Concept-aware Data Construction Improves In-context Learning of Language Models

Michal Štefánik[♣] **Marek Kadlčík**[♣] **Petr Sojka**[♣]

[♣]**Faculty of Informatics**
Masaryk University, Czech Republic

Abstract

Many recent language models (LMs) of the Transformers family are capable of in-context learning (ICL), manifested in the LMs' ability to perform a new task solely from its description in a natural language input. Previous work curating these models assumes that ICL emerges from vast over-parametrization or the scale of multi-task training, but recent theoretical work attributes ICL emergence to training data properties, creating in-context learners with small, synthetic data.

Inspired by these findings, we propose Concept-aware Training (CoAT), a framework for constructing training exemplars that make it beneficial for the LM to learn to utilize the **analogical reasoning concepts** from demonstrations. We find that by using CoAT, pre-trained transformers *can* learn to better utilise new latent concepts from demonstrations and that such ability makes ICL more robust to previously uncovered functional deficiencies. Finally, we show that concept-aware in-context learning improves ICL performance on a majority of new tasks compared to traditional instruction tuning, reaching performance comparable to the multitask learners using magnitudes of more training data.

1 Introduction

The in-context learning (ICL), as initially uncovered by Brown et al. (2020), is a setting requiring language models (LMs) to infer and apply correct functional relationships from the pairs of inputs and outputs (i.e. *demonstrations*) presented solely in user input prompt (Li et al., 2023a). Given that a small set of demonstrations can be obtained for any machine learning task, in-context learning presents a much more versatile and practical alternative to task-specific models.

Modern in-context learners can often perform ICL with quality comparable to task-specialized models (Zhao et al., 2023; Štefánik et al., 2023). However, it remains unclear why some LMs are able of ICL in such quality while others are not; Initial work introducing GPT3 (Brown et al., 2020) followed by (Thoppilan et al. (2022); Chowdhery et al. (2022); *inter alia*) explains ICL as an emergent consequence of models' scale. But more recent LMs (Sanh et al., 2022; Wang et al., 2022; Wei et al., 2021; Ouyang et al., 2022) are based on 100-times smaller models and reach comparable ICL quality, instead attributing the ICL emergence to a vast volume and diversity of training instructions.

Other work identifies covariates of the emergence of ICL in **data irregularities**: training cases that can *not* be explained by mere statistical co-occurrence of tokens (Chan et al. (2022); Hahn & Goyal (2023); Appendix A). Notably, Xie et al. (2022) identify the key property in the occurrence of text dependencies that must be resolved by identifying *latent concepts* that underpin these dependencies.

Inspired by these findings, we propose and implement a data construction framework that *encourages* the occurrence of such concept-dependent irregularity in training samples, and hence, *requires* models to learn to utilise latent concepts that explain these irregularities. We show that language models are able to generalize concept-learning ability to unseen concepts, and on extrinsic evaluation over 70 diverse tasks, we show that in-context learning based on concepts has the potential to enhance training data efficiency, robustness, and performance of in-context learners.

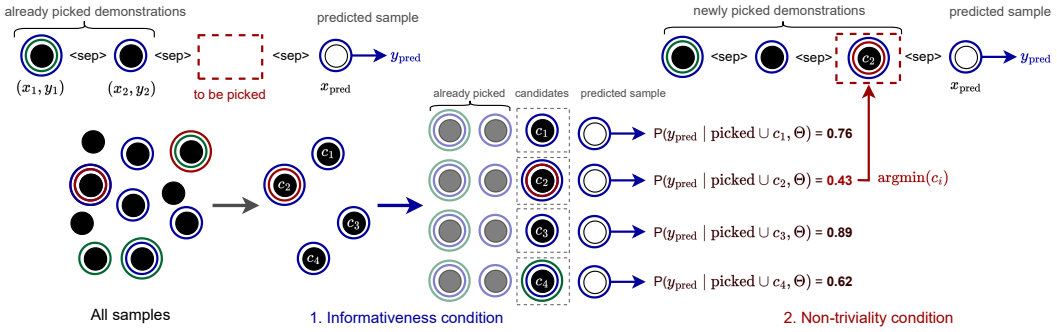

Figure 1: **Demonstrations selection of Concept-aware training (CoAT):** From all samples of the training dataset, we first (i) filter out available samples to ones *sharing* a reasoning concept ◯ with predicted sample $(x_{\text{pred}}, y_{\text{pred}})$. From this subset, we (ii) incrementally pick the next demonstration, i.e. candidate sample c_i such that the model Θ's probability of generating the correct prediction y_{pred} if we pick c_i among demonstrations is *minimal*.

2 Concept-Aware Training

Aiming to create language models able to learn a new latent reasoning concept in-context, we propose a **Concept-Aware Training** (CoAT): an instruction-tuning framework specifying **conditions for a selection of few-shot demonstrations** in the training instructions (Figure 1). We assume the format of training prompts widely used in the previous work training in-context few-shot learners in multitask setting, constructing training instructions from k demonstrations composed of the input texts x with labels y followed by the predicted sample's input text x_{pred}:

$$[x_1, y_1, \langle sep \rangle, \ldots, x_k, y_k, \langle sep \rangle, x_{\text{pred}}] \rightarrow y_{\text{pred}}$$

In this setting, CoAT proposes to filter demonstrations sequentially by two conditions. The main condition, denoted as **informativeness condition**, assures to pick demonstrations exhibiting a specific *reasoning concept* C that is *shared* between each picked demonstration (x_i, y_i) and the predicted example $(x_{\text{pred}}, y_{\text{pred}})$. This makes it beneficial for the model to learn to *extract* and *apply* informative concepts of demonstrations. However, as the sole *informativeness* condition may easily pick demonstrations very similar to the predicted sample, we propose applying a second condition: **Non-triviality condition** chooses from the informative demonstrations the ones with which it is 'difficult' for the model to respond correctly. This also increases the heterogeneity of different co-occurring concepts, avoiding the over-reliance on a small set of specific concepts in a small-data regime.

We implement[1] CoAT in two training stages: First, we train LM on a synthetic QA dataset with annotations of *reasoning chains* as concepts. Second, we restore the LM's ability to work with *natural* language by further training on a standard QA dataset.

Informativeness condition We find a large collection of annotated reasoning concepts in a TeaBReaC dataset of Trivedi et al. (2022), containing more than 900 unique explanations over a relatively large set of *synthetic* QA contexts. Each TeaBReAC's explanation maps a natural question to the answer span through a sequence of declarative *reasoning steps*, such as "select→group→project". Within CoAT, we use these explanations as the shared concepts C; Hence, in the training exemplars, all demonstrations apply the same reasoning chain as the predicted sample.

To restore the model's ability to work with a natural language, in the second step, we fit the resulting model to *natural* inputs by further fine-tuning on AdversarialQA dataset (Bartolo et al., 2021); As the annotations of reasoning concepts in general QA datasets are scarce, in this case, we naively use the initial word of the question ("Who", "Where", ...) as the shared concept, aware that such-grouped samples are not always mutually informative.

Non-triviality condition In both training stages, we implement the *non-triviality condition* in the following steps. (i) We select a random *subset* of 20 samples that passed the *informativeness* condition (denoted X_{info}). (ii) From X_{info}, we iteratively *pick* a sequence of $i \in 1..k$ demonstrations (with a randomly-chosen *number of demonstrations* $k : 2 \leq k \leq 8$) as follows:

[1]CoAT implementation and our trained models are all available on: `https://github.com/MIR-MU/CoAT`.

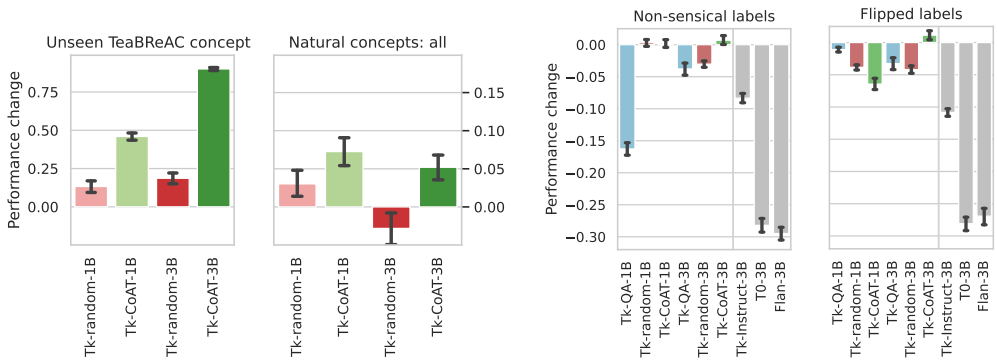

Figure 2: Relative performance change when (Left) a model is presented with demonstrations utilising analogical reasoning concept: a) unseen reasoning chains, or b) selected natural concepts (RQ1); (Right) a model is presented with semantically-unrelated labels in demonstrations (RQ2).

1. For each sample $(x_j, y_j) \in X_{\text{info}}$, we compute a *probability of generating the correct prediction* y_{pred} if a given sample is included among demonstrations. When y_{pred} contains more than one token, we compute such probability as the average of the likelihoods of all y_{pred}'s tokens in the teacher-forced generation.
2. In each step i, we include among the demonstrations a sample (x_j, y_j) with which the probability of generating correct prediction is *minimal*.

An overview of this process is depicted in Fig. 1. Concrete training prompts are displayed in Table 3.

3 Experiments

Our experiments provide empirical evidence towards answering three research questions (**RQs**):

1. **Can we improve models' ability to *benefit* from new reasoning concepts in-context?**
2. **Can concept-aware in-context learners learn functional relations more robustly?**
3. **Can concept-aware in-context learning improve performance in real-world tasks?**

To maximise comparability with the previous work, we fine-tune our models from T5 pre-trained models of Xue et al. (2021). In both training stages (Sec. 2), we fine-tune all model parameters in a teacher-forced next-token prediction (sequence-to-sequence objective) until convergence of evaluation loss. We construct the evaluation exemplars from $k = 3$ randomly but consistently chosen demonstrations consisting of self-containing prompts, with options including expected labels. We further detail the parameters of our training in Appendix B and of our evaluations in Appendix C.

Baselines We evaluate the impact of each CoAT's data construction step against two baselines: **(1) Tk-random** trained identically to CoAT models but picking the in-context demonstrations *randomly* with uniform probability over the whole training set, reproducing the methodology of a majority of previous work on instruction tuning (incl. Tk-Instruct and Flan). **(2) Tk-info** constructing training prompts from demonstrations passing *only* the *informativeness* condition; Such-picked demonstrations can be similar or identical to the predicted sample, making it trivial to learn a correct prediction.

Other evaluated models We also evaluate three recent in-context learners for which we can assess what model and datasets were used in their training mix: (1) **T0** (Sanh et al., 2022) trained on a mixture of 35 datasets of different tasks; (2) **Tk-Instruct** (Wang et al., 2022) trained in a few-shot format (Tk-random), over 1,616 tasks, and (3) **Flan** of (Chung et al., 2022) further extending the dataset of Tk-Instruct to a total of 1,836 tasks, including 9 tasks with chain-of-thought labels.

4 Results

RQ1: Ability to benefit from new concepts Following Štefánik & Kadlčík (2023), we assess models' ability to *benefit* from new reasoning concepts in evaluation with demonstrations that are

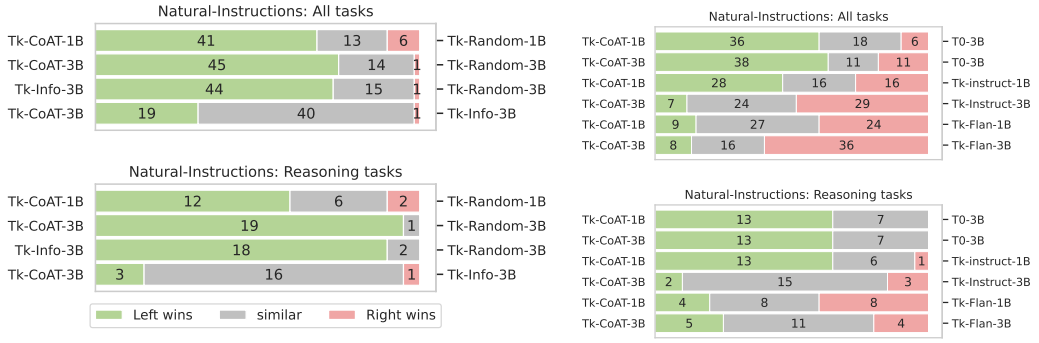

Figure 3: Win rate of CoAT models on Natural Instructions tasks (Left) against the baselines with data construction of previous work (§3), and (Right) against previous models trained on mixtures of 35–1,836 tasks. "Similar" denote tasks where the models' performance does not differ significantly.

guaranteed to *share* a specific concept with the predicted sample. Afterwards, we quantify models' gain from using the demonstrated concept by computing the *difference* in performance between such concept-sharing evaluation and *randomly* chosen demonstrations. In this framework, we perform two analyses: (1) on TeaBReAC's samples with previously *unseen* reasoning chains, and (2) on four natural-language datasets annotating reasoning concepts within their *explanations* adopted from Štefánik & Kadlčík (2023) (overviewed in §C) and compare CoAT models to a Tk-random baselines.

Figure 2 (left) shows that both CoAT and random-demonstration models can improve from analogical reasoning chains presented in TeaBReAC's demonstrations. However, the improvement of CoAT-trained models is two times and four times larger than of Tk-random with the smaller and larger model, respectively. Evaluation of improvements on selected *natural* concepts (Figure 2; right) shows that concept-learning ability obtained with synthetic data also *transfers* to natural language, as the CoAT models can benefit from concepts significantly *more* than Tk-random.

RQ2: Robustness to semantic distractions Previous work reports functional deficiencies of recent in-context learners, including insensitivity to the demonstrations' labels (Min et al., 2022b). Wei et al. (2023) attribute this to models' over-reliance on pre-trained *semantic priors*, i.e. tokens' meaning. While such property is desirable to a certain extent, over-reliance can hinder the learning of *functional* relations necessary for robust in-context learning of truly unseen tasks.

Following Wei et al. (2023), we assess models' reliance on *labels*' semantics over 8 SuperGLUE tasks as a *difference* in performance between standard few-shot evaluation (§3) and an evaluation with one of these modifications; (i) *Changing* the labels to tokens with *irrelevant* meaning, such as 'Foo', 'Bar', etc. (ii) *Shuffling* the labels so that demonstrations exhibit *semantically incorrect* labels, but the input-label mapping remains consistent. Note that in both settings, the task's *functional* relation can still be learnt from demonstrations. We evaluate three model types: (a) CoAT-trained models, (b) models with uncontrolled data construction (Tk-random & previous work), and (c) models with uncontrolled data construction, but fine-tuned *only* on a *natural* QA dataset (Tk-QA).

Figure 2 (right) shows the results. Tk-QA's evaluations show that pre-training with synthetic dataset itself mitigates over-reliance on semantics, but a comparison of Tk-random and Tk-CoAT suggests that Tk-CoAT's robustness to semantic distractions is a composition of *both* synthetic data *and* CoAT's data construction. Further, we note that multitask learners experience substantially larger decay in performance; This could be a *bias* of massive multi-task learning, where label semantics can *explain* a large portion of training data. This result is consistent with Wei et al. (2023), but contrary, we show that ICL robust to semantic distractions does *not* emerge *exclusively* with ($\geq$ 100B) scale.

RQ3: Practical efficiency of concept-aware in-context learners Finally, we assess whether the concept-based ICL ability obtained within CoAT (Sec. 2) also helps in models' ability to in-context learn new tasks, as exhibited by models' performance on a collection of unseen tasks, without any concept annotations. We primarily compare the results of CoAT models to Tk-random, where we can ensure that all training settings, except the data construction, are identical. As an ablation, we also compare to Tk-info (without *Non-triviality* condition; §3). We evaluate models on two task collections: (i) on 60 extractive tasks of the test split of Natural Instructions (Wang et al., 2022), and (ii) on all SuperGLUE tasks (Wang et al., 2019) (in Appendix C).

Figure 3 (left) compares the accuracy of CoAT models to our baselines. In comparison to Tk-random, CoAT models reach significantly higher accuracy on 41 and 45 of 60 tasks, with comparable performance on 13 and 14 of remaining tasks. The difference is further magnified on *reasoning* tasks, which might better reflect on learning tasks' *functional* relations. A comparison of Tk-Info with Tk-random shows that CoAT's improvements are mainly fostered by the *informativeness* condition. SuperGLUE evaluations (Table 1) show similar trends: with a single exception, models utilising a concept-sharing selection of demonstrations (Tk-CoAT and Tk-Info) consistently reach higher scores than Tk-random. Our analyses reveal that the primary difference is in models' ability to follow the instruction; in 7 out of 20 evaluations, Tk-random responds out of valid label space.

Figure 3 (right) compare CoAT trained on two tasks with the models of previous work, trained on mixtures of 35–1,836 tasks. In *All tasks*, CoAT models are comparable on the majority of tasks in 5 out of 6 competitions. The evaluation on reasoning tasks supports our hypothesis that CoAT particularly promotes improvements in learning new *reasoning* abilities, winning on this segment over Flan and Tk-Instruct in a comparable number of cases as the opponents. Table 2 details models' scores on SuperGLUE tasks, providing further evidence on a comparability of CoAT models to multitask learners. For instance, a comparison with Tk-Instruct reveals that CoAT's 1B and 3B models reach higher absolute results on 3 and 5 out of the 7 Tk-Instruct's unseen tasks. More comparisons with previous models can be found in Appendix D.1.

	AxG	Ax-b	WSC	CB	RTE	WiC	ReCoRD	BoolQ	COPA	MultiRC
Tk-random-1B	49.4±5.2	43.6±4.8	52.7±5.1	21.8±3.9	29.3±4.6	18.0±4.0	15.3±3.8	34.0±5.0	74.7±3.4	5.1±2.4
Tk-random-3B	50.2±5.4	57.5±4.8	52.0±5.5	47.8±5.1	48.9±4.8	50.1±4.4	16.3±7.3	62.8±4.6	75.5±2.8	2.1±1.5
Tk-info-1B	50.0±4.2	42.6±5.7	52.0±4.3	47.2±3.9	49.2±4.8	53.2±4.5	15.5±4.0	19.6±2.3	61.5±2.3	3.2±1.2
Tk-info-3B	50.8±4.6	57.2±4.9	53.5±4.8	47.3±5.4	54.7±4.9	53.6±4.7	22.6±4.5	64.4±4.8	76.3±3.0	2.7±2.1
Tk-CoAT-1B	50.4±5.3	52.7±4.6	53.6±5.2	46.9±4.9	53.7±4.9	53.5±5.3	17.0±3.5	63.8±5.4	76.1±3.2	11.4±2.6
Tk-CoAT-3B	57.9±4.9	57.2±4.8	53.6±4.5	60.4±4.8	52.0±5.4	56.9±5.0	23.1±3.8	63.6±4.3	81.3±3.3	56.9±3.6

Table 1: **Efficiency of concept-aware training: SuperGLUE:** ROUGE-L scores of ICL models evaluated in few-shot setting on SuperGLUE tasks (Wang et al., 2019), trained using (i) *random* demonstrations sampling used in previous work, (ii) *informative* demonstrations sampling (§3) and (iii) *informative+non-trivial* sampling (CoAT; §2). Underlined are the best results per each task and model size.

	# train tasks	AxG	Ax-b	WSC	CB	RTE	WiC	ReCoRD	BoolQ	COPA	MultiRC
Flan-1B	1,836	84.8±3.9	21.9±4.0	70.7±4.8	92.5±2.8*	92.1±3.0*	69.9±5.1*	38.9±5.2*	92.3±2.7*	97.8±1.5*	88.3±3.2*
Flan-3B	1,836	95.3±3.7	22.0±8.0	80.2±9.2	92.7±6.7*	96.0±4.0*	79.7±8.3*	62.2±9.7*	92.1±5.1*	99.3±1.6*	90.4±6.4*
Tk-Instruct-1B	1,616	51.9±4.9	57.2±5.8	49.8±4.9	46.0±5.5	55.5±4.8	53.5±5.3	13.1±3.7	63.4±3.4*	76.9±3.2*	62.2±5.1*
Tk-Instruct-3B	1,616	53.5±4.7	49.9±4.9	51.2±4.9	66.3±4.6	62.7±4.6	50.4±4.8	18.6±4.2	68.8±4.4*	73.8±3.5*	59.9±4.9*
T0-3B	35	65.0±4.5	36.1±4.6	53.5±5.2	48.0±5.4	51.3±5.2	54.0±5.0	20.5±4.0	60.1±4.9	56.8±3.6	56.2±4.4
Tk-CoAT-1B	2	50.4±5.3	52.7±4.6	53.6±5.2	46.9±4.9	53.7±4.9	53.5±5.3	17.0±3.5	63.8±5.4	76.1±3.2	11.4±2.6
Tk-CoAT-3B	2	57.9±4.9	57.2±4.8	53.6±4.5	60.4±4.8	52.0±5.4	56.9±5.0	23.1±3.8	63.6±4.3	81.3±3.3	56.9±3.6

Table 2: **Concept-aware training vs previous models: SuperGLUE:** ROUGE-L of CoAT-trained ICL models and models of comparable size in previous work. Evaluation setup is consistent with Table 1. In cases marked with *, the task was used in the model's training; Underlined are the best results per *unseen* task and model size.

Conclusions

We propose Concept-aware Training (CoAT), a framework for constructing training data that make it beneficial for a language model to learn to utilise latent reasoning concepts. We show that language models *can* learn to perform a concept-based ICL (*RQ1*), and that concept-based ICL *is* more robust in learning *functional* relations from demonstrations (*RQ2*). Finally, we find that concept-based ICL also *brings* performance gains in the ICL of a majority of unseen tasks (*RQ3*), performing comparably to models trained on over 1,600 tasks when trained with only *two* QA tasks.

More broadly, we explore an *alternative* axis to scaling the ICL quality, complementing the known *model* and *data scale* axes. We wish to inspire future work towards a more *proactive* consideration of properties of training data so that fitting such data *necessitates* the emergence of specific, robust capabilities, such as the concept modelling ability.

REFERENCES

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? Investigations with linear models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=0g0X4H8yN4I.

Stephen H. Bach, Victor Sanh, Zheng-Xin Yong, Albert Webson, Colin Raffel, Nihal V. Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, Zaid Alyafeai, Manan Dey, Andrea Santilli, Zhiqing Sun, Srulik Ben-David, Canwen Xu, Gunjan Chhablani, Han Wang, Jason Alan Fries, Maged S. Al-shaibani, Shanya Sharma, Urmish Thakker, Khalid Almubarak, Xiangru Tang, Xiangru Tang, Mike Tian-Jian Jiang, and Alexander M. Rush. PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts, 2022.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. Improving Question Answering Model Robustness with Synthetic Adversarial Data Generation. In *Proceedings of the 2021 Conference EMNLP*, pp. 8830–8848, Online and Punta Cana, Dominican Republic, November 2021. ACL. doi: 10.18653/v1/2021.emnlp-main.696. URL https://aclanthology.org/2021.emnlp-main.696.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in NIPS*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Stephanie C.Y. Chan, Adam Santoro, Andrew Kyle Lampinen, Jane X Wang, Aaditya K Singh, Pierre Harvey Richemond, James McClelland, and Felix Hill. Data Distributional Properties Drive Emergent In-Context Learning in Transformers. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=lHj-q9BSRjF.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating Large Language Models Trained on Code, 2021.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. PaLM: Scaling Language Modeling with Pathways, 2022.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling Instruction-Finetuned Language Models. *arXiv e-prints*, art. arXiv:2210.11416, October 2022. doi: 10.48550/arXiv.2210.11416.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of the 2019 Conference of the NAACL: Human Language Technologies*, pp. 4171–4186, Minneapolis, USA, June 2019. ACL. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Hao Fu, Yao; Peng and Tushar Khot. How does GPT Obtain its Ability? Tracing Emergent Abilities of Language Models to their Sources. *Yao Fu's Notion*, December 2022. URL https://yaofu.notion.site/How-does-GPT-Obtain-its-Ability-Tracing-Emergent-Abilities-of-Language-Models-to-their-Sources-b9a57ac0fcf74f30a1ab9e3e36fa1dc1.

Michael Hahn and Navin Goyal. A Theory of Emergent In-Context Learning as Implicit Structure Induction, 2023.

Naoya Inoue, Pontus Stenetorp, and Kentaro Inui. R4C: A benchmark for evaluating RC systems to get the right answer for the right reason. In *Proceedings of the 58th Annual Meeting of the ACL*, pp. 6740–6750, Online, July 2020. ACL. doi: 10.18653/v1/2020.acl-main.602. URL https://aclanthology.org/2020.acl-main.602.

Peter Jansen, Elizabeth Wainwright, Steven Marmorstein, and Clayton Morrison. WorldTree: A corpus of explanation graphs for elementary science questions supporting multi-hop inference. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://aclanthology.org/L18-1433.

Yingcong Li, M. Emrullah Ildiz, Dimitris Papailiopoulos, and Samet Oymak. Transformers as Algorithms: Generalization and Stability in In-context Learning, 2023a.

Yingcong Li, Muhammed Emrullah Ildiz, Dimitris Papailiopoulos, and Samet Oymak. Transformers as Algorithms: Generalization and and Stability in In-context Learning, 2023b.

Chin-Yew Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. ACL. URL https://aclanthology.org/W04-1013.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What Makes Good In-Context Examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pp. 100–114, Dublin, Ireland and Online, May 2022. ACL. doi: 10.18653/v1/2022.deelio-1.10. URL https://aclanthology.org/2022.deelio-1.10.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity. In *Proceedings of the 60th Annual Meeting of the ACL (Volume 1: Long Papers)*, pp. 8086–8098, Dublin, Ireland, May 2022. ACL. doi: 10.18653/v1/2022.acl-long.556. URL https://aclanthology.org/2022.acl-long.556.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a Suit of Armor Conduct Electricity? A New Dataset for Open Book Question Answering. In *Proceedings of the 2018 Conference EMNLP*, pp. 2381–2391, Brussels, Belgium, October-November 2018. ACL. doi: 10.18653/v1/D18-1260. URL https://aclanthology.org/D18-1260.

Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. MetaICL: Learning to learn in context. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2791–2809, Seattle, United States, July 2022a. Association for Computational Linguistics. doi: 10.18653/v1/2022.naacl-main.201. URL https://aclanthology.org/2022.naacl-main.201.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work?, 2022b. URL https://arxiv.org/abs/2202.12837.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022.

Mohammad Taher Pilehvar and Jose Camacho-Collados. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 1267–1273, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1128. URL https://aclanthology.org/N19-1128.

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. Gender bias in coreference resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, 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, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M Rush. Multitask Prompted Training Enables Zero-Shot Task Generalization. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=9Vrb9D0WI4.

Timo Schick and Hinrich Schütze. Exploiting cloze questions for few shot text classification and natural language inference, 2020. URL https://arxiv.org/abs/2001.07676.

Michal Štefánik and Marek Kadlčík. Can in-context learners learn a reasoning concept from demonstrations? In Bhavana Dalvi Mishra, Greg Durrett, Peter Jansen, Danilo Neves Ribeiro, and Jason Wei (eds.), *Proceedings of the 1st Workshop on Natural Language Reasoning and Structured Explanations (NLRSE)*, pp. 107–115, Toronto, Canada, June 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.nlrse-1.8. URL https://aclanthology.org/2023.nlrse-1.8.

Michal Štefánik, Vít Novotný, Nikola Groverová, and Petr Sojka. Adaptor: Objective-Centric Adaptation Framework for Language Models. In *Proceedings of the 60th Annual Meeting of the ACL: System Demonstrations*, pp. 261–269, Dublin, Ireland, May 2022. ACL. doi: 10.18653/v1/2022.acl-demo.26. URL https://aclanthology.org/2022.acl-demo.26.

Michal Štefánik, Marek Kadlčík, Piotr Gramacki, and Petr Sojka. Resources and Few-shot Learners for In-context Learning in Slavic Languages. In *Proceedings of the 9th Workshop on Slavic Natural Language Processing 2023 (SlavicNLP 2023)*, pp. 94–105, Dubrovnik, Croatia, May 2023. ACL. URL https://aclanthology.org/2023.bsnlp-1.12.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, YaGuang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Vincent Zhao, Yanqi Zhou, Chung-Ching Chang, Igor Krivokon, Will Rusch, Marc Pickett, Pranesh Srinivasan, Laichee Man, Kathleen Meier-Hellstern, Meredith Ringel Morris, Tulsee Doshi, Renelito Delos Santos, Toju Duke, Johnny Soraker, Ben Zevenbergen, Vinodkumar Prabhakaran, Mark Diaz, Ben Hutchinson, Kristen Olson, Alejandra Molina, Erin Hoffman-John, Josh Lee, Lora Aroyo, Ravi Rajakumar, Alena Butryna, Matthew Lamm, Viktoriya Kuzmina, Joe Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguera-Arcas, Claire Cui, Marian Croak, Ed Chi, and Quoc Le. LaMDA: Language Models for Dialog Applications, 2022.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Teaching Broad Reasoning Skills for Multi-Step QA by Generating Hard Contexts. In *Proceedings of the 2022 Conference EMNLP*, pp. 6541–6566, Abu Dhabi, United Arab Emirates, December 2022. ACL. URL https://aclanthology.org/2022.emnlp-main.439.

Michal Štefánik and Marek Kadlčík. Can in-context learners learn a reasoning concept from demonstrations? In *Proceedings of ACL 2023: Natural Language Reasoning and Structured Explanations (NLRSE)*. ACL, 2023. URL https://arxiv.org/abs/2212.01692.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proc. of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. ACL. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems. *arXiv preprint 1905.00537*, 2019.

Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. Towards Understanding Chain-of-Thought Prompting: An Empirical Study of What Matters. In *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2023. URL https://openreview.net/forum?id=L9UMeoeU2i.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. Super-NaturalInstructions: Generalization via Declarative Instructions on 1600+ NLP Tasks. In *Proceedings of the 2022 Conference EMNLP*, pp. 5085–5109, Abu Dhabi, United Arab Emirates, December 2022. ACL. URL https://aclanthology.org/2022.emnlp-main.340.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners, 2021. URL https://arxiv.org/abs/2109.01652.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. Chain of Thought Prompting Elicits Reasoning in Large Language Models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=_VjQlMeSB_J.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and Tengyu Ma. Larger language models do in-context learning differently, 2023.

Noam Wies, Yoav Levine, and Amnon Shashua. The Learnability of In-Context Learning, 2023.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-Art Natural Language Processing. In *Proc. of the 2020 Conf. EMNLP: System Demonstrations*, pp. 38–45. ACL, October 2020. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An Explanation of In-context Learning as Implicit Bayesian Inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=RdJVFCHjUMI.

Zhengnan Xie, Sebastian Thiem, Jaycie Martin, Elizabeth Wainwright, Steven Marmorstein, and Peter Jansen. WorldTree v2: A corpus of science-domain structured explanations and inference patterns supporting multi-hop inference. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pp. 5456–5473, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.671.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the ACL: Human Language Technologies*, pp. 483–498, Online, June 2021. ACL. doi: 10.18653/v1/2021.naacl-main.41. URL https://aclanthology.org/2021.naacl-main.41.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the 2018 Conference EMNLP*, pp. 2369–2380, Brussels, Belgium, October-November 2018. ACL. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259.

Sheng Zhang, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. ReCoRD: Bridging the Gap between Human and Machine Commonsense Reading Comprehension. *arXiv e-prints*, art. arXiv:1810.12885, October 2018. doi: 10.48550/arXiv.1810.12885.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A Survey of Large Language Models, 2023.

A Background

Methods for training in-context learners In-context learning ability, including few-shot ICL, was first uncovered in GPT3 Brown et al. (2020) trained unsupervisedly for causal language modelling. With no other substantial differences to previous GPT models, the emergence of ICL was attributed to GPT3's *scale*, having grown to over 170-billion parameters since GPT2 (≈800M params).

Not long after, a pivotal work of Schick & Schütze (2020) on a Pattern-exploiting training (PET) has shown that even much smaller (110M) models like BERT Devlin et al. (2019) can be fine-tuned using self-training in a similarly small data regime, first disputing the assumption on the necessity of the scale in rapidly learning new tasks.

A new branch of autoregressive generation models further undermined the assumption of the size conditioning of ICL. In one of the pivotal works, Min et al. (2022a) fine-tune smaller pre-trained models (<1B parameters) on a large mixture of tasks in the few-shot prompt format and shows that such models are also able to perform well on previously unseen tasks. Following approaches also train smaller models for instruction following Sanh et al. (2022); Wang et al. (2022) on large mixtures of tasks, assuming that the model's ability to learn an unseen task without updates emerges from a large variety of diverse instruction formats and task types. A recently popularised reinforcement learning approach of InstructGPT Ouyang et al. (2022) also presents an adaptation of instruction-following objectives, training on a large variety of instructions with automatic feedback.

Recently, the instruction following approach was complemented by joint training on programming code generation tasks Chen et al. (2021) and by Chain-of-Thought (CoT) objective Wei et al. (2022), where the model is trained to respond with a sequence of natural-language steps deducing its answer (Zhao et al., 2023). Both these extensions were empirically shown to enhance ICL ability Fu & Khot (2022) and were adopted by FLAN models Chung et al. (2022).

Analyses of ICL Despite the accuracy of ICL in many recent LMs, it remains a matter of open discussion as to *why* the in-context learning emerges.

Recent studies shed some light in this direction through controlled experimentation, finding that the LMs' decision-making in ICL does not align with human intuition; Notably, Lu et al. (2022) first report on the sensitivity of LMs to the specific formulation of the instructions in the prompt, while Liu et al. (2022) report on LMs' surprising sensitivity to the ordering of in-context demonstrations. Further, it was shown that LMs perform ICL comparably well when the labels of the demonstrations are randomly shuffled (Min et al., 2022b) or when the presented CoT sequences do not make sense (Wang et al., 2023). We note that such behaviours differ from learning a *functional* relation of inputs and labels from demonstrations that we might expect from in-context learners Li et al. (2023a).

Still, other studies report that under the right conditions, LMs *are* able to learn functional relationships *solely* from the input prompt; For instance, studies of Akyürek et al. (2023); Li et al. (2023b) show that Transformers can be trained to accurately learn regression functions *solely* from the prompt.

Xie et al. (2022) might be the first to identify the causal effects on ICL quality in specific data properties, rather than data scale, identifying the causal in the presence of the latent concepts that the model needs to utilise to improve in the training task (either pre-training or fine-tuning). Related work attributes ICL to similar data irregularities, such as statistical *burstiness* Chan et al. (2022) or *compositionality* (Hahn & Goyal, 2023). Note that these studies are *not* conflicting with the aforementioned empirical results, but rather explain the causes of their success; For instance, in multi-task training, smaller LMs might indeed necessarily learn to identify shared concepts from inputs (Wies et al., 2023).

Our work builds upon these findings, but compared to the referenced studies limited to in-silico experiments, we bring the idea of concept-aware training into real-world settings, implemented with publicly available datasets and widely-used pre-trained models. We measure the impact of concept-aware data construction in *extrinsic* evaluation over 70 diverse tasks and show its potential to substantially enhance data efficiency and robustness in training in-context learners, compared to previous work using *magnitudes* of more data and compute.

B TRAINING DETAILS

Table 3 shows a full training example for each stage of training: (1) TeaBReaC with synthetic contexts (top) and (2) AdversarialQA with natural-language contexts (bottom). In all our training setups, we fine-tune all model parameters for teacher-forced next-token prediction, conventionally used in training sequence-to-sequence language models. In the two training stages (TeaBReaC and AdversarialQA), we use a **learning rate** of $5e^{-5}$ and $2e^{-5}$, respectively. Other parameters remain identical between stages: effective **batch size** = 30 samples and **early stopping** with the patience of 2,000 updates based on evaluation loss on a standardized validation set of each dataset. We do not report the absolute values of evaluation loss as these are not directly comparable. In CoAT training, we use a random subsample of 20 informative examples as a candidate set for a selection of non-trivial demonstrations.

Other parameters of training configuration default to Training Arguments of Transformers library Wolf et al. (2020) in version 4.19.1. For readability, we implement the relatively complex demonstrations' selection as a new objective of the Adaptor library Štefánik et al. (2022). The picked demonstrations are encoded into a format consistent with the evaluation.

Dataset	Concept	Training instruction	Target
TeaBReaC	**Exactly-matching reasoning chain** [*"select"* → *"maximum"* → *"list"* → *"maximum"* → *"sum"*]	"**Input:** how many points did the Monte Vesuvio" score in their two highest scoring matches? **Context:** scores of games of Pentagon". 99 scores of games of monte vesuvio". 67 scores of games of Pentagon". 6 scores of games of monte vesuvio". 76 scores of games of Pentagon". 37 scores of games of monte vesuvio". 56 scores of games of Pentagon". 8 scores of games of Pentagon". 90 scores of games of Pentagon". 20 Answer: **Prediction:** 143 **[2 more examples] Input:** how many points did the Bell 212 score in their two highest scoring games? **Context:** scores of games of bell 212. 90 scores of games of S-50. 54 scores of games of bell 212. 41 scores of games of bell 212. 36 scores of games of S-50. 23 scores of games of bell 212. 6 scores of games of bell 212. 2 scores of games of S-50. **Prediction:** "	
AdversarialQA	**Matching question-word** "Who"	"**Input:** Who was the Speaker in 1909? **Context:** Second, Democrats have always elevated their minority floor leader to the speakership upon reclaiming majority status. Republicans have not always followed this leadership succession pattern. In 1919, for instance, Republicans bypassed James R. Mann, R-IL, who had been minority leader for eight years, and elected Frederick Gillett, R-MA, to be Speaker. Mann "had angered many Republicans by objecting to their private bills on the floor;" also he was a protégé of autocratic Speaker Joseph Cannon, R-IL (1903–1911), and many Members "suspected that he would try to re-centralize power in his hands if elected Speaker." More recently, although Robert H. Michel was the Minority Leader in 1994 when the Republicans regained control of the House in the 1994 midterm elections, he had already announced his retirement and had little or no involvement in the campaign, including the Contract with America which was unveiled six weeks before voting day. **Prediction:** Joseph Cannon, R-IL. **[2 more examples] Input:** Who created the legal system still in use in Florida? **Context:** As a result of these initiatives northeastern Florida prospered economically in a way it never did under Spanish rule. Furthermore, the British governors were directed to call general assemblies as soon as possible in order to make laws for the Floridas and in the meantime they were, with the advice of councils, to establish courts. This would be the first introduction of much of the English-derived legal system which Florida still has today including trial by jury, habeas corpus and county-based government. Neither East Florida nor West Florida would send any representatives to Philadelphia to draft the Declaration of Independence. Florida would remain a Loyalist stronghold for the duration of the American Revolution. **Prediction:** "	"British"

Table 3: Examples of **training instructions** with expected outputs, for both our datasets applied in training. Note that the shared reasoning concept is not a part of the model's input.

C Evaluation details

Tables 4 shows an example of an instruction for each evaluation that we perform within the concept-learning evaluation. For readability, we only shorten the examples of HotpotQA, where we omit some sources of data available for the model. In the case of TeaBReaC not shown in this table, the evaluation prompt format is the same as in training (Table 3), whereas we make sure that the reasoning chains of evaluation samples differ from the training.

SuperGLUE Evaluation format For SuperGLUE tasks, we verbalize both the demonstrations and predicted sample using all available templates within PromptSource library (Bach et al., 2022), obtaining prompts for each demonstration prompt x_i and its label y_i in a free-text form. The prompts commonly contain the full-text match of the possible labels as options for the model.

Following the example of Wang et al. (2022), we additionally prepend the demonstrations and labels with keywords "Input" and "Prediction" and separate demonstrations with new lines. Thus, the resulting input→output pairs in evaluation take this format:

$$\begin{aligned}&\textit{"Input: } x_1 \textit{ Prediction: } y_1 \textit{ <newline>}\\ &\textit{Input: } x_2 \textit{ Prediction: } y_2 \textit{ <newline>}\\ &\textit{Input: } x_3 \textit{ Prediction: } y_3 \textit{ <newline>}\\ &\textit{Input: } x_{pred} \textit{ Prediction: "} \rightarrow \text{"}y_{\text{pred}}\text{"}\end{aligned}$$

where demonstrations (x_i, y_i) are picked randomly but consistently between all evaluated models.

We report results for the best-performing template for each model.

Natural-Instructions Evaluation format In the evaluations on Natural-Instructions, we closely follow the example of Wang et al. (2022) and additionally prepend the sequence of demonstrations with an instruction provided for each task:

Dataset	Concept	Model instruction	Expected output
GLUE NLI Diag.	Double negation	"**Input:** I will say that she stole my money. Question: I won't say that she didn't steal my money. True, False, or Neither? **Prediction:** Neither **Input:** I won't say that she didn't steal my money. Question: I will say that she stole my money. True, False, or Neither? **Prediction:** Neither **Input:** A rabbi is at this wedding, standing right there standing behind that tree. Question: It's not the case that there is no rabbi at this wedding; he is right there standing behind that tree. True, False, or Neither? **Prediction:** True **Input:** Even after now finding out that it's animal feed, I won't ever stop being addicted to Flamin' Hot Cheetos. Question: Even after now finding out that it's animal feed, I will never stop being addicted to Flamin' Hot Cheetos. True, False, or Neither? **Prediction:** "	"True"
OpenBookQA	**Shared facts:** {"Earth is greater in mass than Mars", "gravity means gravitational pull; gravitational force; gravitational attraction", "as the force of gravity increases, the weight of objects will increase."}	"**Facts:** a decrease is a kind of change. increase means more. as mass of a planet; of a celestial body increases, the force of gravity on that planet will increase. to change means to become different. an animal is a kind of living thing. the gravitational force of a planet; of a celestial object does not change the mass of an object on that planet or celestial body. an increase is the opposite of a decrease. an astronaut is a kind of human. massive means great in mass. the Mars Rover is a kind of vehicle. a living thing is a kind of object. Earth is greater in mass than Mars. gravity means gravitational pull; gravitational energy; gravitational force; gravitational attraction. greater means higher; more in value. stay the same means not changing. a moon is a kind of celestial object; body. an increase is a kind of change. Earth is a kind of planet. as the force of gravity increases, the weight of objects will increase. less is similar to decrease. Mars is a kind of planet. **Input:** An object has a weight of 10 kg on the surface of Earth. If the same object were transported to the surface of Mars, the object would have a weight of 3.8 kg. Which best explains why the weight of the object changed when transported from Earth to Mars? (A) The density of the object is greater on Earth than it is on Mars. (B) The volume of the object is greater on Earth than it is on Mars. (C) Gravitational force is greater on Earth than it is on Mars. (D) Atmospheric pressure is less on Earth than it is on Mars. **Prediction:** Gravitational force is greater on Earth than it is on Mars **[two more examples] Input:** When astronauts walked on the Moon, they used weighted boots to help them walk due to the lower gravitational pull. What difference between Earth and the Moon accounts for the difference in gravity? (A) density (B) diameter (C) mass (D) volume. **Prediction:** "	"mass"
HotpotQA	**Shared relation in reasoning**: "X is a genus"	"**Input:** Are Broughtonia and Laeliocattleya both orchids? Hint: use the information from the paragraphs below to answer the question. Otaara, abbreviated Otr. in the horticultural trade, is an intergeneric hybrid of orchids, with "Brassavola", "Broughtonia", "Cattleya", "Laelia" and "Sophronitis" as parent genera. Paracaleana commonly known as duck orchids, is a genus of flowering plants in the orchid family, Orchidaceae that is found in Australia and New Zealand. Duck orchids have a single leaf and one or a few, dull-coloured, inconspicuous flowers. (...) **Prediction:** yes **[two more examples] Input:** Are both Parodia and Thalictrum flowering plants? Hint: use the information from the paragraphs below to answer the question. - Thalictrum ( ) is a genus of 120-200 species of herbaceous perennial flowering plants in the Ranunculaceae (buttercup) family native mostly to temperate regions. Meadow-rue is a common name for plants in this genus. - Parodia is a genus of flowering plants in the cactus family Cactaceae, native to the uplands of Argentina, Peru, Bolivia, Brazil, Colombia and Uruguay. This genus has about 50 species, many of which have been transferred from "Eriocactus", "Notocactus" and "Wigginsia". They range from small globose plants to 1 m tall columnar cacti. All are deeply ribbed and spiny, with single flowers at or near the crown. Some species produce offsets at the base. They are popular in cultivation, but must be grown indoors where temperatures fall below 10 degrees. **Prediction:** "	"yes"
WorldTree	**Relation of objects**: "generate"	"**Input:** Despite what some think, instead around themselves, our planet spins around... Choices: pluto, the moon, the milky way, the sun. **Prediction:** the sun **Input:** In a single year, a giant globe will do this to a giant star. Choices: fight, burn, circle, explode. **Prediction:** circle **Input:** The earth revolves around... Choices: a heat source, the Milky Way, a neighboring planet, the moon. **Prediction:** a heat source **Input:** the central object of our solar system is also... Choices: the smallest object in the solar system, the coldest heavenly body, the farthest star from us, the closest star from us. **Prediction:** "	"the closest star from us"

Table 4: Examples of **evaluation instructions** with expected outputs, for each dataset used in evaluation of in-context learning of new concepts (RQ1). Note that the demonstrations within the instructions share the annotated *Concept* with the following *predicted sample*.

Dataset	Concept	Model instruction	Expected output
SuperGLUE	-	"**Input**: The soldiers were concealed in the brush. Select the most plausible cause: - They were armed with rifles. - They wore camouflage uniforms. **Prediction**: They wore camouflage uniforms. **Input**: The print on the brochure was tiny. Select the most plausible effect: - The man put his glasses on. - The man retrieved a pen from his pocket. **Prediction**: The man put his glasses on. **Input**: I excused myself from the group. Select the most plausible cause: - I turned off my phone. - My phone rang. **Prediction**: My phone rang. **Input**: My body cast a shadow over the grass. Select the most plausible cause: - The sun was rising. - The grass was cut. **Prediction**:"	"The sun was rising."
Natural-Instructions	-	"Indicate with 'Yes' if the given question involves the provided reasoning 'Category'. Indicate with 'No', otherwise. We define five categories of temporal reasoning. First: "event duration" which is defined as the understanding of how long events last. For example, "brushing teeth", usually takes few minutes. Second: "transient v. stationary" events. This category is based on the understanding of whether an event will change over time or not. For example, the sentence "he was born in the U.S." contains a stationary event since it will last forever; however, "he is hungry" contains a transient event since it will remain true for a short period of time. Third: "event ordering" which is the understanding of how events are usually ordered in nature. For example, "earning money" usually comes before "spending money". The fourth one is "absolute timepoint". This category deals with the understanding of when events usually happen. For example, "going to school" usually happens during the day (not at 2 A.M). The last category is "frequency" which refers to how often an event is likely to be repeated. For example, "taking showers" typically occurs 5 times a week, "going to Saturday market" usually happens every few weeks/months, etc. **Input:** Sentence: Jack played basketball after school, after which he was very tired. Question: How long did Jack play basketball? Category: Event Duration. **Prediction:** Yes **Input:** Sentence: He was born in China, so he went to the Embassy to apply for a U.S. Visa. Question: How often does he apply a Visa? Category: Frequency. **Prediction:** Yes **Input:** Sentence: Jack played basketball after school, after which he was very tired. Question: Was Jack still tired the next day? Category: Event Duration. **Prediction:** No **Input:** Sentence: It refers to a woman who is dangerously attractive, and lures men to their downfall with her sexual attractiveness. Question: How long does it take to lure men to their downfall? Category: Event Duration. **Prediction:** "	"Yes"

Table 5: Examples of **evaluation instructions** with expected outputs, for selected tasks of **SuperGLUE** and **Natural-Instructions** (RQ3). Displayed samples are from CoPA and MCTato Temporal Reasoning tasks, respectively. Note that in these evaluations, demonstrations are picked **randomly**, regardless of their concepts.

$$
\begin{aligned}
&\textit{"<task instruction>} \quad \textit{<newline>}\\
&\textit{Input: } x_1 \textit{ Prediction: } y_1 \textit{ <newline>}\\
&\textit{Input: } x_2 \textit{ Prediction: } y_2 \textit{ <newline>}\\
&\textit{Input: } x_3 \textit{ Prediction: } y_3 \textit{ <newline>}\\
&\textit{Input: } x_{pred} \textit{ Prediction: "} \rightarrow \text{"}y_{\text{pred}}\text{"}
\end{aligned}
$$

where the *<task instruction>* contains the instruction as would be given to the annotators of the evaluation task, usually spanning between 3–6 longer sentences. The demonstrations are again picked randomly but consistently between models.

We complement all the evaluations with confidence intervals from the bootstrapped evaluation. We analyse the error cases (Appendix C) and choose to report the results in ROUGE-L for SuperGLUE, and in a standard accuracy for Natural-Instructions. This selection is justified in the following Section.

C.1 Concepts evaluation

Following Štefánik & Kadlčík (2023), we evaluate our models on four natural-language concepts: (i) *reasoning logic* of NLI samples of GLUE-Diagnostic dataset (Wang et al., 2018), (ii) *entity relations* annotated in human explanations (Inoue et al., 2020) in the HotpotQA dataset (Yang et al., 2018), (iii) *functional operations* within general elementary-grade tests of OpenBookQA (Mihaylov et al., 2018), and (iv) shared *facts* in science exams of WorldTree dataset (Jansen et al., 2018; Xie et al., 2020).

Figure 4 shows separate evaluation per each of these concepts. We see that while CoAT improves models' ability to work with concepts in average, this ability is still not consistent, leaving the challenge open for future work.

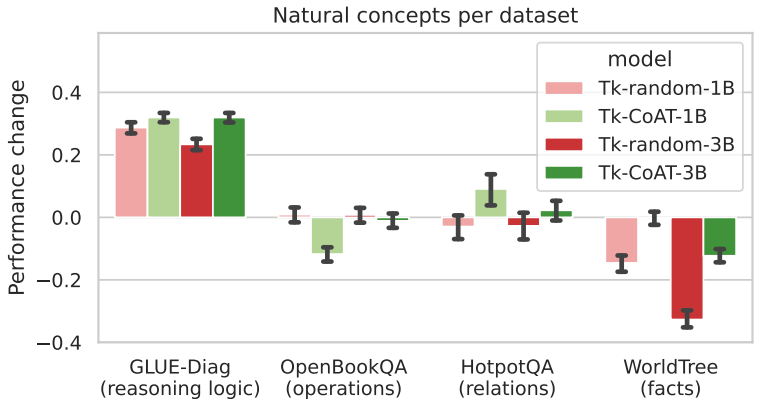

Figure 4: **In-context learning of specific natural concepts**: While CoAT improves the ability to benefit from reasoning concepts on average (Fig. 2), per-concept evaluation reveals that this ability is not consistently robust.

C.2 Evaluation metrics selection

Previous work training in-context few-shot learners is not consistent in the use of evaluation metrics, and the choice usually boils down to either using the exact-match accuracy (Sanh et al., 2022; Chung et al., 2022) or ROUGE-L of Lin (2004) (Wang et al., 2022), evaluating the longest common sequence of tokens. We investigate these two options with the aim of not penalising the models for minor discrepancies in the output format (in the accuracy case) but avoiding false positive evaluations in predictions that are obviously incorrect (in the ROUGE case).

Investigation of the models' predictions reveals that the selection of the metric makes a large difference only in the case of Tk-Instruct models, where the situation differs between SuperGLUE and Natural-Instructions, likely due to the character of the evaluation prompts.

(1) On **SuperGlue**, e.g. on MultiRC task, for the evaluation prompt: "Does answer sound like a valid answer to the question: question", Tk-Instruct-3B in our evaluation predicts "Yes." or "Yes it is" (instead of "Yes"), or "No not at all" (instead of "No"), likely due to the resemblance with the format of training outputs. As we do not wish to penalize these cases, we use ROUGE-L over all SuperGLUE evaluations.

(2) In **Natural-Instructions** evaluation, we find that Tk-Instruct often predicts longer extracts from the input prompt. This is problematic with ROUGE-L in the cases where the extract contains *all* possible answers, such as in the Tk-Instruct-1B's prediction: "yes or no" to the prompt whose instruction ends with "Please answer in the form of yes or no.". As we encounter this behaviour in a large portion of Natural-Instructions tasks, we evaluate all models on Natural-Instructions for exact-match accuracy after the normalization of the casing and the removal of non-alphabetic symbols. To make sure that the model is presented with the exact-matching answer option, we exclude from evaluation the tasks where the correct answer is not presented in the task's instruction. The reference to the list of Natural-Instructions evaluation tasks can be found in Appendix D.4.

For the reported evaluations of the Reasoning tasks, we pick from the list of evaluation tasks the ones concerned with the reasoning task by simply matching the tasks with 'reasoning' in their name, resulting in the collection of 20 evaluation tasks.

D Further evaluations

D.1 SuperGLUE evaluations of other models

Table 2 compares the performance over the tasks of SuperGLUE collection (Wang et al., 2019) for CoAT models trained on two tasks of the same (QA) type with in-context learners trained on 35–

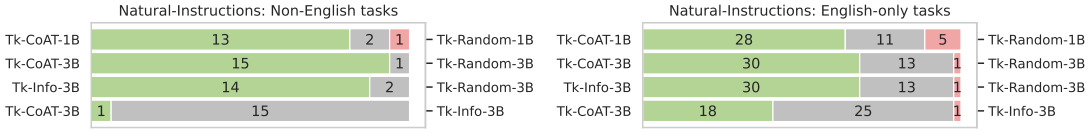

Figure 5: **Impact of Concept-aware training per different language settings:** Pairwise comparison of models trained using selected training configurations (§3) on (top) *Non-English* tasks and (bottom) *English-only* tasks of Natural-Instructions collection. Values in green and red bars indicate a number of tasks where the referenced model reaches significantly higher accuracy than the other. For the tasks denoted as *similar*, the difference in performance falls within the evaluation's confidence intervals.

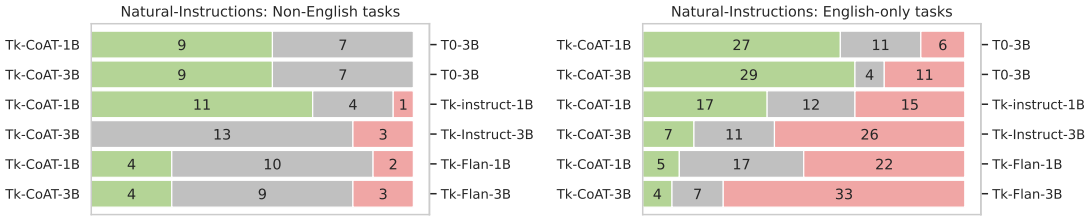

Figure 6: **Comparison to previous work per different language settings:** Pairwise comparison of CoAT models vs. the models of previous work on (top) *Non-English* tasks and (bottom) *English-only* tasks of Natural-Instructions collection. Values denote the number of tasks where the model reaches significantly better accuracy. For the tasks denoted as *similar*, the difference in performance falls within the evaluation's confidence intervals.

1,836 tasks of the comparable size. Despite the significantly smaller volumes and complexity of the training dataset, CoAT-trained models show competitive results to similar-size or even larger in-context learners of previous work. For instance, the 1-billion-parameter Tk-CoAT performs better than the 3-billion T0 in 3 cases (Ax-b, RTE, COPA) and comparably in another 3 cases (WSC, CB, WiC). In comparison with Tk-instruct of the same size, Tk-CoAT-1B outperforms Tk-instruct in 3 out of 7 unseen tasks (WSC, CB, ReCoRD), and reaches similar scores in most other cases, even in 2 out of 3 tasks that were included in Tk-instruct's training mix. Similarly, larger Tk-CoAT-3B outperforms Tk-instruct on 4 of 7 new tasks (Ax-b, WSC, WiC, ReCoRD), but with larger gaps on the others.

D.2 Natural-Instructions: other task types

Figure 5 evaluates the impact of CoAT's mechanism on the quality of in-context learning separately on the English and non-English tasks. The figure reveals that CoAT works particularly well for non-English tasks. Our analyses found this is mainly due to the low performance of the baseline on the non-English tasks. We speculate that this can be a consequence of the higher reliance of the baseline on token semantics (Section 4, RQ2); As our models are fine-tuned on an English-only QA model, such learnt reliance is not applicable in multilingual settings.

Figure 6 compares the performance of CoAT models against the models of previous work, separately on the English and non-English tasks. We can see that CoAT is slightly better at the multilingual portion of Natural-Instructions, but the difference is not principal.

D.3 Per-concept evaluations

Figure 4 evaluates the performance gains of the baseline models (§3) and CoAT-trained models individually per each of the concepts of the natural datasets. While the CoAT models are able to benefit from concepts the largest in the relative change of quality, they are also not consistent in the ability to benefit from all the concepts.

D.4 Evaluation tasks and other configurations

SuperGLUE (Wang et al., 2019) consists of the following tasks (as ordered in our Results, §4): Winogender Schema Diagnostics (AxG) (Rudinger et al., 2018), Broadcoverage Diagnostics (CB), The Winograd Schema Challenge, CommitmentBank (CB), Recognizing Textual Entailment (RTE), ContextWords in Context (WiC) (Pilehvar & Camacho-Collados, 2019), Reading Comprehension with Commonsense Reasoning (ReCoRD) (Zhang et al., 2018), BoolQ (Clark et al., 2019), Choice of Plausible Alternatives (COPA), Multi-Sentence Reading Comprehension (MultiRC).

Natural-Instructions consists of a larger mixture of tasks, which we do not enumerate here to maintain readability; the full list of evaluation tasks can be found in the original work of Wang et al. (2022) in Figures 11 and 12.

To maintain comparability of evaluations among models, we deterministically fix the demonstration selection procedure so that only the full prediction prompts for all the models are the same. In the analyses comparing the differences in performance (§4; RQ1+2), we fixed the prediction samples (x_{pred}) between different demonstrations' sampling strategies to avoid perplexing our comparison with possible data selection biases. Further details can be found in the referenced implementation.

E Computational Requirements

We run both training and evaluation experiments on a machine with dedicated single NVIDIA A100-SXM-80GB, 40GB of RAM and a single CPU core. Hence, all our reproduction scripts can run on this or a similar configuration. Two stages of training in total take at most 6,600 updates and at most 117h of training for Tk-CoAT to converge.