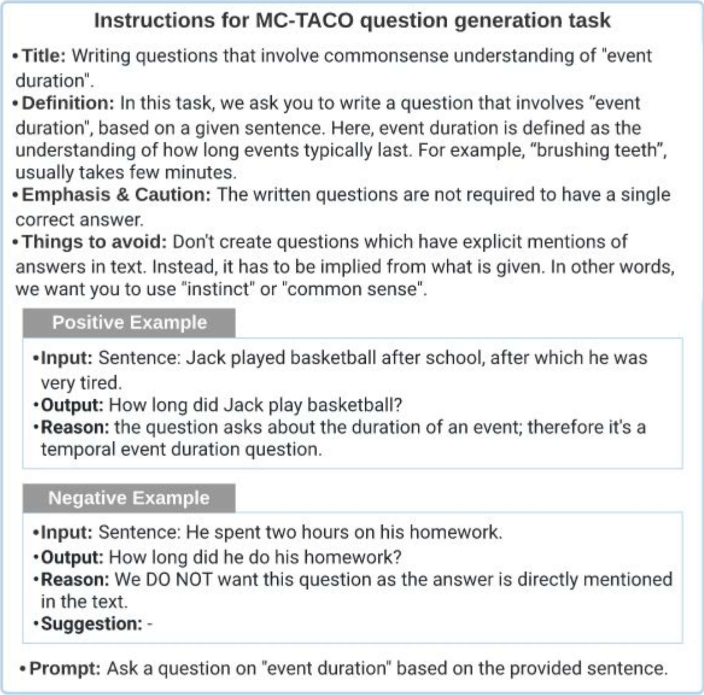


model ↓	task category →	QG	AG	CF	IAG	MM	VF	avg
BART (fine-tuned)	NO INSTRUCTION	26	6	0	21	33	7	13
	PROMPT	27	22	7	22	34	**9**	20
	+DEFINITION	35	24	50	25	36	7	30↑ (+50)
	+THINGS TO AVOID	33	24	4	24	**58**	**9**	25↑ (+25)
	+EMPHASIS	38	23	16	**26**	49	3	26↑ (+30)
	+POS. EXAMPLES	53	22	14	25	17	7	23↑ (+15)
	+DEFINITION+POS. EXAMPLES	51	23	**56**	25	37	6	33↑ (+65)
	POS. EXAMP.	**55**	6	18	25	8	6	20
	FULL INSTRUCTION	46	**25**	52	25	35	7	32↑ (+60)
GPT3 (not fine-tuned)	FULL INSTRUCTION	33	18	8	12	60	11	24 (+11)

Analysis under different task categories



model ↓	task category →	QG
	No Instruction	26
BART (fine-tuned)	PROMPT	27
	+DEFINITION	35
	+THINGS TO AVOID	33
	+EMPHASIS	38
	+POS. EXAMPLES	53
	+DEFINITION+POS. EXAMPLES	51
	POS. EXAMP.	**55**
	Full Instruction	46
GPT3 (not fine-tuned)	Full Instruction	33



Instructions for MC-TACO question generation task

- **Title:** Writing questions that involve commonsense understanding of "event duration".
- **Definition:** In this task, we ask you to write a question that involves "event duration", based on a given sentence. Here, event duration is defined as the understanding of how long events typically last. For example, "brushing teeth", usually takes few minutes.
- **Emphasis & Caution:** The written questions are not required to have a single correct answer.
- **Things to avoid:** Don't create questions which have explicit mentions of answers in text. Instead, it has to be implied from what is given. In other words, we want you to use "instinct" or "common sense".

Positive Example

- **Input:** Sentence: Jack played basketball after school, after which he was very tired.
- **Output:** How long did Jack play basketball?
- **Reason:** the question asks about the duration of an event; therefore it's a temporal event duration question.

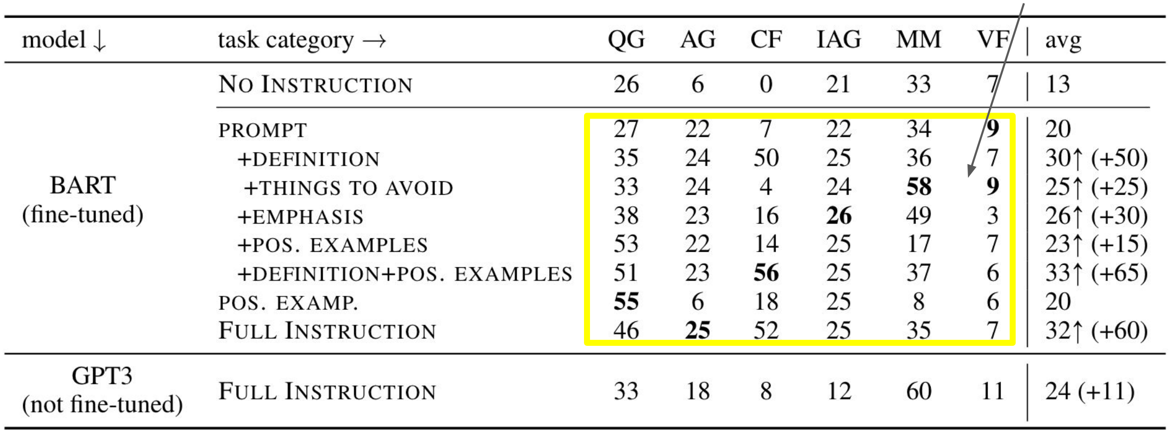
Negative Example

- **Input:** Sentence: He spent two hours on his homework.
- **Output:** How long did he do his homework?
- **Reason:** We DO NOT want this question as the answer is directly mentioned in the text.
- **Suggestion:** -

- **Prompt:** Ask a question on "event duration" based on the provided sentence.

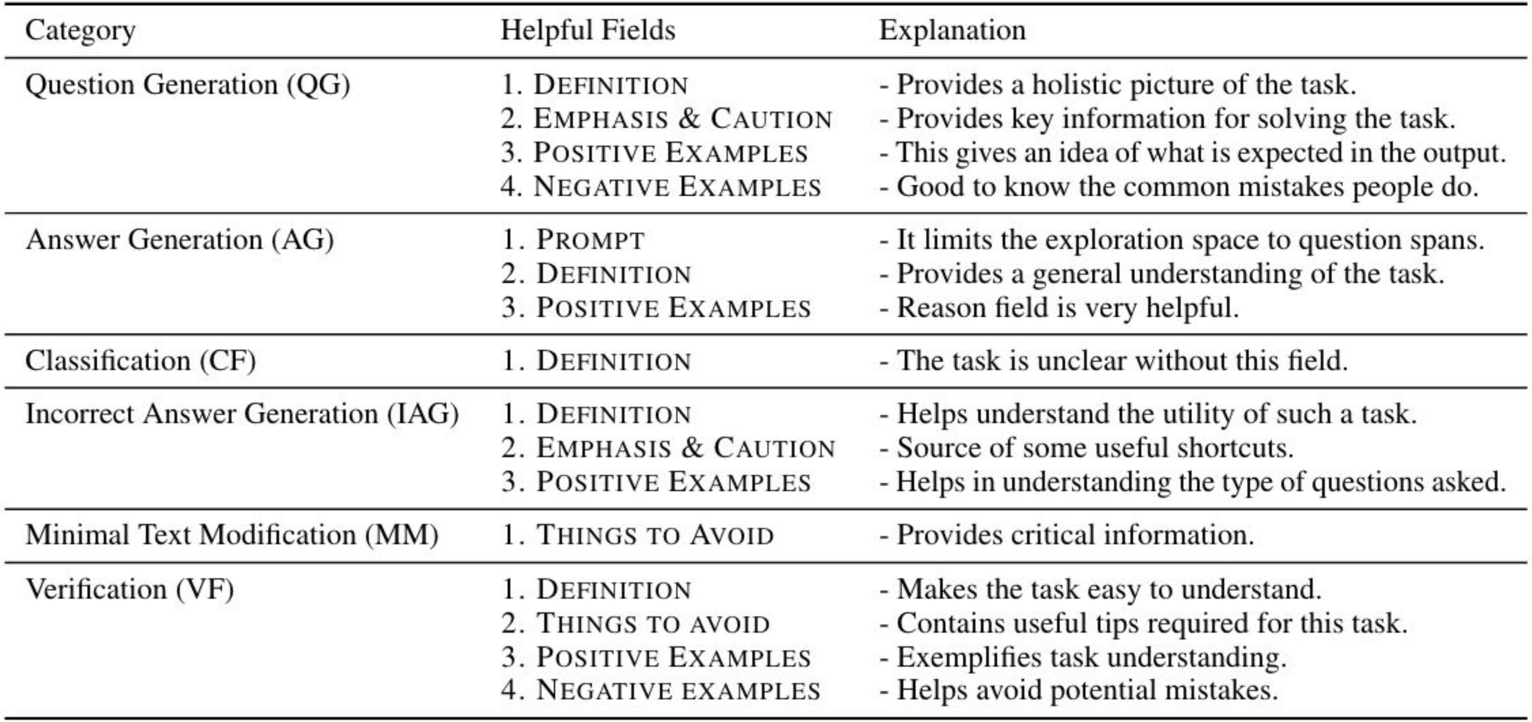
Analysis under different task categories

the benefit of the instruction elements seems to *depend on the target task category*. Full instruction is *not always the best*.



model ↓	task category →	QG	AG	CF	IAG	MM	VF	avg
BART (fine-tuned)	NO INSTRUCTION	26	6	0	21	33	7	13
	PROMPT	27	22	7	22	34	**9**	20
	+DEFINITION	35	24	50	25	36	7	30↑ (+50)
	+THINGS TO AVOID	33	24	4	24	**58**	**9**	25↑ (+25)
	+EMPHASIS	38	23	16	**26**	49	3	26↑ (+30)
	+POS. EXAMPLES	53	22	14	25	17	7	23↑ (+15)
	+DEFINITION+POS. EXAMPLES	51	23	**56**	25	37	6	33↑ (+65)
	POS. EXAMP.	**55**	6	18	25	8	6	20
	FULL INSTRUCTION	46	**25**	52	25	35	7	32↑ (+60)
GPT3 (not fine-tuned)	FULL INSTRUCTION	33	18	8	12	60	11	24 (+11)

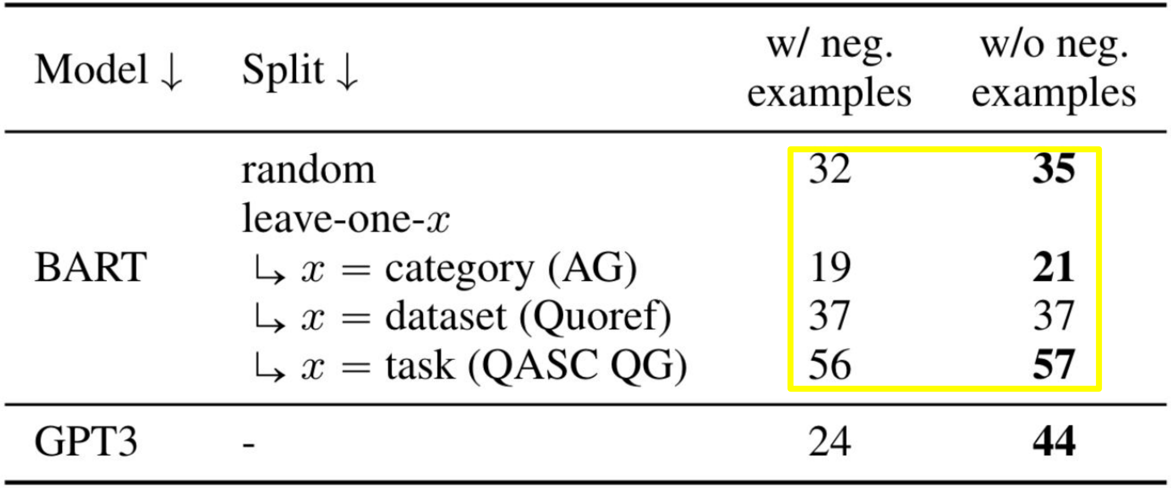
Analysis under different task categories



Category	Helpful Fields	Explanation
Question Generation (QG)	1. Definition 2. Emphasis & Caution 3. Positive Examples 4. Negative Examples	- Provides a holistic picture of the task. - Provides key information for solving the task. - This gives an idea of what is expected in the output. - Good to know the common mistakes people do.
Answer Generation (AG)	1. Prompt 2. Definition 3. Positive Examples	- It limits the exploration space to question spans. - Provides a general understanding of the task. - Reason field is very helpful.
Classification (CF)	1. Definition	- The task is unclear without this field.
Incorrect Answer Generation (IAG)	1. Definition 2. Emphasis & Caution 3. Positive Examples	- Helps understand the utility of such a task. - Source of some useful shortcuts. - Helps in understanding the type of questions asked.
Minimal Text Modification (MM)	1. Things to Avoid	- Provides critical information.
Verification (VF)	1. Definition 2. Things to avoid 3. Positive Examples 4. Negative examples	- Makes the task easy to understand. - Contains useful tips required for this task. - Exemplifies task understanding. - Helps avoid potential mistakes.

Analysis of Negative Examples

Negative examples *harms* model performance, counterintuitive!



Model ↓	Split ↓	w/ neg. examples	w/o neg. examples
BART	random	32	**35**
	leave-one-x		
	↳ x = category (AG)	19	**21**
	↳ x = dataset (Quoref)	37	37
	↳ x = task (QASC QG)	56	**57**
GPT3	-	24	**44**

Analysis of Performance Upper-bound

On average, task-specific models score 66% which is *considerably higher* than our models' best generalization (32%), indicating *a considerable room for improving* generalization-based models.

Huge gap!

Limitations

- Small size of the dataset
  - The paper utilize their proposed dataset NATURAL INSTRUCTION to verify the effectiveness of instructions on the cross-task generalization. However, the dataset size not *large* (61 tasks) and *diverse* (6 categories) enough for the conclusion.
- Suboptimal model performance
  - The proposed model (finetuned BART with full instruction evaluation) enhances cross-task generalization performance, but still possess a *big gap* from the task-specific models.

Conclusion

- Introduce dataset: NATURAL INSTRUCTIONS
  - A dataset of human-authored instructions from existing well-known datasets mapped to a unified schema
- Through finetuning with seen tasks and evaluate on unseen tasks, we find that
  - With designed *instruction*, LM *can generalize* across tasks.
  - There is still a *large headroom for improvement* of cross-task generalization.

SUPER-NATURALINSTRUCTIONS: Generalization via Declarative Instructions on 1600+ NLP Tasks

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Maitreya Patel, Kuntal Kumar Pal, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Shailaja Keyur Sampat, Savan Doshi, Siddhartha Mishra, Sujan Reddy, Sumanta Patro, Tanay Dixit, Xudong Shen, Chitta Baral, Yejin Choi, Noah A. Smith, Hannaneh Hajishirzi, Daniel Khashabi

Motivation

- **Observation**: Cross-task generalization has great progress, but
  - *Role of supervised data* is unexplored due to limited available data.
  - It is *hard to retrain and reproduce* their experiments due to the gigantic models.
- **Goal**: Build a *large-scale benchmark* of a broad range of NLP tasks and their instructions to facilitate developing and evaluating models that can generalize to unseen tasks.

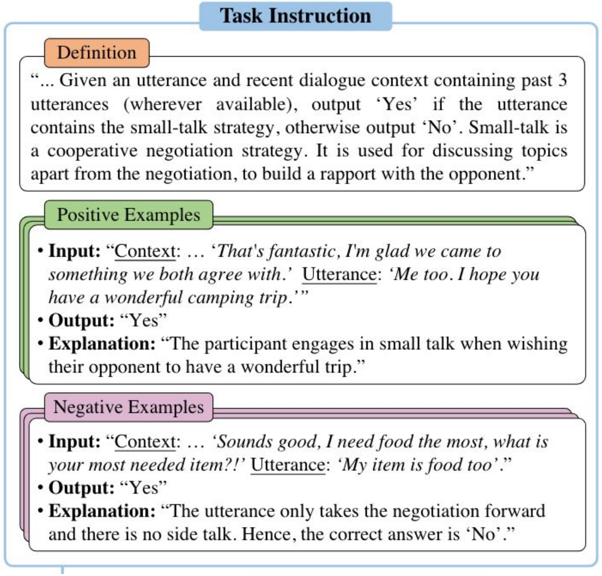
Previous paper also provide same kind of dataset, but much smaller (only 61 tasks with 6 types), this paper propose 1616 tasks with 76 types and many languages.

This paper will…

- **Construct dataset:** a meta-dataset called SUPER-NATURALINSTRUCTIONS with wide variety of NLP tasks.
- **Build a cross-task generalization model:** a generative model Tk-INSTRUCT is proposed that *outperforms much larger models* such as InstructGPT.
- Measure the effectiveness of *data size* and *diversity*, justifying the necessity of the new constructed dataset.

Datasets Preparation: SUPER-NATURALINSTRUCTIONS

- **Schema**: same with the previous paper, but much larger size (see next page).
- e.g.



Task Instruction

Definition

"... Given an utterance and recent dialogue context containing past 3 utterances (wherever available), output 'Yes' if the utterance contains the small-talk strategy, otherwise output 'No'. Small-talk is a cooperative negotiation strategy. It is used for discussing topics apart from the negotiation, to build a rapport with the opponent."

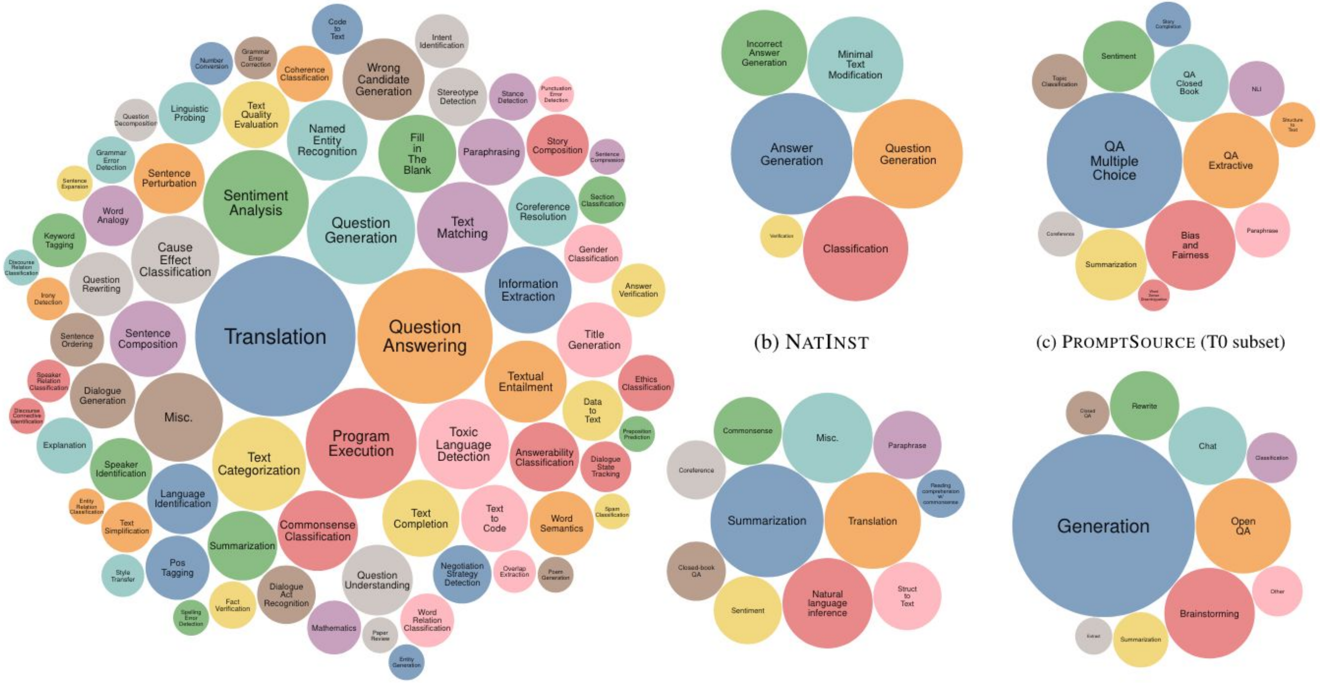
Positive Examples

- **Input:** "Context: ... *'That's fantastic, I'm glad we came to something we both agree with.'* Utterance: *'Me too. I hope you have a wonderful camping trip.'*"
- **Output:** "Yes"
- **Explanation:** "The participant engages in small talk when wishing their opponent to have a wonderful trip."

Negative Examples

- **Input:** "Context: ... *'Sounds good, I need food the most, what is your most needed item?!'* Utterance: *'My item is food too'*."
- **Output:** "Yes"
- **Explanation:** "The utterance only takes the negotiation forward and there is no side talk. Hence, the correct answer is 'No'."

Datasets Preparation: SUPER-NATURALINSTRUCTIONS



(b) NATINST

(c) PROMPTSOURCE (T0 subset)

(a) SUP-NATINST (this work)

(d) FLAN

(e) INSTRUCTGPT

Datasets Preparation: SUPER-NATURALINSTRUCTIONS

- **Data sources (community effort on GitHub)**:
  - *Public NLP datasets.*
  - *Intermediate annotations* in crowdsourcing experiments (e.g., paraphrasing questions or rating their quality during crowdsourcing a QA dataset)
  - *Synthetic tasks* that can be communicated to an average human in a few sentences (e.g., basic algebraic operations like number comparison, finding the longest palindrome substring, etc.)
- *88* contributors

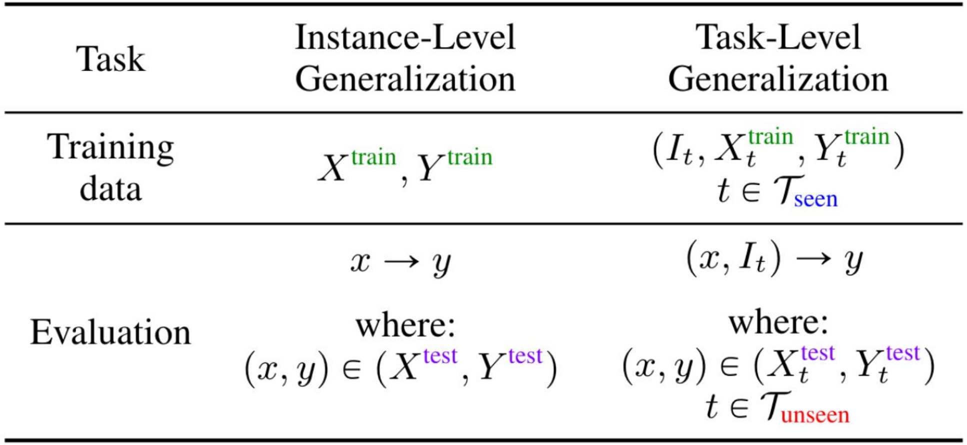
Problem Definition

Same as the previous paper.

Definition of Cross-task models

Cross-task models: learn a model M that *at inference* obtains the output y given the *input x* and the *task instruction I_t*:

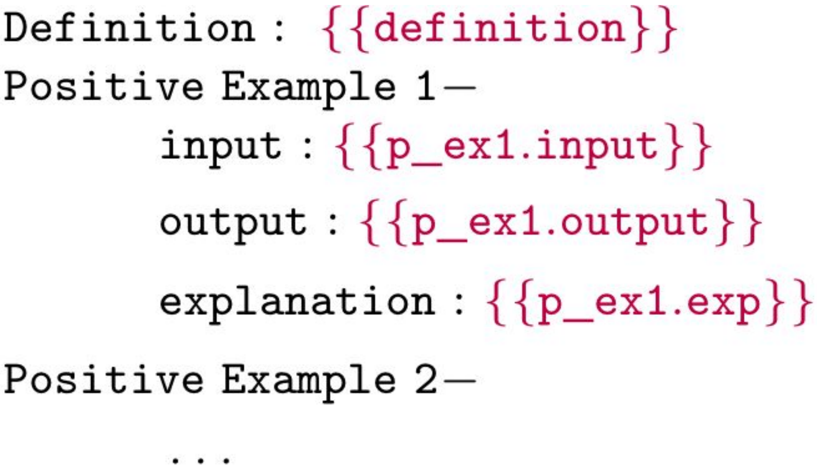
$$M(x) = y \text{ for } (x, y) \in (X_t^{\text{train}}, Y_t^{\text{train}})$$



Task	Instance-Level Generalization	Task-Level Generalization
Training data	$X^{\text{train}}, Y^{\text{train}}$	$(I_t, X_t^{\text{train}}, Y_t^{\text{train}})$ $t \in \mathcal{T}_{\text{seen}}$
Evaluation	$x \rightarrow y$ where: $(x, y) \in (X^{\text{test}}, Y^{\text{test}})$	$(x, I_t) \rightarrow y$ where: $(x, y) \in (X_t^{\text{test}}, Y_t^{\text{test}})$ $t \in \mathcal{T}_{\text{unseen}}$

T*k*-INSTRUCT: Learning to Follow Instructions at Scale

- Acquired by finetune T5 with instruction composed of a task definition and two positive examples.



```
Definition : {{definition}}
Positive Example 1—
        input : {{p_ex1.input}}
        output : {{p_ex1.output}}
        explanation : {{p_ex1.exp}}
Positive Example 2—
        ...
```

T*k*-INSTRUCT: Learning to Follow Instructions at Scale

- Acquired by finetune T5 with instruction composed of a task definition and two positive examples.
- For multilingual variant, mT*k*-INSTRUCT is finetuned based on mT5 model.

Benchmarking Cross-Task Generalization with SUP-NATINST

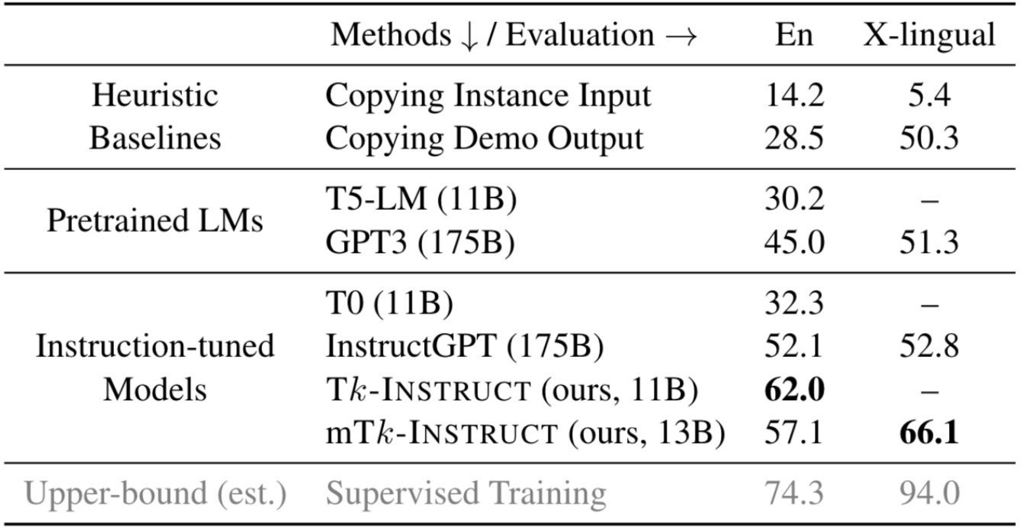
- **Evaluation setup:**
  - **Evaluation split of unseen tasks:** we sample a maximum of 100 instances for each task, which results in 15,310 testing instances in total. The remaining tasks are used for training models.
  - **Two evaluation tracks:** *English cross-task generalization* (119 tasks) and *cross lingual cross-task generalization* (35 tasks).
- **Evaluation metric**: ROUGE-L

Benchmarking Cross-Task Generalization with SUP-NATINST

- **Baselines:**
    - **Heuristic:**
        - **Copying Demo Output:** copies the output of a random demonstration example.
        - **Copying Instance Input:** copies the given instance input.
    - **Off-the-shelf pre-trained language models:**
        - **T5 (11B).**
        - **GPT-3 (175B).**
    - **Instruction tuned models:**
        - **InstructGPT:** acquired by using RLHF to GPT-3.
        - **T0:** T5 finetuned with a collection of task prompts in PROMT-SOURCE.
    - **Upper bound estimates:** fine-tuning an oracle model on the tasks' labeled instances.

Experiments

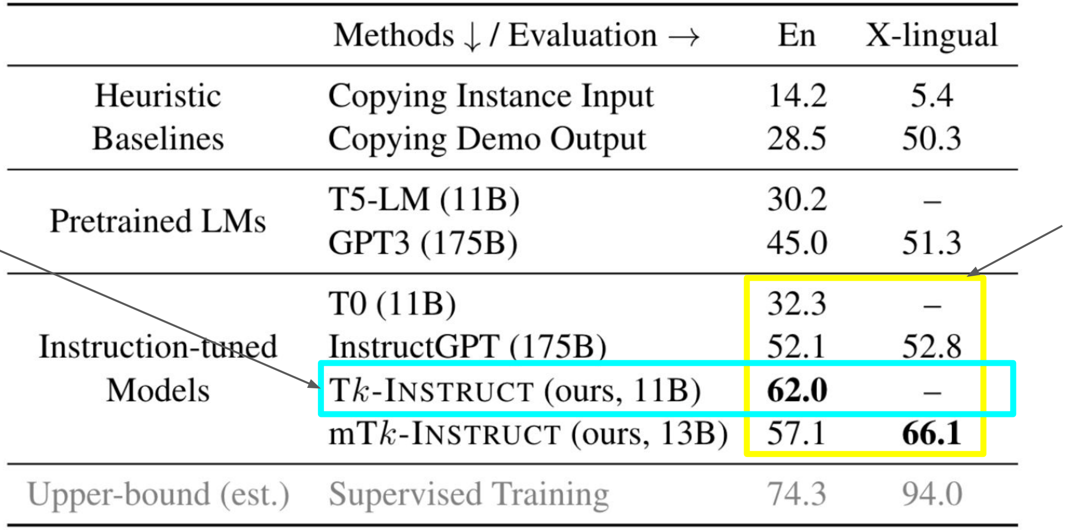
The overall performance of different methods on unseen tasks:



	Methods ↓ / Evaluation →	En	X-lingual
Heuristic Baselines	Copying Instance Input	14.2	5.4
	Copying Demo Output	28.5	50.3
Pretrained LMs	T5-LM (11B)	30.2	–
	GPT3 (175B)	45.0	51.3
Instruction-tuned Models	T0 (11B)	32.3	–
	InstructGPT (175B)	52.1	52.8
	T*k*-INSTRUCT (ours, 11B)	**62.0**	–
	mT*k*-INSTRUCT (ours, 13B)	57.1	**66.1**
Upper-bound (est.)	Supervised Training	74.3	94.0

Experiments

Our model that is fine-tuned on SUP-NATINST outperforms InstructGPT and T0 by a large margin.

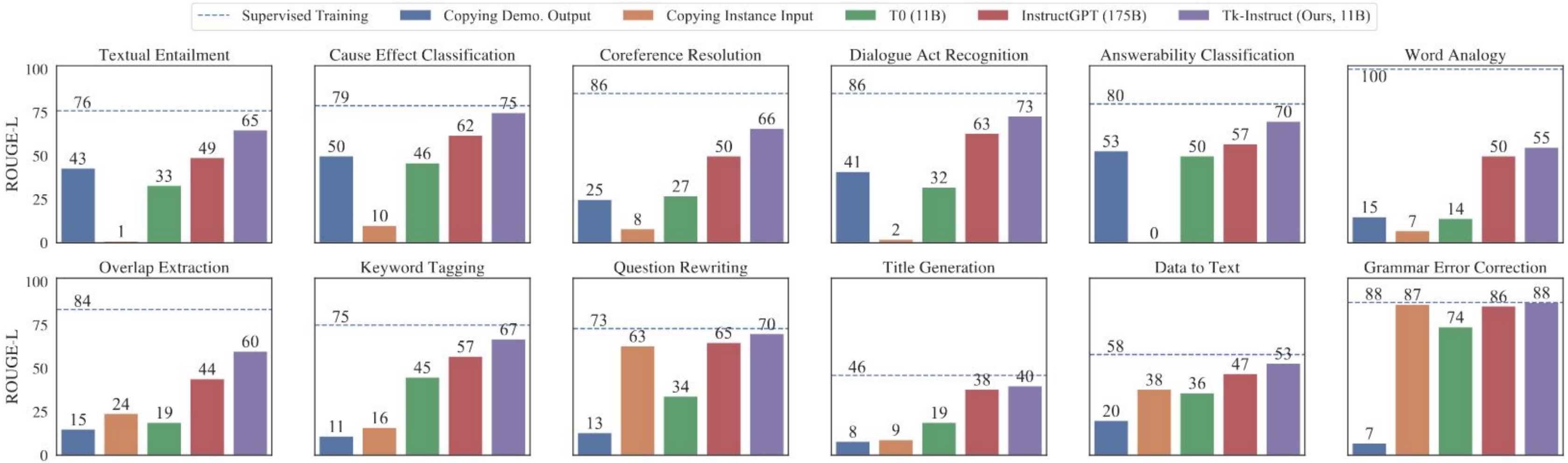


	Methods ↓ / Evaluation →	En	X-lingual
Heuristic Baselines	Copying Instance Input	14.2	5.4
	Copying Demo Output	28.5	50.3
Pretrained LMs	T5-LM (11B)	30.2	–
	GPT3 (175B)	45.0	51.3
Instruction-tuned Models	T0 (11B)	32.3	–
	InstructGPT (175B)	52.1	52.8
	Tk-INSTRUCT (ours, 11B)	**62.0**	–
	mTk-INSTRUCT (ours, 13B)	57.1	**66.1**
Upper-bound (est.)	Supervised Training	74.3	94.0

Models that leverage instructions show stronger generalization to unseen tasks.

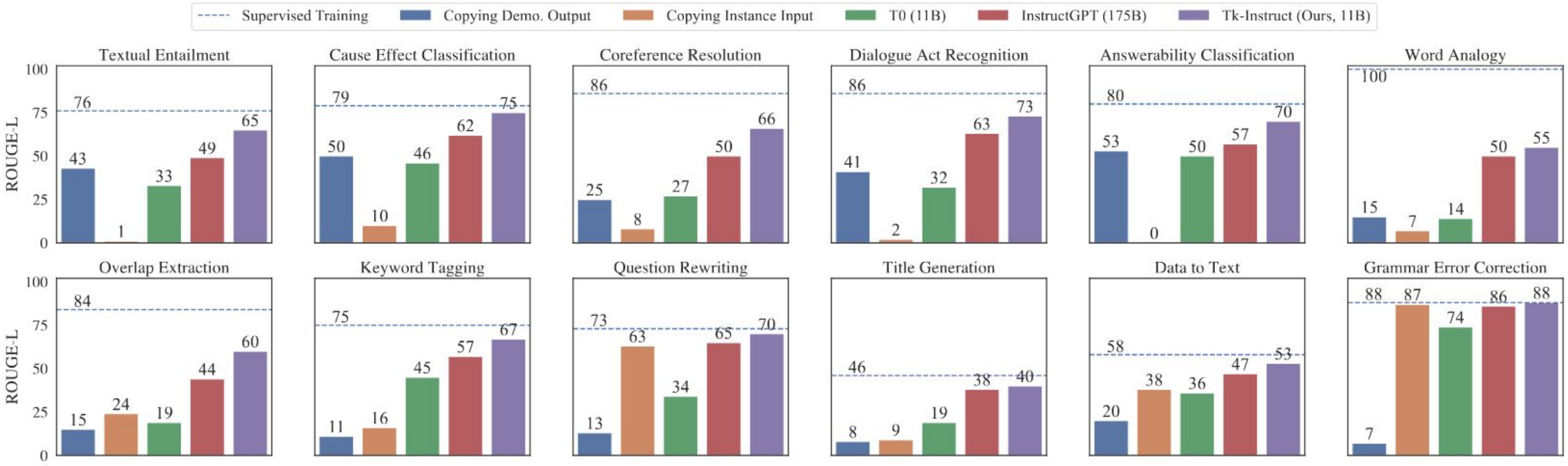
Experiments

Performance according to categories:



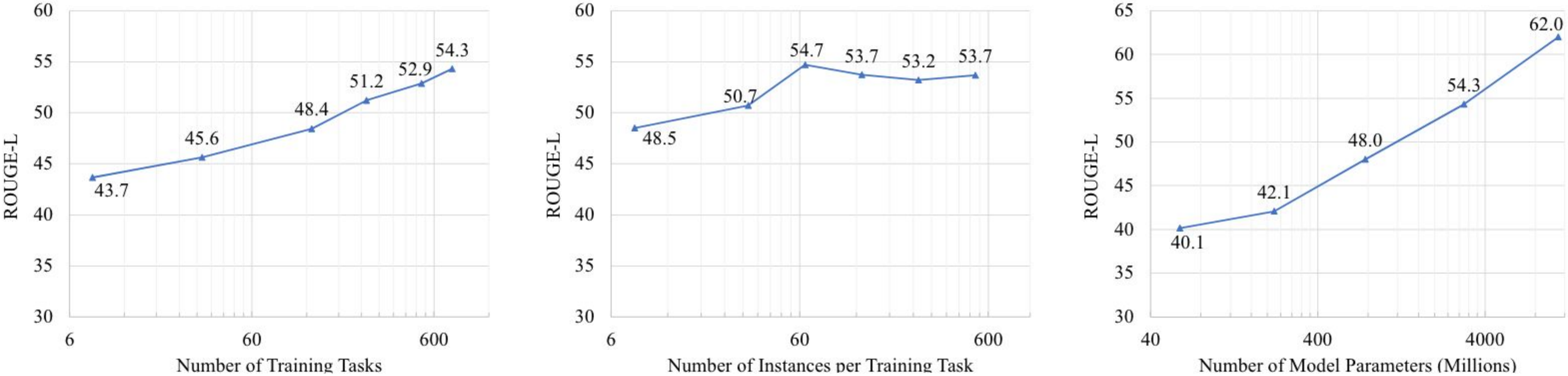
Experiments

Tk-INSTRUCT *consistently performs best* across baselines on all task types, while there is *still a sizable gap* compared to supervised training.



Experiments

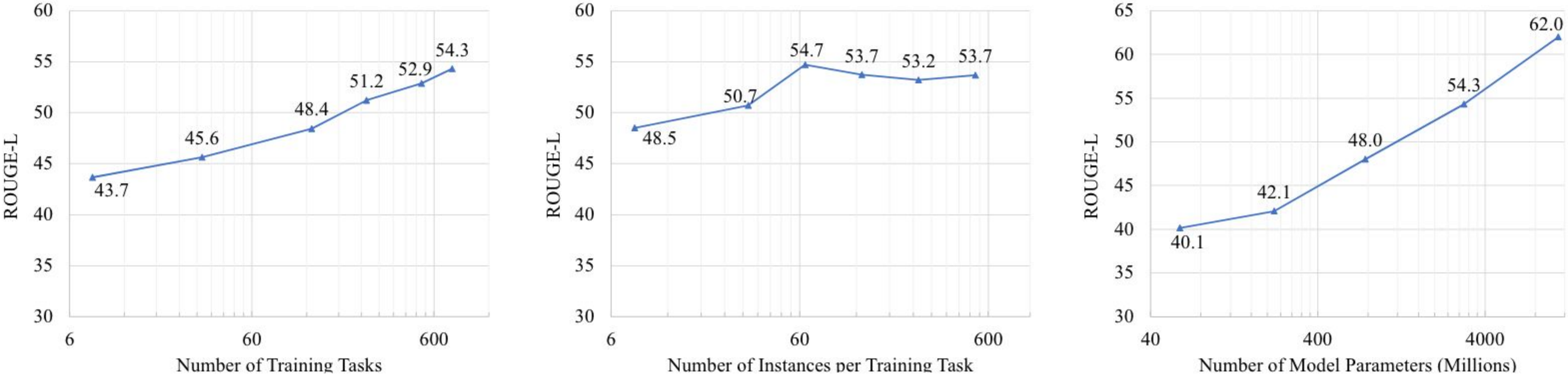
Cross-task generalization performance w.r.t. # train tasks, # instances per task, and model size.



(a) (b) (c)

Experiments

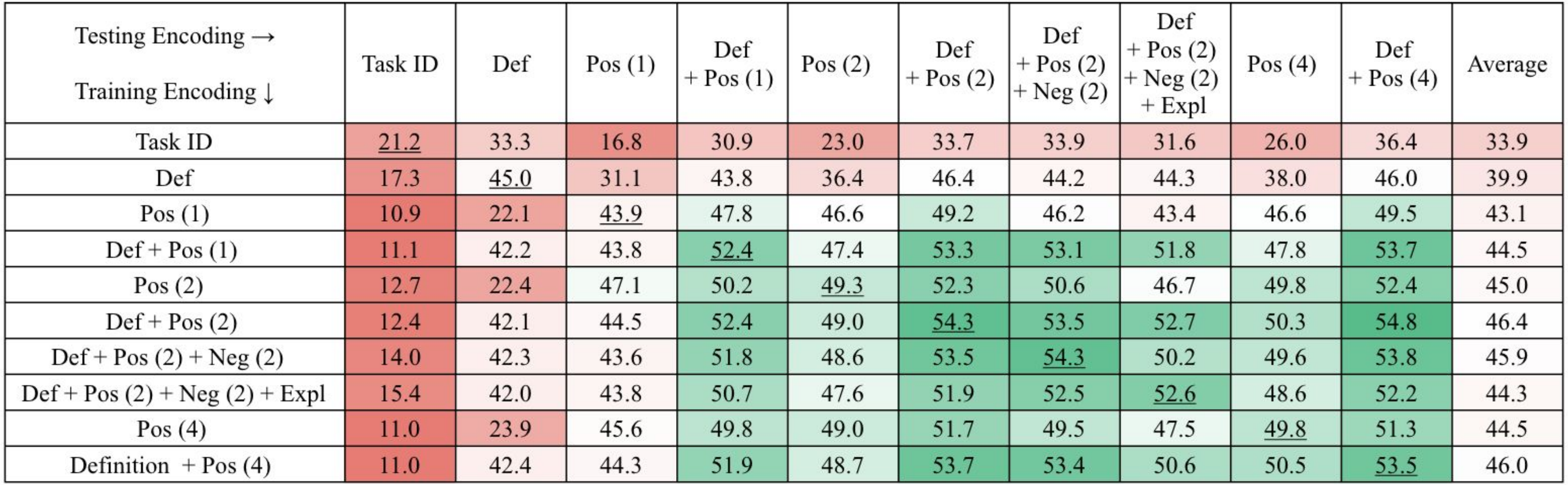
linear growth of model performance with exponential increase in observed tasks and model size. Evidently, the *performance gain* from more instances is *limited*.



(a) (b) (c)

Experiments

Performance (ROUGE-L) of models trained and evaluated with various encodings.



Testing Encoding → Training Encoding ↓	Task ID	Def	Pos (1)	Def + Pos (1)	Pos (2)	Def + Pos (2)	Def + Pos (2) + Neg (2)	Def + Pos (2) + Neg (2) + Expl	Pos (4)	Def + Pos (4)	Average
Task ID	21.2	33.3	16.8	30.9	23.0	33.7	33.9	31.6	26.0	36.4	33.9
Def	17.3	45.0	31.1	43.8	36.4	46.4	44.2	44.3	38.0	46.0	39.9
Pos (1)	10.9	22.1	43.9	47.8	46.6	49.2	46.2	43.4	46.6	49.5	43.1
Def + Pos (1)	11.1	42.2	43.8	52.4	47.4	53.3	53.1	51.8	47.8	53.7	44.5
Pos (2)	12.7	22.4	47.1	50.2	49.3	52.3	50.6	46.7	49.8	52.4	45.0
Def + Pos (2)	12.4	42.1	44.5	52.4	49.0	54.3	53.5	52.7	50.3	54.8	46.4
Def + Pos (2) + Neg (2)	14.0	42.3	43.6	51.8	48.6	53.5	54.3	50.2	49.6	53.8	45.9
Def + Pos (2) + Neg (2) + Expl	15.4	42.0	43.8	50.7	47.6	51.9	52.5	52.6	48.6	52.2	44.3
Pos (4)	11.0	23.9	45.6	49.8	49.0	51.7	49.5	47.5	49.8	51.3	44.5
Definition + Pos (4)	11.0	42.4	44.3	51.9	48.7	53.7	53.4	50.6	50.5	53.5	46.0

Experiments

- Performance (ROUGE-L) of models trained and evaluated with various encodings.
- Encoding methods:
  - **Task ID** is a short string composed of dataset name and task category.
  - **Def** represents the task definition.
  - **Pos (k)** represents k positive examples.
  - **Neg (k)** represents k negative examples.
  - **Expl** represents explanation.
- Observations:
  - Various evaluate instructional elements leads to different generalization performance.
  - Model performance are relatively stable w.r.t. the change of test encoding method.

Limitation

- The *categories* and *language diversity* in the proposed dataset are skewed.
- Collected tasks are skewed to *short responses*; This *under-representation* of the longtail of tasks poses a challenge for building general-purpose models in the future.
- For some tasks, ROUGE-L is *not an effective quality measure* (such as rewriting tasks where copying the input gives high score).

Final Summary

- *Finetuned Language Models are Zero-Shot Learners*
  - Instruction tuning as a form of fine-tuning
  - The FLAN LLM model, finetuned with instruction tuning, learns zero-shot learning effectively
  - Performance improvement is observed on larger models (>100B parameters)
- *Multitask Prompted Training Enables Zero Shot Task Generalization*
  - Created Public Pool of Prompts (P3) through public effort
  - Testing model generalization ability using multiple unseen tasks for evaluation

Final Summary (Cont.)

- *Cross-Task Generalization via Natural Language Crowdsourcing Instructions*
    - Defining natural instructions before compiling task instances through crowdsourcing
    - Analyzed the effects of prompt variations on model performance
- *Super-Natural Instructions: Generalization via Declarative Instructions on +1600 NLP Tasks*
    - Building a large-scale diverse instruction dataset with 1600+ tasks.
    - Exploring multi-language cross-task generalization.

Main Ideas

- Multitask instruction/prompt tuning improves task generalization.
- Diverse prompts is an alternative to model scaling.
- Further developments in prompt engineering and task evaluation are the next steps for developing general LLMs.

Thanks for listening!