universität wien

# Scaling Instruction-Finetuned Language Models

Andreas Stephan, 15.12

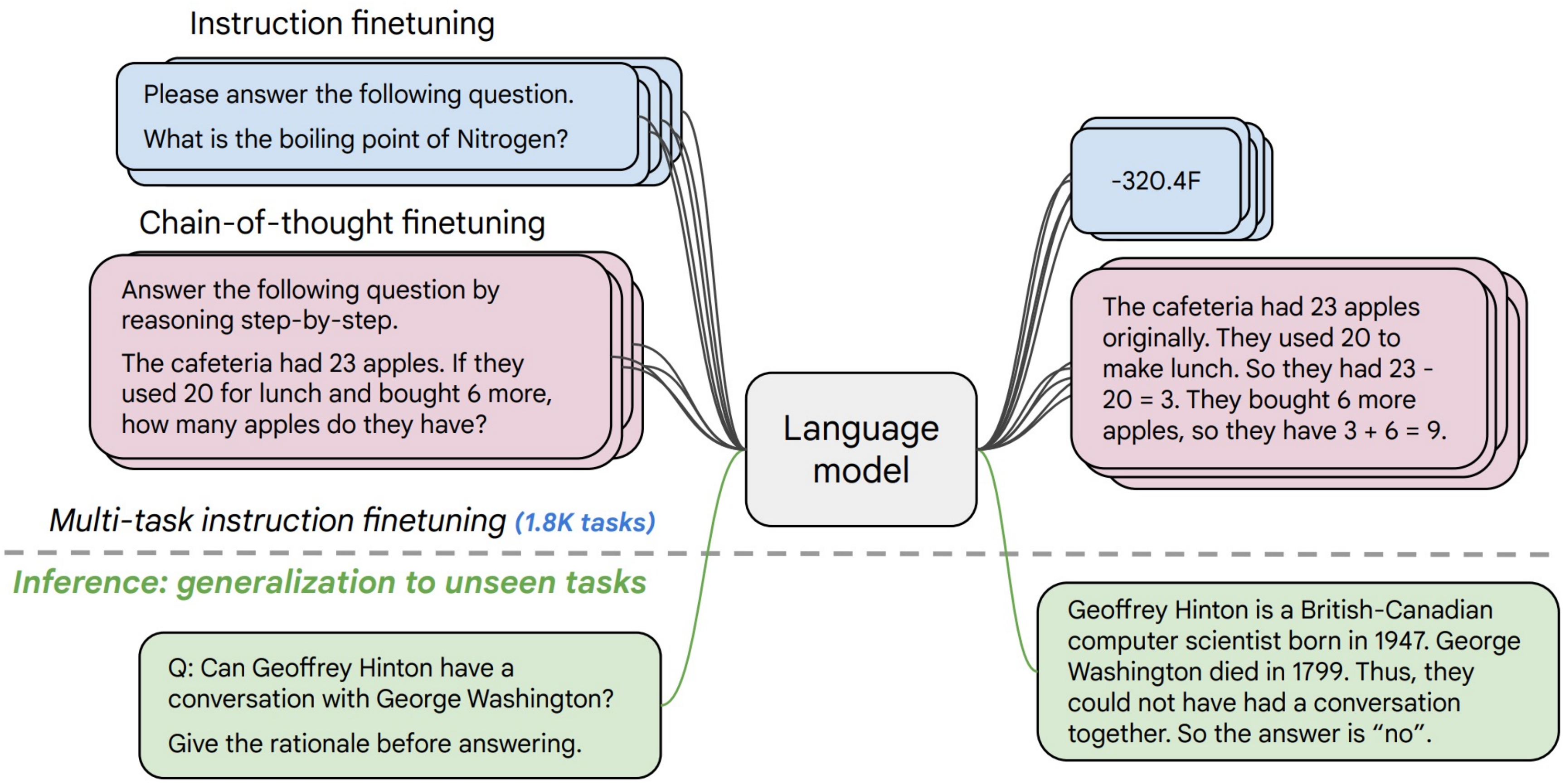
# Ideas behind big LM’s

- Use transformers
  - Decoder-only models (Only look at text to the left)
  - Encoder-Decoder models, e.g. translation: Encoder sees all text (german), Decoder sees german and translated text to left
- Language Modelling (LM): P(X_n | X_1, …, X_{n-1})
- Build “world model” which does not need special training for new tasks
- Zero/Few-Shot / “in-context” learning: Provide task description and zero or few examples for each prediction, but don’t train for anything specifically

# Scaling Instruction-Finetuned Language Models

Finetuning language models on a collection of datasets phrased as instructions has been shown to improve model performance and generalization to unseen tasks. In this paper we explore instruction finetuning with a particular focus on (1) scaling the number of tasks, (2) scaling the model size, and (3) finetuning on chain-of-thought data. We find that instruction finetuning with the above aspects dramatically improves performance on a variety of model classes (PaLM, T5, U-PaLM), prompting setups (zero-shot, few-shot, CoT), and evaluation benchmarks (MMLU, BBH, TyDiQA, MGSM, open-ended generation, RealToxicityPrompts). For instance, Flan-PaLM 540B instruction-finetuned on 1.8K tasks outperforms PaLM 540B by a large margin (+9.4% on average). Flan-PaLM 540B achieves state-of-the-art performance on several benchmarks, such as 75.2% on five-shot MMLU. We also publicly release Flan-T5 checkpoints,[1] which achieve strong few-shot performance even compared to much larger models, such as PaLM 62B. Overall, instruction finetuning is a general method for improving the performance and usability of pretrained language models.

# Instruction finetuning and CoT



# Self-consistency

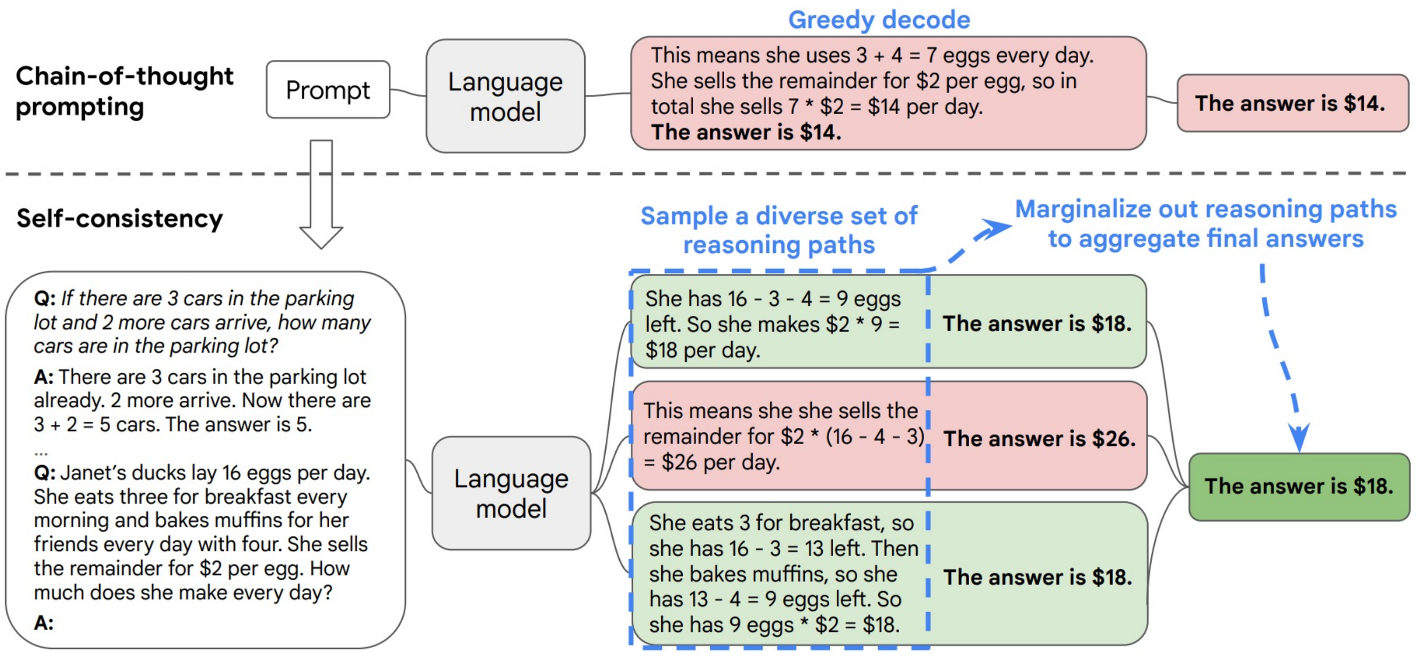



Figure 1: The self-consistency method contains three steps: (1) prompt a language model using chain-of-thought (CoT) prompting; (2) replace the "greedy decode" in CoT prompting by sampling from the language model's decoder to generate a diverse set of reasoning paths; and (3) marginalize out the reasoning paths and aggregate by choosing the most consistent answer in the final answer set.

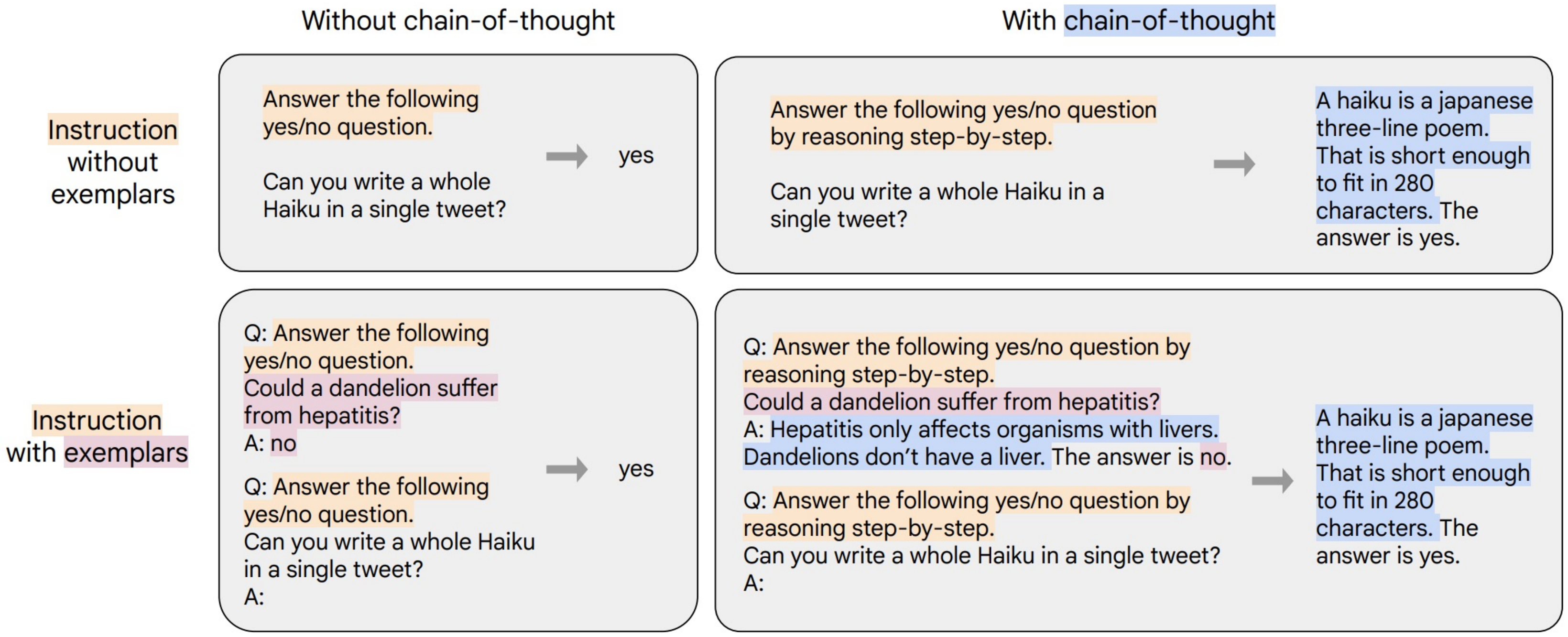



Figure 3: Combinations of finetuning data formats in this work. We finetune with and without exemplars, and also with and without chain-of-thought. In addition, we have some data formats without instructions but with few-shot exemplars only, like in Min et al. (2022) (not shown in the figure). Note that only nine chain-of-thought (CoT) datasets use the CoT formats.

# Diversify tasks

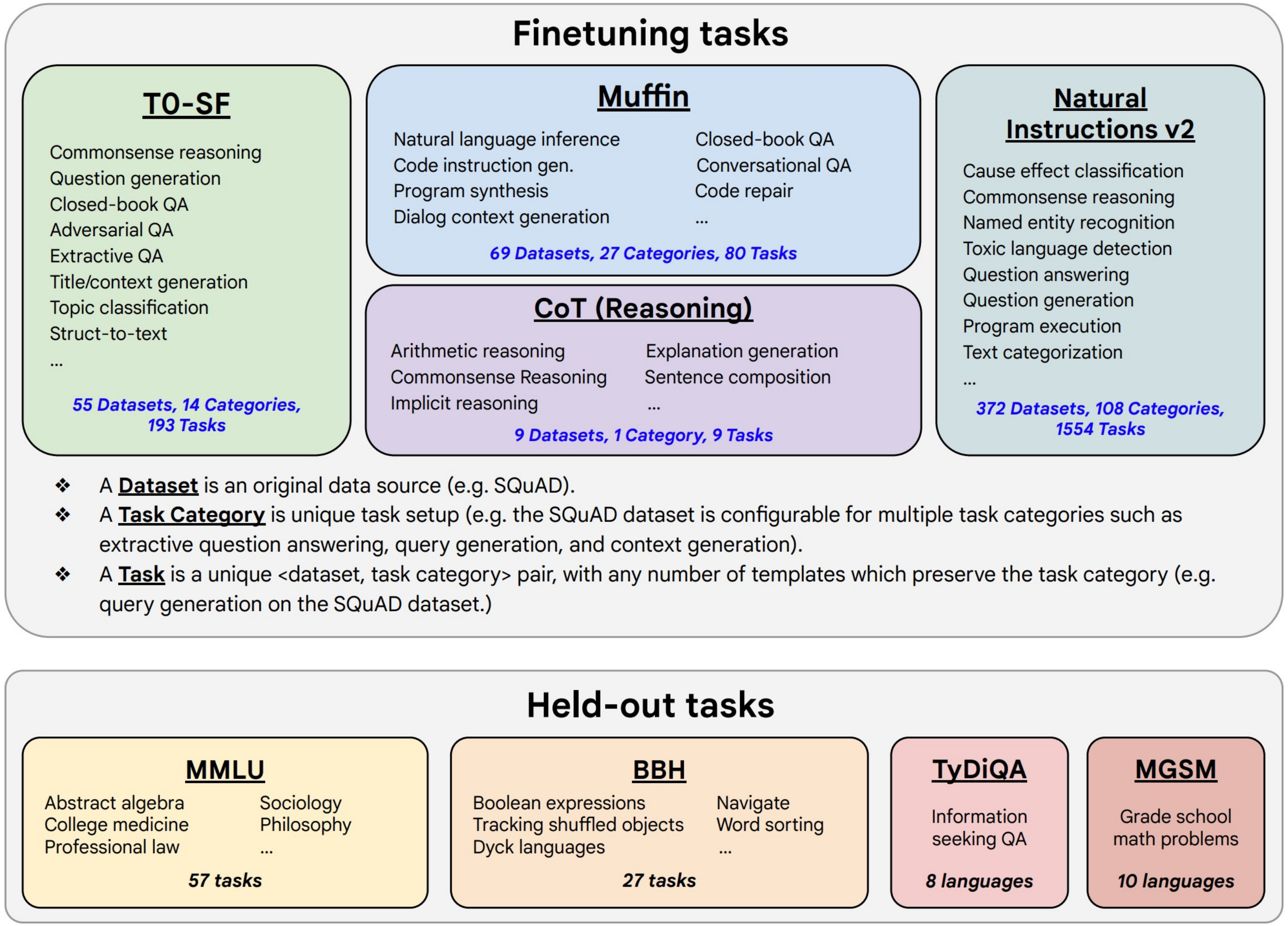



Figure 2: Our finetuning data comprises 473 datasets, 146 task categories, and 1,836 total tasks. Details for the tasks used in this paper is given in Appendix F.

-> **Numbers in the paper**

# Related

- U-Palm: https://arxiv.org/pdf/2210.11399.pdf
- Emergent Abilities: “Things which start working with larger models”
  - https://arxiv.org/abs/2206.07682
- Inverse Scaling: “Things that stop working with larger models”
  - https://github.com/inverse-scaling/prize