Concept-aware Training Improves In-context Learning of Language Models

Anonymous ACL submission

Abstract

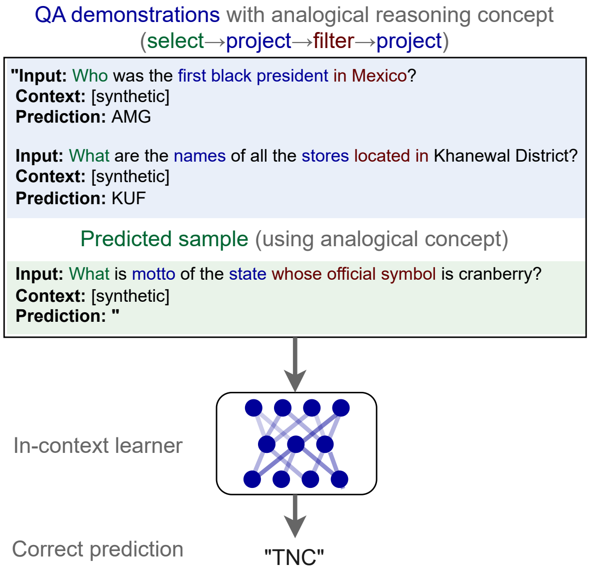
Many recent language models (LMs) of the Transformers family are capable of *in-context learning* (ICL), manifested in the LMs' ability to perform a new task solely from its description in a natural language input. Previous work curating these models assumes that ICL emerges from vast over-parametrization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

Inspired by these findings, we propose a Concept-aware Training (CoAT) method constructing training scenarios that make it beneficial for the LM to learn to utilize the **analogical reasoning concepts**. We measure that data sampling of CoAT substantially improves models' ICL on unseen tasks, resulting in the performance comparable to the previous in-context learners trained on over 1600 tasks when we apply CoAT with only two QA datasets. Our analyses show that CoAT's improvements can be attributed to models' reinforced ability to benefit from natural concepts from demonstrations over the reliance on the pre-trained semantic priors common for previous ICL models.

1 Introduction

The in-context learning (ICL), as initially uncovered by Brown et al. (2020), is a specific task requiring language models (LMs) to infer and apply correct functional relationships from the pairs of inputs and outputs (i.e. *demonstrations*) presented in user-provided 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).


QA demonstrations with analogical reasoning concept
(select→project→filter→project)

"**Input:** Who was the first black president in Mexico?
Context: [synthetic]
Prediction: AMG

Input: What are the names of all the stores located in Khanewal District?
Context: [synthetic]
Prediction: KUF

Predicted sample (using analogical concept)

Input: What is motto of the state whose official symbol is cranberry?
Context: [synthetic]
Prediction: "

In-context learner
Correct prediction
"TNC"

Figure 1: Example from synthetic TeaBReAC dataset with demonstrations sharing analogical reasoning chain. In **Concept-aware Training (CoAT)**, we use such examples in training to enable models to benefit from latent reasoning concepts within in-context learning.

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 10 to 100 times smaller models while reaching comparable ICL quality, instead attributing the ICL ability to a vast volume and diversity of pre-training tasks and instruction formats. Hence, should we claim in-context learning ability to the scale of training data or model size?

The complementary branch of theoretical studies is more specific in identifying covariates responsible for the emergence of ICL in **data irregularities**, i.e. the properties of the data that can *not* be explained by mere statistical co-occurrence of tokens.

Notably, Xie et al. (2022) identify the key property in the occurrence of text dependencies that can be resolved by identifying *latent concepts* that underpin these dependencies. In this and other works that we survey in Section 2, Authors show that ICL can also emerge with *both* small data *and* small models, by curating and training on small synthetic datasets exhibiting specific properties.

In this work, we adapt and empirically verify recent theories on data irregularities fostering ICL in synthetic settings. In Section 3, we propose a data construction method that *encourages* the occurrence of concept-dependent irregularity in training samples, and hence, *requires* models to learn to utilise latent concepts that explain these irregularities (Fig. 1). We refer to this method as **Concept-aware Training** (**CoAT**).

In Section 4, we explore the impact of this adjustment in controlled settings. On a set of over 70 tasks of SuperGLUE and Natural-Instructions, we find that CoAT can largely improve in-context learning performance over commonly-used uncontrolled data selection, in many cases enabling ICL of otherwise not learnable tasks. Consequentially, models trained with CoAT on merely two (2) QA datasets reach performance *comparable* to in-context learners of similar or larger size trained on massive collections of over 1,600 diverse tasks.

Our analyses attribute these improvements to the enhanced ability of CoAT-trained models to recover unseen concepts from demonstrations and to their robustness over labels' *semantics*, in favour of *functional* relations presented in demonstrations.

2 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 and 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 and 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.

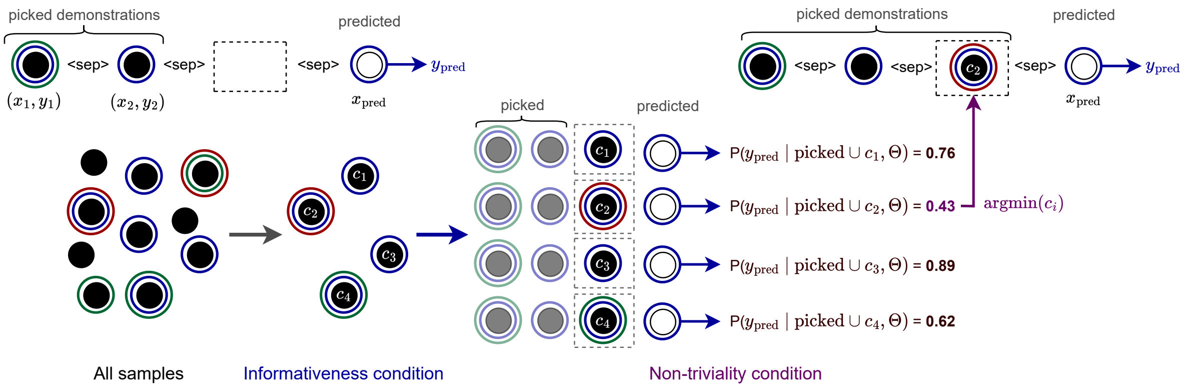


Figure 2: **Demonstrations selection of Concept-aware training (CoAT):** From all samples of the training dataset, we first (i) filter out available samples to ones *sharing* a reasoning concept ○ with predicted sample $(x_{\text{pred}}, y_{\text{pred}})$. From this subset, we (ii) incrementally pick 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*.

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 of the ICL in the presence of the latent concepts that LMs need 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 and 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.

3 Concept-Aware Training (CoAT)

We propose a Concept-Aware Training (CoAT) method that adapts the findings of previous work in data-driven emergence of ICL by applying a **conditional selection of few-shot demonstrations** presented in the training prompts (Figure 2). We assume the format of training prompts widely used in the previous work training in-context few-shot learners, constructing training prompts from k demonstrations consisting of the inputs x with labels y followed by the predicted input 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 in-context demonstrations by two sequential conditions. The main condition, denoted as **informativeness condition**, assures to pick demonstrations that *present* a *reasoning concept* C that is *shared* between a picked demonstration (x_i, y_i) and the predicted example $(x_{\text{pred}}, y_{\text{pred}})$, thus picking only the demonstrations that are *informative* for the correct prediction. In such settings, it is beneficial for the trained model to learn to *extract* and *apply* concepts presented in the input prompt.

However, as the sole *informativeness* condition may pick demonstrations very similar to the predicted sample, we propose a second, **non-triviality condition**. This condition aims to filter the demonstrations to ones with which it is 'difficult' for the model to respond correctly. Further, this condition may increase the heterogeneity of different concepts that co-occur among the demonstrations, avoiding the over-reliance on the presence of a small set of specific concepts in small-data settings.

3.1 Proposed Implementation

We propose to instantiate the CoAT method in two training stages: First, we train LM on a synthetic QA dataset with explicitly annotated reasoning concepts. Second, we refresh the LM's ability to work with natural language prompts by further tuning on a QA dataset with only natural language inputs. Therefore, contrary to previous work, our resulting models are trained on only two QA datasets.

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 explanation maps a natural question to the answer span through a sequence of declarative *reasoning steps*, such as "select→group→project". We use these patterns as informative concepts C and hence, in CoAT, we construct training input texts *only* from demonstrations *sharing* the reasoning chain with the predicted sample (Fig. 1).

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 We implement the *non-triviality condition* of CoAT by (i) selecting a random set of samples $X_{\text{info}} : |X_{\text{info}}| = 20$ from the demonstrations that pass the *Informativeness* condition. (ii) Afterwards, we iteratively pick a sequence of $i \in 1..k$ demonstrations from this set, with a randomly-chosen $k : 2 \leq k \leq 8$.

1. For each sample $(x_j, y_j) \in X_{\text{info}}$ passing *informativeness* condition, we compute a likelihood of generating the correct prediction if a given sample is included among demonstrations. The likelihood is computed as a *product* of likelihoods of generating correct prediction y_{pred} in the teacher-forced generation.
2. In each step i, we add to the demonstrations a sample with which the likelihood of generating correct prediction is *minimal* (Figure 2).

4 Experiments

The primary goal of our experiments is to assess whether the theoretically-supported data construction in CoAT can also enhance the practical quality of ICL in the resulting model. To evaluate this hypothesis, we construct training configurations using the same settings, but either *using* or *not using* CoAT's filters in training data construction.

We follow with analyses attributing CoAT's empirical improvements to changes in specific model's abilities; We assess whether (i) CoAT-trained model can really benefit from presented latent concepts and (ii) whether this ability also applies in a *natural-language* settings. Finally, we explore (iii) whether the gains of CoAT can be attributed to improved ability to override models' sole reliance on the "meaning" of the labels, observed in smaller ICL models (Wei et al., 2023).

4.1 Training and Evaluation Setup

To maximise comparability with the previous work, we fine-tune Tk-CoAT from mT5 pre-trained models of Xue et al. (2021) on (1) TeaBReaC dataset, followed by (2) AdversarialQA dataset. In both stages, we fine-tune all model parameters in teacher-forced next-token prediction (sequence-to-sequence objective) until convergence of evaluation loss in each training stage.[1] We further detail the parameters of the training process in Appendix A.

We survey the evaluation settings adopted in previous work in a few-shot learning with the aim of constructing our evaluation testbed from the widest possible variety of tasks, but avoiding tasks that do not require a reasoning ability, or that are close to the training tasks of ours, or of the previous work we compare to. With this objective, we perform our evaluations on two collections of tasks: (i) SuperGLUE (Wang et al., 2019) consisting of 10 tasks requiring a variety of reasoning skills, and (ii) evaluation set of Natural-Instructions (Wang et al., 2022) from which we pick 60 extractive tasks.

We construct the evaluation scenarios from $k = 3$ randomly but consistently chosen demonstrations consisting of self-contained prompts, with options including the expected label (Sanh et al., 2022). For SuperGLUE tasks, we verbalize both the demonstrations and predicted sample using all available templates within PromptSource library (Bach et al., 2022) for the best-performing template for each model. For Natural-Instructions tasks, we prefix the demonstrations with the instruction provided with each task, consistently with the training format of Tk-Instruct and Flan models. We complement all the evaluations with confidence intervals from bootstrapped evaluation (population $n = 100$, repeats $r = 200$). To maximise fairness of evaluation among the models, we analyse the error cases and choose to report the results in ROUGE-L for SuperGLUE, and in a standard accuracy for Natural-Instructions. We specify the metrics selection analysis and other evaluation details in Appendix B.

[1] All our experiments and final models are on https://github.com/authoranonymous321/concept-training

4.2 Baselines

Random demonstrations selection (Tk-random) We assess the impact of CoAT's controlled selection of demonstrations against a baseline trained in the same settings but picking the in-context demonstrations *randomly* with uniform probability over the whole training set. This methodology of constructing training prompts was used by a majority of the referenced work training smaller in-context learners, including Tk-Instruct (Wang et al., 2022) and Flan (Chung et al., 2022). Apart from the demonstration selection, all other training configurations, including data settings, are identical to §4.1 to assure comparability with CoAT models.

Demonstrations passing only *Informativeness* condition (Tk-info) In this baseline, we perform ablation of the Non-triviality condition introduced in Section 3 by picking the demonstrations passing *only* the *Informativeness* condition. Hence, such-picked demonstrations in the training input context are mutually informative by the shared concept but can exhibit cases where some of the demonstrations are very similar to the predicted sample, making it trivial for the model to perform correct prediction. All other training settings are unchanged (§4.1).

4.3 Other evaluated models

To give additional context to our results, we also evaluate three recent in-context learners for which we can assess which datasets were used in their training mix: (1) **T0** of Sanh et al. (2022) trained on a mixture of 35 datasets of different tasks in zero-shot settings, mostly of QA type, mapped into a self-containing human-understandable interaction format; (2) **Tk-Instruct** of Wang et al. (2022) pre-trained in a few-shot format similar to ours, on a mixture of 1,616 diverse tasks; (3) **Flan** models of Chung et al. (2022) further extend data settings of Tk-Instruct to a total of 1,836 tasks, including chain-of-thought labels, i.e. a step-by-step reasoning chain mapping input prompt to a label.

All these models are based on the same pre-trained model (T5), making the results comparable to the level of fine-tuning methodology. The latter two works use the data construction comparable to our Tk-random but in vastly larger data settings.

4.4 Analyses

In our analyses, we question major assumptions that our implementation of CoAT builds upon.

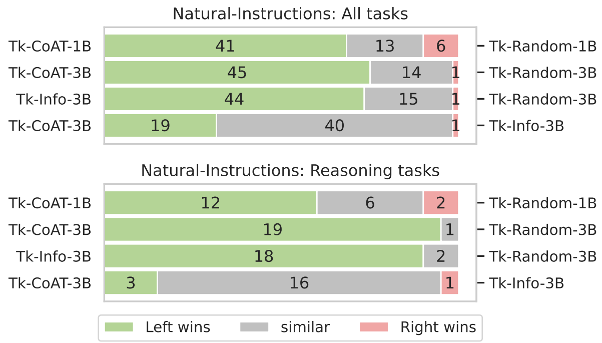


Figure 3: **Efficiency of Concept-aware training: Natural-Instructions:** Pairwise comparison of models trained using selected training configurations (§4.2) on (top) *all* and (bottom) *reasoning* tasks of Natural-Instructions collection. Values in green and red bars indicate a number of tasks where the referenced model reaches significantly higher accuracy than the other. For the tasks denoted as *similar*, the difference in performance falls into the evaluation's confidence intervals.

Can CoAT-trained models really *use* the reasoning concepts? Fundamentally, Concept-aware training assumes that (i) CoAT improves the ability to *extract* and *benefit* from presented reasoning concepts when available, and that (ii) this ability *generalizes* to concepts not observed in training.

If the model can truly utilize a reasoning concept C, it will be able to *improve* on a set of predictions where C is applicable when presented with demonstrations *exhibiting* the use of C in-context. Thus, our first analysis evaluates models' performance in a few-shot setting where we ensure that the demonstrations *share* a concept with the predicted sample. Afterwards, we quantify models' ability to *improve* from the concept by computing the *difference* in accuracy between such concept-sharing evaluation and conventional evaluation using *randomly* chosen demonstrations.

We perform the first analysis on TeaBReAC with annotated *reasoning chains* as concepts C, which are guaranteed to be informative for prediction. To evaluate generalization to unseen concepts, we filter out all samples with reasoning chains that were present in training. This results in 316 evaluation scenarios presenting models with 14 previously unseen reasoning patterns. In this setting, we compare the concept-improving ability of CoAT-trained models with the baseline model (Tk-random).

Can pre-training with *synthetic* demonstrations also improve the of use *natural* concepts? Our implementation of CoAT assumes that the ability to

	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	50.3±5.2	57.2±4.8	53.0±4.5	50.8±2.7	52.0±5.4	53.0±5.6	20.6±3.8	63.6±4.3	81.3±3.3	11.2±3.6

Table 1: **Efficiency of concept-aware training: SuperGLUE:** ROUGE-L scores of ICL models evaluated in few-shot setting on tasks of SuperGLUE (Wang et al., 2019), trained using (i) *random* demonstrations sampling used in previous work, (ii) *informative* demonstrations sampling (§4.2) and (iii) *informative+non-trivial* sampling (CoAT; §3). Underlined are best results per each task and model size.

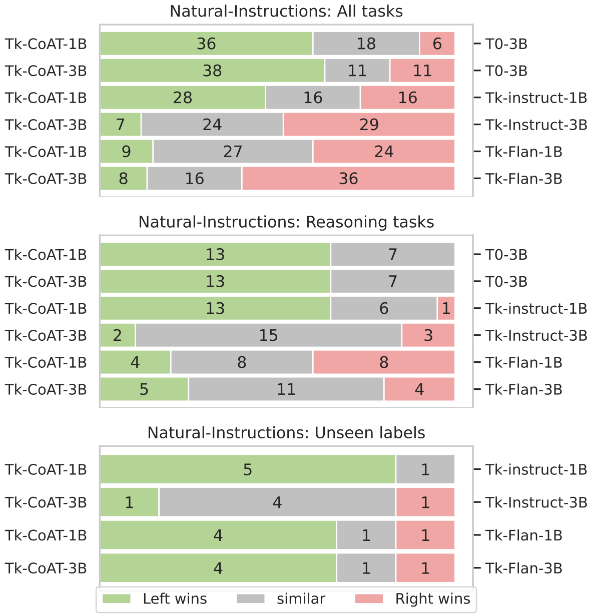


Figure 4: **Performance comparison to previous work: Natural-Instructions:** : Pairwise comparison of CoAT models trained using two (2) tasks vs. the models of previous work trained on mixtures of 35 (T0), 1,616 (Tk-Instruct) and 1,836 tasks (Tk-Flan). Values denote the number of tasks where the model reaches significantly better accuracy. Evaluations over (top) all tasks, (middle) reasoning tasks, (bottom) tasks with labels not present in the training mix of Tk-Instruct and Tk-Flan.

recover reasoning concepts *transfers* from *synthetic* pre-training dataset to *natural*-language applications. Additionally, although TeaBReAC covers over 900 different reasoning chains, it is unclear how relevant these concepts are for real use cases. Therefore, to evaluate if CoAT-trained models improve in recovering concepts also from a *natural* language, we evaluate CoAT-trained models on the ability to *improve* from natural-language demonstrations presenting applicable concepts.

Previous work of Štefánik and Kadlčík (2023) evaluated ICL ability over four different functional concepts, all extracted from *explanations* of natural-language datasets. We adopt the concepts of this work and evaluate models for in-context learning of the following 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* annotated in 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).

Identically to the case of synthetic concepts, we evaluate the ability of CoAT models to benefit from these concepts presented in demonstrations and compare to random demonstrations' selection (Tk-random) used in previous work.

Does concept-aware training mitigate models' over-reliance on learnt semantic priors? As mentioned in Section 2, previous work reports functional deficiencies of previous in-context learners, including surprising insensitivity of in-context learners to the assigned demonstrations' labels (Min et al., 2022b). Wei et al. (2023) attribute this to models' over-reliance on the *semantic priors* obtained in pre-training, which *override* in-context learning of the *functional* relations. However, such behaviour is defective, as the ability to learn *functional* relations is necessary for robust and interpretable in-context learning of truly unseen tasks.

To evaluate the impact of concept-aware training on models' sole reliance on its semantic priors, we follow the setup of Wei et al. (2023) and assess models' reliance on *labels*' semantics in a standard few-shot evaluation (§4.1), with one of the two modifications; (i) Changing the labels to tokens with *irrelevant* meaning for the prediction task, such as 'Foo', 'Bar' etc. (ii) Shuffling the labels so that semantically incorrect labels are assigned in the demonstrations, but the input-label mapping remains consistent. In both settings, the task's functional relation can still be recovered from demonstrations, but the sole reliance on semantics will either not help, or will mislead the model.

In this setting, we evaluate three model types: (i) CoAT-trained models, (ii) models with random training demonstrations (Tk-random), and (iii) models trained identically as Tk-random, but fine-tuned *only* on a natural-context QA dataset (denoted Tk-QA). We perform the evaluation over 8 SuperGLUE tasks with discrete labels.

5 Results

Efficiency of Concept-aware training Evaluation on SuperGLUE presented in Table 1 compares the quality of ICL by models trained using our CoAT implementation (Tk-CoAT; §3) to random demonstrations selection used by previous work (Tk-random). Values show that CoAT significantly improves the quality of in-context learning against the Tk-random baseline in 6 of 10 cases of smaller models and in 3 of 10 cases at larger models, with comparable results in all the other cases. Averages over the scores show that CoAT provides the most substantial gains in the case of the smaller model (+34.4%) by avoiding the failures to understand the task at hand exhibited in CB, RTE and BoolQ with Tk-Random. In smaller models, part of this robustness can be attributed to *both* the Informativeness *and* Non-triviality condition, but the sole concept-aware demonstrations selection (Tk-Info) carries the largest portion of improvements.

While the failures to understand the task seem mostly mitigated in the larger baseline model, the evaluation and our subsequent analyses of models' predictions over Natural-Instruction (NI) tasks (Figure 3) again show a similar trend; For 18 and 24 of 60 Natural-Instructions' tasks, 1B and 3B Tk-Random completely misunderstands the provided instruction, responding mostly outside the domain of labels. This is the case only for 5 and 4 NI tasks in the case of 1B and 3B CoAT models. The advance of CoAT models is further magnified when we zoom to only the reasoning tasks. Evaluations by other task types can be found in Appendix C.2.

Comparison to multitask learners Figure 4 compares the performance of CoAT models with the models of previous work, trained on large mixtures of 35–1,836 tasks. In the comparison over all the NI tasks (Fig. 4; *top*), we can see that the performance of CoAT models is better or comparable for the majority of the tasks in 5 out of 6 competitions.

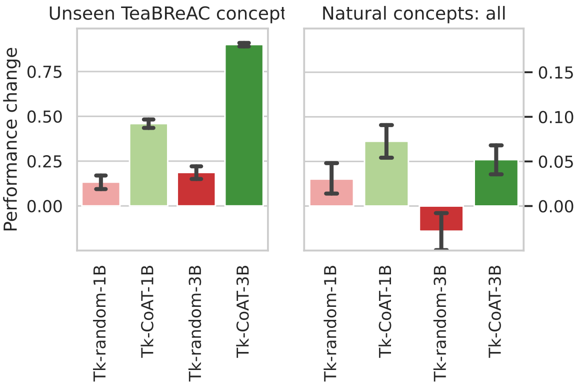


Figure 5: **In-context learning of new concepts**: Improvements of in-context learners when presented with demonstrations exhibiting an informative reasoning concept. Evaluated with *synthetic* examples of TeaBReAC (left), and diverse *natural* examples (right; §4.4).

Despite being explicitly trained with the instructions, our per-task analyses show that Tk-Instruct models also fail to understand instructions in 9 and 4 cases for 1B and 3B models, respectively.

The evaluation on reasoning tasks (Fig. 4; *middle*) supports our hypothesis that CoAT particularly promotes improvements in in-context learning of new reasoning ability, winning on reasoning tasks over Flan and Tk-Instruct in a comparable number of cases than the opponents. Finally, we look at a few tasks where Tk-Instruct and Flan can not rely on the exposition of labels presented in their training mix (Fig. 4; *bottom*). We find that in 3 out of 4 comparisons, CoAT models reach significantly better accuracy on the majority of these tasks.

Further evaluations evidencing comparability of CoAT models with multitask learning are available in Appendix C; Noticeably, a comparison with Tk-Instruct on SuperGLUE shows that CoAT's 1B and 3B models reach higher absolute results on 3 and 4 out of the 7 Tk-Instruct's unseen tasks.

5.1 Analyses

Concept-aware training improves the ability to benefit from unseen concepts Figure 5 evaluates models' ability to *improve* from presented concepts as the absolute difference in performance between random and concept-sharing demonstration selection. Specifically, evaluation with unseen TeaBReAC concepts (left) also assesses models' ability to extrapolate the utilisation of latent concepts to 14 previously unseen reasoning chains.

Both CoAT and random-demonstration models (§4.2) can improve from concepts presented in demonstrations. However, the improvement of CoAT-trained models is significantly larger and ex-

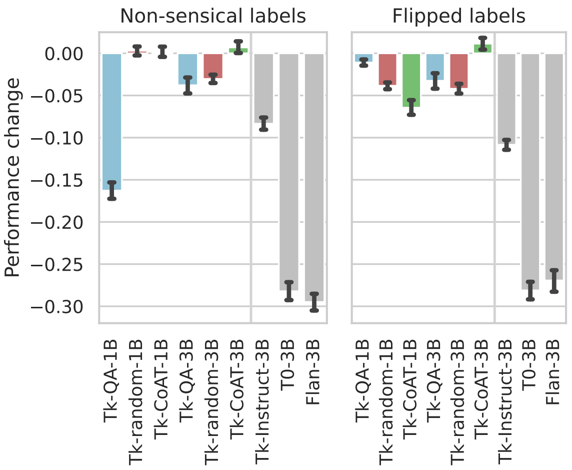


Figure 6: **Models' reliance on semantic priors**: To evaluate models' reliance on its semantic representations, we **(left) replace** labels with 'non-sensical' tokens, with no direct correspondence to the semantics of the task, such as '*foo*', '*bar*', etc.; and **(right) flip** the original labels, so that e.g. '*negative*' label corresponds to a positive-sentiment sample. CoAT models can in-context learn the input-output mapping similarly well with non-sensical labels and rely on the labels' semantics much less than previous in-context learners.

ceeds gains of Tk-random by 2-fold and 4-fold with the smaller and larger model, respectively. This comparison verifies that CoAT's data construction really improves our targeted skill of utilizing latent concepts when presented in demonstrations.

CoAT pre-training on *synthetic* data also improves the use of *natural* concepts Evaluation of improvements on selected natural concepts shown in Figure 5 (right) shows that concept-learning ability obtained on synthetic TeaBReAC concepts indeed transfers to natural-language settings, as the CoAT-trained models can benefit from concepts significantly *more* than models trained without concept-aware data construction (Tk-random).

However, evaluations over the individual reasoning concepts (Figure 7 in Appendix C.3) show that even CoAT models can not benefit robustly from *all* concepts. Nevertheless, we note that in the cases where CoAT models do not improve, also *none* of the baselines benefit from presented concepts. This might be attributed to several reasons: (i) the presented concepts are not really *informative* for prediction, (ii) our training data allowed the models to *memorize* relevant knowledge and, hence, do not *need* (and *benefit from*) the concepts' exposure, or (iii) our training concepts were simply not sufficient to generalize over these new concepts.

CoAT mitigates over-reliance on labels' semantic priors Evaluation with non-sensical labels (Figure 6) shows that models pre-trained on a synthetic TeaBReAC dataset (Tk-random, and Tk-CoAT) can both better comprehend a new task from sole input-output mapping when labels bear no meaning. A comparison of Tk-random and Tk-QA further suggests that the emergence of this property in Tk-CoAT is a composition of *both* using a synthetic dataset in pre-training (also used by Tk-random) *and* CoAT's data construction mechanism.

A comparison to previous models reveals that multitask models experience substantially larger decay in performance than our models, with labels of incorrect meaning in demonstrations. We suspect this may be a bias specific to massive multitask settings where it can explain a large portion of training data. This result is consistent with Wei et al. (2023), but contrary to their conclusions, we show that ICL robust to semantic distractions is *not* an exclusive ability of very large (≥ 100B) models.

Nevertheless, we note that the smaller CoAT model still relies on labels' semantics when recognizable (Flipped labels case), less significantly than previous work, but comparable to our baselines.

6 Conclusion

This paper introduces a Concept-aware Training (CoAT) method; Building upon the recent theories on the emergence of in-context learning, CoAT proposes to train in-context learners in data settings which manifest irregularities necessitating the emergence of in-context learning. We implement CoAT by constructing training prompts with demonstrations that share a reasoning concept with the predicted sample, allowing the trained model to benefit from learning to extract and utilize the reasoning concept that explains the prediction.

We find that data construction of CoAT fosters in-context few-shot learning ability more efficiently than strategies used in previous work. As a result, CoAT delivers performance comparable to models trained on over 1,600 tasks with only two QA tasks while also making models more robust in learning the *functional* relations from demonstrations based on underlying concepts.

In a broader perspective, our work explores an alternative axis of in-context learning to the known *model* and *data scale* axes. We show that concept-aware training presents a fruitful opportunity to enhance the quality and robustness of in-context learning in instructional and conversational models.

Limitations

Although our main objective is to assess the efficiency of concept-aware training, we acknowledge the limitations of our comparison to the previous work, where several aspects convolute the representative comparison of different in-context learners: (i) each of the multitask learners was trained on a different, yet massive set of tasks, making it difficult to find a broader collection that is *new* for multiple models; For this purpose, we surveyed three standard collections used for few-shot evaluation: CLUES (Mukherjee et al., 2021), RAFT (Alex et al., 2021) and FLEX (Bragg et al., 2021), but found in total only three tasks unseen by the multitask learners of previous work, all of the same type (classification). Therefore, we use in our evaluations (a) Tk-Instruct's own evaluation set and (b) SuperGLUE with a significant overlay with the training tasks of previous work. (ii) many aspects make it "easier" for the model to improve, including the domain of labels or prompt format matching the training distribution (relevant to Tk-Instruct and Flan evaluated on Natural-Instructions).

Another aspect that we neglect in our experiments in favour of more in-depth analyses is the *impact of pretraining* projected into the properties of the foundation model that we use. We pick mT5 as a base model for our experiments to maximise comparability with previous methods. While we do not identify any concrete reason to assume that CoAT would perform worse with other base models, one should note that our results do not provide any evidence against such an assumption.

Finally, we note that the applicability of CoAT is conditioned by the availability of the annotated *concepts C* in the training datasets, which might be difficult to obtain for natural-language datasets. Our implementation circumvents this issue by using a synthetically curated dataset; In our experiments, we simultaneously show that concept-aware abilities can also be obtained in the restrictive settings of synthetic-dataset pre-training. However, from our experiments, it remains an open question as *to what extend* could further extension of synthetically-generated datasets, possibly covering even more complex concepts, *scale* to further gains.

Ethical Considerations & Broader Impact

The primary motivation of our work is to minimise the computing demands for the creation of accurate in-context learners. We believe that our presented method, as well as the future data-efficient methods driven by our still-deepening understanding of in-context learning, will enable the democratization of the creation of robust and accurate in-context learning models for both research and industry.

Finally, we note that data-efficient methods (as opposed to *multitask training*) for training ICLs might open possibilities for creating accurate ICLs specialized to languages outside English, where training data is scarce. We look forward for the future work exploring the potential of fine-tuning specifically on the target-language datasets, creating in-context learners specially tailored for target languages outside English.

References

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. 2023. What learning algorithm is in-context learning? Investigations with linear models. In *The Eleventh International Conference on Learning Representations*.

Neel Alex, Eli Lifland, Lewis Tunstall, Abhishek Thakur, Pegah Maham, C. Jess Riedel, Emmie Hine, Carolyn Ashurst, Paul Sedille, Alexis Carlier, Michael Noetel, and Andreas Stuhlmüller. 2021. RAFT: A real-world few-shot text classification benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

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. 2022. PromptSource: An Integrated Development Environment and Repository for Natural Language Prompts.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. 2021. Improving Question Answering Model Robustness with Synthetic Adversarial Data Generation. In *Proceedings of the 2021 Conference EMNLP*, pages 8830–8848, Online and Punta Cana, Dominican Republic. ACL.

Jonathan Bragg, Arman Cohan, Kyle Lo, and Iz Beltagy. 2021. Flex: Unifying evaluation for few-shot nlp. In *Neural Information Processing Systems*.

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. 2020. Language Models are Few-Shot Learners. In *Advances in NIPS*, volume 33, pages 1877–1901. Curran Associates, Inc.

Stephanie C.Y. Chan, Adam Santoro, Andrew Kyle Lampinen, Jane X Wang, Aaditya K Singh, Pierre Harvey Richemond, James McClelland, and Felix Hill. 2022. Data Distributional Properties Drive Emergent In-Context Learning in Transformers. In *Advances in Neural Information Processing Systems*.

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. 2021. Evaluating Large Language Models Trained on Code.

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. 2022. PaLM: Scaling Language Modeling with Pathways.

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. 2022. Scaling Instruction-Finetuned Language Models. *arXiv e-prints*, page arXiv:2210.11416.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. 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)*, pages 2924–2936, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of the 2019 Conference of the NAACL: Human Language Technologies*, pages 4171–4186, Minneapolis, USA. ACL.

Hao Fu, Yao; Peng and Tushar Khot. 2022. How does GPT Obtain its Ability? Tracing Emergent Abilities of Language Models to their Sources. *Yao Fu's Notion*.

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

Naoya Inoue, Pontus Stenetorp, and Kentaro Inui. 2020. 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*, pages 6740–6750, Online. ACL.

Peter Jansen, Elizabeth Wainwright, Steven Marmorstein, and Clayton Morrison. 2018. 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. European Language Resources Association (ELRA).

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

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

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. ACL.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. 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*, pages 100–114, Dublin, Ireland and Online. ACL.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. 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)*, pages 8086–8098, Dublin, Ireland. ACL.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a Suit of Armor Conduct Electricity? A New Dataset for Open Book Question Answering. In *Proceedings of the 2018 Conference EMNLP*, pages 2381–2391, Brussels, Belgium. ACL.

Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2022a. 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*, pages 2791–2809, Seattle, United States. Association for Computational Linguistics.

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

Subhabrata Mukherjee, Xiaodong Liu, Guoqing Zheng, Saghar Hosseini, Hao Cheng, Greg Yang, Christopher Meek, Ahmed Hassan Awadallah, and Jianfeng Gao. 2021. Few-shot learning evaluation in natural language understanding. In *NeurIPS 2021*.

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. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. 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)*, pages 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. 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. 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. 2022. Multitask Prompted Training Enables Zero-Shot Task Generalization. In *International Conference on Learning Representations*.

Timo Schick and Hinrich Schütze. 2020. Exploiting cloze questions for few shot text classification and natural language inference.

Michal Štefánik, Marek Kadlčík, Piotr Gramacki, and Petr Sojka. 2023. 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)*, pages 94–105, Dubrovnik, Croatia. ACL.

Michal Štefánik, Vít Novotný, Nikola Groverová, and Petr Sojka. 2022. Adaptor: Objective-Centric Adaptation Framework for Language Models. In *Proceedings of the 60th Annual Meeting of the ACL: System Demonstrations*, pages 261–269, Dublin, Ireland. ACL.

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. 2022. LaMDA: Language Models for Dialog Applications.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Teaching Broad Reasoning Skills for Multi-Step QA by Generating Hard Contexts. In *Proceedings of the 2022 Conference*

EMNLP, pages 6541–6566, Abu Dhabi, United Arab Emirates. ACL.

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

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

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. 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*, pages 353–355, Brussels, Belgium. ACL.

Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. 2023. Towards Understanding Chain-of-Thought Prompting: An Empirical Study of What Matters. In *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*.

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. 2022. Super-NaturalInstructions: Generalization via Declarative Instructions on 1600+ NLP Tasks. In *Proceedings of the 2022 Conference EMNLP*, pages 5085–5109, Abu Dhabi, United Arab Emirates. ACL.

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

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*.

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

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

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. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proc. of the 2020 Conf. EMNLP: System Demonstrations*, pages 38–45. ACL.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An Explanation of In-context Learning as Implicit Bayesian Inference. In *International Conference on Learning Representations*.

Zhengnan Xie, Sebastian Thiem, Jaycie Martin, Elizabeth Wainwright, Steven Marmorstein, and Peter Jansen. 2020. 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*, pages 5456–5473, Marseille, France. European Language Resources Association.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. 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*, pages 483–498, Online. ACL.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the 2018 Conference EMNLP*, pages 2369–2380, Brussels, Belgium. ACL.

Sheng Zhang, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. 2018. ReCoRD: Bridging the Gap between Human and Machine Commonsense Reading Comprehension. *arXiv e-prints*, page 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. 2023. A Survey of Large Language Models.

A Training details

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.

B Evaluation details

SuperGLUE Evaluation format As mentioned in Section 4.1, we verbalize both the demonstrations and predicted sample using all available templates of 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:

"*Input:* x_1 *Prediction:* y_1 *<newline>*
Input: x_2 *Prediction:* y_2 *<newline>*
Input: x_3 *Prediction:* y_3 *<newline>*
Input: x_{pred} *Prediction:* " → "y_{pred}"

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

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:

"*<task instruction>* *<newline>*
Input: x_1 *Prediction:* y_1 *<newline>*
Input: x_2 *Prediction:* y_2 *<newline>*
Input: x_3 *Prediction:* y_3 *<newline>*
Input: x_{pred} *Prediction:* " → "y_{pred}"

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.

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

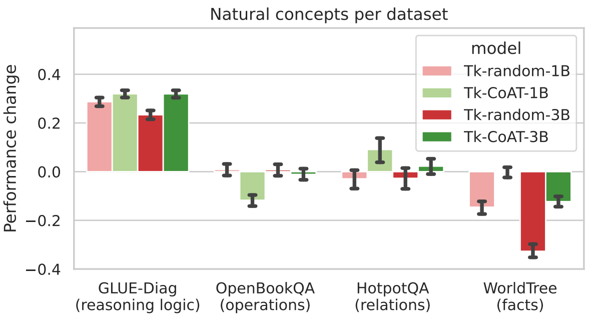


Figure 7: **In-context learning of specific natural concepts**: While CoAT improves the ability to benefit from reasoning concepts on average (Fig. 5), per-concept evaluation reveals that this ability is not consistently robust.

to the list of Natural-Instructions evaluation tasks can be found in Appendix C.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.

C Further evaluations

C.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 35–1,836 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.

C.2 Natural-Instructions: other task types

Figure 8 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 5.1); As our models are fine-tuned on an English-only QA model, such learnt reliance is not applicable in multilingual settings.

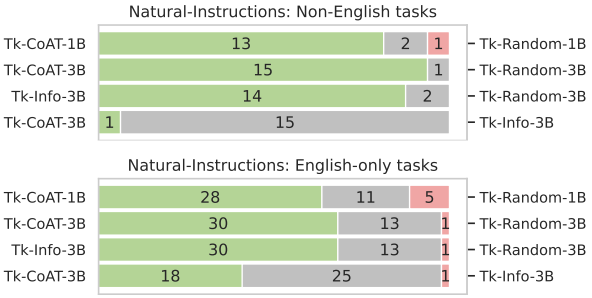


Figure 8: **Impact of Concept-aware training per different language settings:** Pairwise comparison of models trained using selected training configurations (§4.2) on (top) *Non-English* tasks and (bottom) *English-only* tasks of Natural-Instructions collection. Values in green and red bars indicate 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 9 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.

C.3 Per-concept evaluations

Figure 7 evaluates the performance gains of the baseline models (§4.2) 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. However, as discussed in Section 5.1, this does not imply that CoAT is unable to utilize these concepts.

C.4 Evaluation tasks and other configurations

SuperGLUE (Wang et al., 2019) consists of the following tasks (as ordered in our Results, §5): Winogender Schema Diagnostics (AxG) (Rudinger et al., 2018), Broadcoverage Diagnostics (CB), The Winograd Schema Challenge, Commitment-Bank (CB), Recognizing Textual Entailment (RTE),

	# 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		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		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		50.3±5.2	57.2±4.8	53.0±4.5	50.8±2.7	50.6±5.4	53.0±5.6	20.6±3.8	63.6±4.3	75.3±3.3	11.2±3.6

Table 2: **ICL performance: comparison to previous ICL models** ROUGE-L of CoAT-trained ICL models and models of comparable size in previous work. Evaluation setup consistent with Table 1 and (§4.1). In cases marked with *, the task was used in the model's training; Underlined are the best results per unseen task and model size.

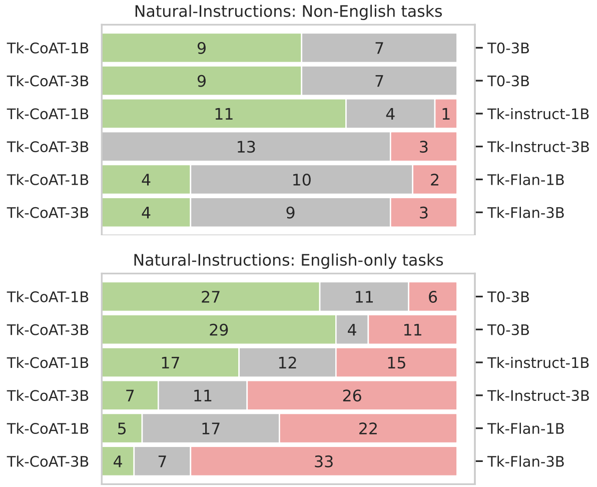


Figure 9: **Comparison to previous work per different language settings:** Pairwise comparison of CoAT models vs. the models of previous work on (top) *Non-English* tasks and (bottom) *English-only* tasks of Natural-Instructions collection. Values denote the number of tasks where the model reaches significantly better accuracy. For the tasks denoted as *similar*, the difference in performance falls within the evaluation's confidence intervals.

ContextWords in Context (WiC) (Pilehvar and 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 (§4.4), 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.

D 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.