Mind the instructions: a holistic evaluation of consistency and interactions in prompt-based learning

Lucas Weber
University Pompeu Fabra
lucas.weber@upf.edu

Elia Bruni
Osnabrück University
elia.bruni@gmail.com

Dieuwke Hupkes
FAIR
dieuwkehupkes@meta.com

Abstract

Finding the best way of adapting pre-trained language models to a task is a big challenge in current NLP. Just like the previous generation of *task-tuned* models (TT), models that are adapted to tasks via in-context-learning (ICL) are robust in some setups but not in others. Here, we present a detailed analysis of which design choices cause instabilities and inconsistencies in LLM predictions. First, we show how spurious correlations between input distributions and labels – a known issue in TT models – form only a minor problem for prompted models. Then, we engage in a systematic, holistic evaluation of different factors that have been found to influence predictions in a prompting setup. We test all possible combinations of a range of factors on both vanilla and instruction-tuned (IT) LLMs of different scale and statistically analyse the results to show which factors are the most influential, interactive or stable. Our results show which factors can be used without precautions and which should be avoided or handled with care in most settings.

1 Introduction

Transfer learning from large-scale pre-trained language models is nowadays the standard approach to a wide range of NLP tasks. One of its great challenges is to optimally interface information that pre-trained language models accumulate in their parameters and adapt it to the task of interest (Zhou et al., 2023; Ouyang et al., 2022). The standard approach for task adaptation has recently shifted from updating model parameters for a specific task (from here on task tuning or *TT*) to using prompting-based methods based on in-context learning (from here on *ICL*). ICL can be subdivided into few-shot (Brown et al., 2020) or zero-shot inference (primarily using instruction-tuned models Wei et al., 2022). Both approaches offer certain benefits over TT: it eliminates costly, task-specific finetuning and provides greater flexibility, as a single model can be applied to many tasks. However, ICL also currently yields overall weaker performance compared to task-tuning and is less stable and reliable on many benchmarks (see, e.g. Bang et al., 2023; Ohmer et al., 2023; Min et al., 2022; Lu et al., 2022; Zhao et al., 2021).

While for TT, much research has been conducted to understand weaknesses in the paradigm (for an overview, see Hupkes et al., 2023), the sources of instabilities in ICL remain nebulous. Since ICL is more constrained (less data and no parameter updates), out-of-distribution generalisation has been suggested to be less of a problem (Awadalla et al., 2022; Si et al., 2023). On the other hand, new frontiers emerge. For example, the format, order, or semantics of provided in-context examples can greatly influence learning outcomes, as does the proportion of labels in the context and the exact labels used (Liang et al., 2022). Little is known, however, about how these factors interact (work from Wei et al., 2023; Yoo et al., 2022, suggests that they cannot be isolated); it is unclear which aspects are consistently beneficial, which vary across setups, and which are sensible to combine or decouple. The volatility of the paradigm warrants more research into the reliability of different design choices.

In this paper, we conduct a detailed exploration of vanilla and instruction-tuned LLMs across various shifts and setups to understand their robustness. We start with one of the prominent themes in robustness studies for TT models: robustness to spurious correlations between input and label distributions (Kavumba et al., 2019; McCoy et al., 2019; Niven and Kao, 2019) and find that in ICL, spurious correlations do not have a significant impact on learning outcomes.

We go on to investigate ICL's sensitivity to other features of adaptation context, as well as the consistency of predictions across different design choices. To do so, we conduct a large-scale grid search

Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL), pages 294–313
December 6–7, 2023. ©2023 Association for Computational Linguistics

across various combinations of factors and statistically analyse the results to shed light on the inter-dependencies of different design choices. We find that the exact in-context setup (the number of in-context examples, the distribution of in-context labels, or the type of instructions given in the context) has a surprisingly small but reliable impact on prediction outcomes. On the other hand, the type of instructions used to query the target has, by far, the most significant impact on model behaviour. It is also the most volatile across settings, making it the most pivotal factor.

2 Background and related work

In the following, we first briefly define TT and ICL and then cover known problems with model robustness.

2.1 Task tuning and spurious correlations

TT aligns a pre-trained model with a specific task by iteratively updating model parameters to minimise prediction loss on adaptation data. In our definition here, TT does not include finetuning on more abstract objectives like instruction tuning (IT; Wei et al., 2022). TT models often fit spurious correlations between inputs and associated labels that are idiosyncratic artefacts to the specific dataset (Niven and Kao, 2019; Kavumba et al., 2019; McCoy et al., 2019; Geva et al., 2019; Poliak et al., 2018; Gururangan et al., 2018; Kavumba et al., 2022) and do not align with the causal structure of the process that generated the data in 'the real world' (Schölkopf et al., 2012). Such adaptations (sometimes also referred to as 'shortcut solutions'; Geirhos et al., 2020) usually fail as soon as the data distribution shifts between the adaptation and test phase. Pre-training improves robustness compared to task training from scratch (Hendrycks et al., 2019, 2020). However, the necessary posthoc task adaptation still overfits spurious correlations (Niven and Kao, 2019). An effective way to mitigate issues in task adaptation is to expose the model to counterexamples of spurious correlations (Kaushik et al., 2020).

2.2 In-context learning

ICL describes the adaptation of a model to a task by inferring the task from the input given to the model. ICL can be subdivided into (1) few-shot learning, where in-context examples (consisting of input-output pairs) are given in the left-handed context of a tested input, and (2) zero-shot learning, referring to the case in which there are no examples. In this paper, we investigate few-shot scenarios.

In contrast to TT, ICL is a considerably cheaper adaptation method as it does not require any parameter updates. Akyürek et al. (2022) and Garg et al. (2022) show that adaptation of transformer models via ICL exhibits the same degree of expressivity as simple linear algorithms, small neural networks or decision trees. While ICL emerges spontaneously with increasing size of untuned LLMs (Brown et al., 2020), the ICL performance of such *'vanilla'* LLMs lags behind the tuned state-of-the-art on almost all common NLP benchmarks (Liang et al., 2022).

Previous research has also shown that ICL is highly unstable. For example, the order of in-context examples (Lu et al., 2022), the recency of certain labels in the context (Zhao et al., 2021) or the format of the prompt (Mishra et al., 2022) as well as the distribution of training examples and the label space (Min et al., 2022) strongly influence model performance. Curiously, whether the labels provided in the examples are *correct* is less important(Min et al., 2022). However, these findings are not uncontested: Yoo et al. (2022) paint a more differentiated picture, demonstrating that in-context input-label mapping *does* matter, but that it depends on other factors such as model size or instruction verbosity. Along a similar vein, Wei et al. (2023) show that in-context learners can acquire new semantically non-sensical mappings from in-context examples if presented in a specific setup.

From this listing, we see that ICL entails many design choices, that task-unrelated design choices change prediction outcomes and that the effects of design choices do not exist in isolation. The field is only beginning to understand the complex interplays of different prompting setups.

3 Experiment I: Robustness to spurious correlations

We clarify open questions about robustness of in-context learners by elucidating their sensitivity to factors to which they should be invariant (from here on *invariance factors*). First, we focus on one of the most prominent forms of non-robustness in TT models: susceptibility to spurious correlations between inputs and labels (see Section 2.1). In the first set of experiments, we test how differ-

Motivation					
Practical	*Cognitive*		*Intrinsic*		*Fairness*
□ △			□ △		
Generalisation type					
Compositional	*Structural*	*Cross Task*	*Cross Language*	*Cross Domain*	*Robustness*
		□			□ △
Shift type					
Covariate	*Label*		*Full*		*Assumed*
△					□
Shift source					
Naturally occuring	*Partitioned natural*		*Generated shift*		*Fully generated*
	□		△		
Shift locus					
Train–test	*Finetune train–test*		*Pretrain–train*		*Pretrain–test*
	△				□

Table 1: Our analyses, categorised according to the GenBench taxonomy (Hupkes et al., 2023). The token △ represents Experiment I and □ represents Experiment II.

ent models behave when spurious correlations are contained in their adaptation data.

3.1 Setup

We here describe the datasets and models used to test sensitivity to spurious correlations.

Task	Base dataset	Adversarial dataset
NLI	MNLI (Williams et al., 2018)	HANS (McCoy et al., 2019)
		ANLI (Nie et al., 2020)
PI	QQP (Wang et al., 2017)	PAWS (Zhang et al., 2019)
QA	SQuAD (Rajpurkar et al., 2016)	SQuAD adv. (Jia and Liang, 2017)
		adv. QA (Bartolo et al., 2020)
		SQuAD shifts (Miller et al., 2020)

Table 2: Tasks and corresponding datasets.

Datasets We use different common NLU datasets (from here on *base datasets*) which are known to contain spurious correlations between input and label distributions (Gururangan et al., 2018; Geva et al., 2019; Poliak et al., 2018), as well as *adversarial datasets* of the same tasks. Adversarial datasets are designed to not contain the spurious correlations of the base datasets; then, they can be used to test whether models use short-cut solutions (for an overview see Table 2). Our base datasets span three different types of NLU tasks: natural language inference (NLI), paraphrase identification (PI) and extractive question answering (QA). An overview can be found in Table 2 and additional details about dataset properties and their construction in Appendix C.

Models Our first experiment compares TT models with models that perform tasks through ICL. For the latter, we consider two types of models: '*vanilla*' LLMs, and LLMs that are tuned to follow instructions (*IT* see e.g. Wei et al., 2022; Zhong et al., 2021).

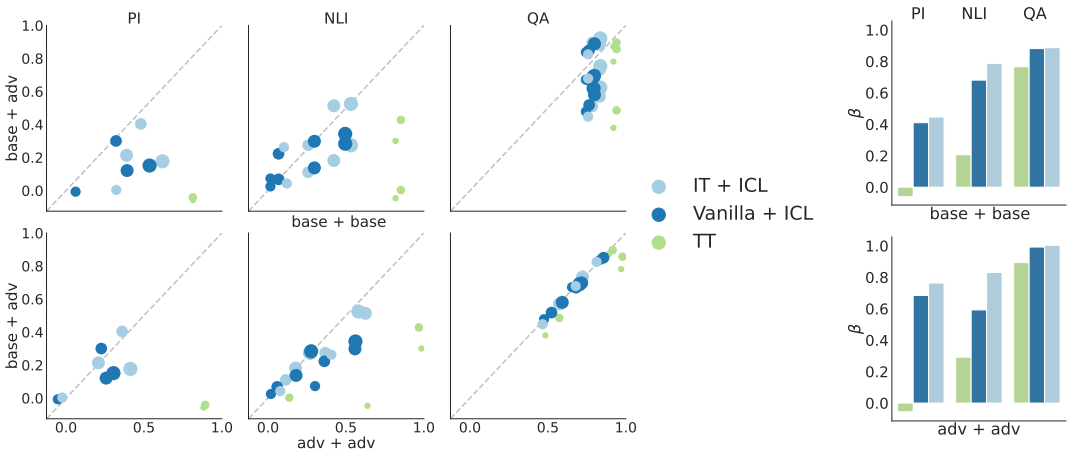
For TT, we use models based on RoBERTa$_{\text{BASE}}$ and RoBERTa$_{\text{LARGE}}$ (Liu et al., 2019). If available, we reutilise finetuned versions of RoBERTa that have been open-sourced through the huggingface hub (Wolf et al., 2019); if not available, we finetune the respective models ourselves (with training details in Appendix B).

Type of learning	Model
TT	RoBERTa-base
	RoBERTa-large
ICL + vanilla	LLaMA 7B, 13B, 30B, 65B
ICL + Instruction-tuning	Alpaca 7B, 13B, 30B, 65B

Table 3: Adaptation types and the respective models, as used in Section 3. We use the same ICL models in Section 4.

Our vanilla LLMs consist of the series of LLaMA models (7B, 13B, 33B, 65B; Touvron et al., 2023). The IT counterparts are the freely available Alpaca models, which are based on the same LLaMA models but are additionally finetuned via low-rank adaptation (LoRA; Hu et al., 2022) on the Alpaca self-instruct dataset (Taori et al., 2023; Wang et al., 2022). We run all models using mixed-precision decomposition as described by Dettmers et al. (2022). For an overview of all used models, see Table 3.

Evaluation We evaluate ICL models by concatenating the target example x with k labelled in-context examples and greedily decoding from the



(a) (b)

Figure 1: Figure (a) shows the f1-scores of different models – normalised for random accuracy – on different datasets when adapted via base or adversarial data. On each y-axis, we plot accuracy under distributional shift (*base + adv*) while on each x-axis there is no shift (*base + base* or *adv + adv*). Each column shows a different type of task. Marker size represents model size and colour represents the type of task adaptation. Dots close to the diagonal indicate invariance to the adaptation data and therefore robust generalisation, while dots in the bottom right indicate sensitivity to spurious correlations. Figure (b) shows the β-parameter of the linear regression (fixed intercept) on the data of Figure (a). We fit a linear regression for each task and adaptation type separately. Values close to 0 indicate very strong sensitivity to adaptation data, while values close to 1 indicate no sensitivity.

probability distribution over possible labels $y \in \mathcal{C}$ using $argmax_{y \in \mathcal{C}} P(y|x_1, y_1 ... x_k, y_k, x)$ where $\mathcal{C}$ is the set of possible labels. Every data point x is wrapped by an *instruction* template that explains the task the model should solve in natural language. The label space $\mathcal{C}$ is determined by the type of instruction template and can differ across templates. We mitigate the influences of potential confounds like the template format, the order of (x_i, y_i), imbalanced distribution of y_i or the semantics of x_i by a pseudo-random sampling x_i for every new model inference. Our sampling of x_i ensures that the in-context labels y_i are balanced over all possible labels (similar to Wei et al., 2023; Brown et al., 2020, inter alia). Moreover, we use multiple instruction templates sourced from FLAN (Wei et al., 2022) to avoid systematic bias.

3.2 Results

We first evaluate the capacity of different models to robustly generalise from adaptation data to test data. In the taxonomy of generalisation capabilities, this constitutes a *covariate shift* between the adaptation data (finetuning data in TT and in-context data in ICL) and the test data (compare GenBench; Hupkes et al., 2023). The corresponding GenBench evaluation card can be found in Table 1.

Base data in-context First, we adapt the TT and ICL models on the base data and then compare their performance between the base data and the respective adversarial counterparts. If an approach is robust to spurious correlations in the adaptation data (which are the fine-tuning data or in-context examples, respectively), it should perform approximately equally on the base dataset and the adversarial dataset. We relate both scores in the first row of Figure 1.

Results from in-context learners land generally closer to the diagonal, hence indicating – despite overall weaker performance – that they are more robust to the spurious correlations in their adaptation data. To quantify this visual result, we fit a linear regression model on the data presented in the scatterplot in Figure 1a (hence, predict the adversarial- from the base accuracies) with the intercept fixed at $\beta_0 = 0$. The coefficient β_1 can then be interpreted as a degree of robustness to the different adaptation data, with $\beta_1 = 1$ indicating complete robustness and $\beta_1 = 0$ complete reliance on non-generalisable patterns in the base data. The β_1 values for different adaptation types can be found in the top row of Figure 1b. The β_1 values across all tasks are significantly closer to the parity value of 1 for ICL models than for TT models, with IT models having

the edge over vanilla models.

Our results demonstrate that ICL models are much less sensitive to spurious correlations in their adaptation data than TT models. However, the fact that ICL models do not reach the parity value of 1 means that gains on adversarial data are smaller compared to gains on the base data. This suggests that ICL may still be mildly sensitive to spurious correlations, or, alternatively, that the adversarial datasets used are simply inherently more difficult, resulting in lower performances compared to the base data[1]. We will further explore this question in the next experiment.

Adversarial data in-context As a follow-up experiment, we consider what happens when the adaptation data contains adversarial examples. As those examples do not contain the same spurious correlations, models cannot overfit them (Kaushik et al., 2020). This should not make a difference for models that are robust to spurious correlations, but we expect a performance drop between these two conditions for models that learned solutions that exploited those correlations. As we are now evaluating the adversarial data points in both scenarios, we eliminate the potential impact of the dataset difficulty on the scores. In the second row of Figure 1, we plot performances with base adaptation examples in the context against the performance with adversarial adaptation data, noting that ICL models are mostly unaffected by adaptation data type while TT models land far underneath the diagonal again. A regression analysis shows almost all β-values of ICL models moving closer to parity, showing us how the dataset difficulty impacted the results. However, even without the effect of dataset difficulty on the β-values, they are still not quite equal to 1, suggesting that the type of adaptation data *has* a small influence on ICL learners.

4 Experiment II: Consistency evaluation in ICL

In the previous section, we saw that the robustness of in-context learners is likely influenced more by other factors than by spurious correlations in the in-context data. Although previous studies have reported the susceptibilities of LLMs to various factors, the impact of different design decisions and their interactions in the context of ICL robustness has not been systematically evaluated. Here, we test the effects of an extensive range of these factors on prediction outcomes in consistency and accuracy.

4.1 Experimental details

For all of the following experiments, we use promptsource templates (P3; Bach et al., 2022) and the ANLI dataset (Nie et al., 2020). We continue to use the models and the evaluation procedure from Section 3 (excluding the TT models). The following briefly describes the factors we consider in our analysis.

4.1.1 Factors

We distinguish two types of factors. Firstly, we consider factors that constitute interventions to improve consistency and performance, which we call **variance factors**[2] or λ_{var} for short. We expect a model to *change* their response when we change the value of those factors:

Size We consider models with 7B, 13B, 30B and 65B learnable parameters.

Instruction tuning Whether models are instruction-tuned or not ('vanilla' models).

Calibration Whether model outputs are calibrated using 'content-free prompts' following Zhao et al. (2021).

n-shots Whether there are many (k = 5) or few (k = 2) in-context examples in the prompt.

Instruction quality Whether instructions belong to one of two groups of semantically equivalent but *differently performing* instruction templates (high- vs. low-performing; more details in Section 4.1.4).

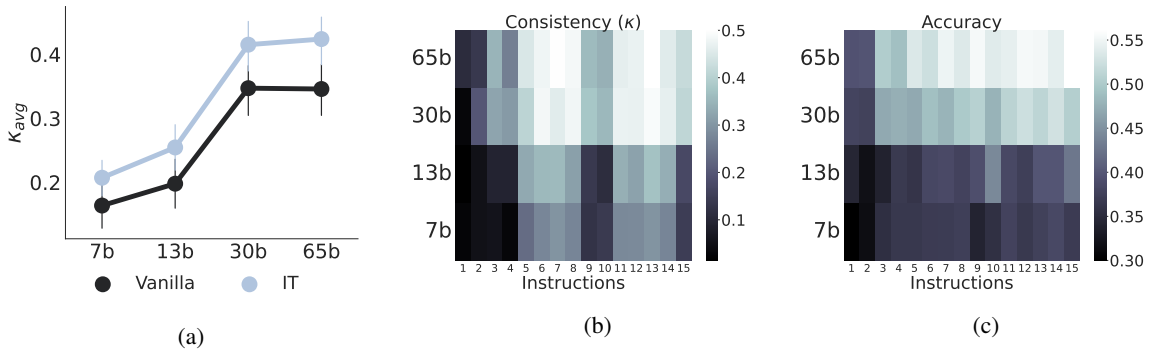
Balanced labels Whether examples with labels are balanced across all possible classes in the context or use randomly sampled examples.

Secondly, we consider factors from which we want a model to *not change* their response (or 'be robust to') when we change their value. We will call these **invariance factors** or λ_{inv}:

Cross-templates Whether in-context instructions are drawn randomly from all available instruction templates or use the same instructions as for the target.

[1] An illustrative example of the base data being easier: adversarial QA contains only a single answer alternative while squad contains three.

[2] For detailed explanations on the different factors, we refer to Appendix F.



(a) (b) (c)

Figure 2: Figure (a) shows the consistency of a model when used with all 15 different P3 instructions, in an otherwise fixed setup. A value of 1 indicates perfect agreement (all templates produce the same prediction); Figure (b) shows how consistent individual instructions are with all other instructions. A value of 0 indicates a complete change of predictions while a value of 1 indicates perfect agreement; Figure (c) shows the respective accuracies of the instructions in Figure (b).

Cross-task Whether another classification task (QQP) is used as in-context examples or the same task as the target task (ANLI) is used.

Instructions Different semantically equivalent target instructions that *perform similarly* (more details in Section 4.1.4).

One label Whether in-context examples have only a *single* randomly selected label or diverse labels.

Combining the above factors results in 1536 setups. We evaluate each of these constellations using the same subset of 600 data points[3] that we draw uniformly from either of the ANLI validation sets. In-context examples are drawn at random from the respective training sets.

4.1.2 Analysis methods

Our analysis entails two steps:

1. Main effects: how much does a single factor impact consistency and the accuracy across many setups?

2. Interactions: when we disentangle the main effects, do we find systematic interactions across pairs or triplets of factors?

Main effects To evaluate the main effect of each factor λ, we employ linear regression to predict the accuracy of a model based on λ, considering all possible combinations of the remaining factors. The regression model is formulated as $Acc = \beta_1\lambda + \beta_0$. The coefficient β_1 represents the main effect of a specific λ, approximating the average change in accuracy across all possible setups given λ. We also fit the intercept β_0, but won't interpret it.

Interactions We analyse interactions by fitting a factorial ANOVA considering the effect of all possible 2- and 3-way interactions[4] of factors on the accuracy of predictions. We then count the number of significant interactions every factor maintains with other factors. A larger number of interactions suggests that a factor is volatile, i.e. it changes the predictions depending on the overall setup. Further, as the factors have been chosen to be orthogonal and should not influence each other. On the other hand, if factors are not interacting, we can interpret their main effects directly.

4.1.3 Consistency metrics

We measure the consistency of model predictions using Cohen's κ (Cohen, 1960), a measure of inter-rater agreement adjusted for agreement by chance. The metric κ equals 1 if two (or more) sets of predictions perfectly align while agreement by chance results in κ equalling 0. In our case, we calculate κ to compare the predictions of a model before and after we change the value of a factor λ (e.g. if all labels in-context are the same or if they are not; see `One label`) across all possible setups. We make the metric less dependent on the accuracy of a model by calculating κ only on the subset of predictions that have been correctly predicted in either of the two cases.

[3]We found 600 examples to yield sufficiently similar results to evaluating the whole dataset, tested on a small subset of setups.

[4]We exclude the *instructions* factor because the independence of *instruction quality* is not given. Moreover, we adapt the significance levels via Bonferroni correction for multiple comparisons ($\alpha < 0,00059$) and show only significant interactions.

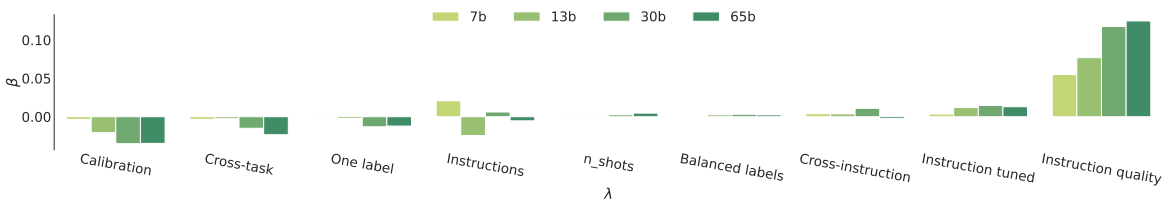


Figure 3: The β-values of the main effects of each individual factor across many different runs. The values can be directly interpreted as '*expected accuracy gain/loss*' when a factor is present compared to when it is absent.

4.1.4 Probing instructions

To find a set of high- and low-performing instructions for the `instruction quality` factor, we run a preliminary analysis where we probe model behaviour in response to all 15 available P3 ANLI instructions. We assess the performance of different instructions based on accuracy and consistency.

We first get a general picture of each model's average consistency κ_{avg} across all templates. We find that κ_{avg} increases with the number of parameters and is overall higher when a model has been instruction tuned (Figure 2a).

We then consider the consistency of each individual instruction and find a congruent pattern of consistency across all models (Figure 2b) that corresponds generally to the accuracy scores of the same instructions (compare Figure 2c). Interestingly, we also find two groups of high-accuracy instructions making very different predictions (see the consistency scores of 9, 10 and 15 vs. rest). Based on these observations, we choose the two highest- and lowest-performing instructions to constitute the `instruction quality` factor and templates 14 and 15 as realisations of the `instructions` factor.

4.2 Results

We evaluate the models on all possible combinations of λ_{var} and λ_{inv}. Appendix G shows the distribution of accuracy scores across all runs for different models. The wide spread of scores is striking: large models score from below chance to up to 67% accuracy, depending on the overall setup. This extreme variability underlines the importance of better understanding the impact of different design decisions and prediction consistency in ICL. The subsequent section comprehensively summarises the results of our statistical analysis.

4.2.1 Main effects

The main effects separated by model size are shown in Figure 3, illustrating each factor's impact in isolation.

Variance factors The variance factors we chose are generally thought to improve accuracy and, hence, should have positive main effects. We find two out of five *variance factors* significantly improve performance on average, from which `instruction quality` stands out as the most influential factor across all model sizes. Similarly, we find that `instruction tuning` is consistently beneficial while `balancing` the in-context labels and the number of in-context examples (`n-shots`) have on average positive but small and non-significant effects. Surprisingly, `calibration` harms rather than helps performance for all but our smallest model.

Invariance factors Different from variance factors, invariance factors are chosen such that they should not influence a robust model's predictions. Accordingly, the main effects should be optimally close to 0. We find that models are generally robust to having varied instructions in-context (`cross-instruction`), or even having a slightly positive effect. This is intriguing, as this factor entails considerable changes to the in-context setup, and we previously saw how the type of *target instructions* (in `instruction quality`) plays a major role. Further, we identify vulnerabilities of large models to the factors `cross-task` and `one label`. The ambivalent effect of the `instructions` factor suggests high volatility across similarly performing instructions (i.e. different instructions perform differently for different models and setups).

These main effects give us a general idea of the tendencies of factors. To better understand all main effects, we will investigate interactions in Section 4.2.1.

Consistency of invariance factors Additionally to a factor's impact on accuracy, we also compute the prediction consistency κ of the factors (as defined in Section 4.1.3). To do so, we calculate the agreement of predictions when a factor is present

with when it is absent. This way, the value of κ shows us the degree of robustness of a model to an invariance factor by quantifying the degree of prediction change caused by that factor. Figure 4 shows how robustness increases with size and instruction tuning. The very low κ scores for the detrimental `cross-task` factor come as no surprise, while low scores in the `instructions` factor corroborate the previous suspicion that instructions are highly volatile: if we change the type of used `instructions`, the predictions across a lot of setups change.

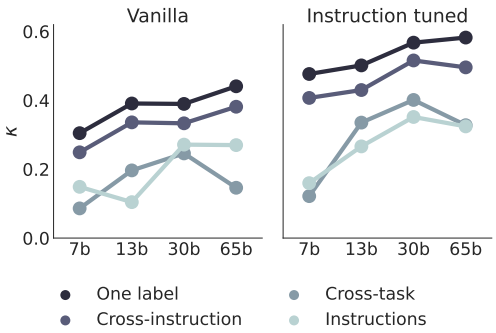


Figure 4: The consistency values when a specific factor is present or not across all other setups. A value of 0 indicates a complete change of predictions while a value of 1 indicates perfect agreement (i.e. a low value indicates that a model is not robust to a change in a specific factor).

4.2.2 Interactions

The main effects give us a good idea of the general direction of the impact of a single factor. However, the main effects do not tell the whole story: consider the case in which factor `A` improves performance if it is paired with factor `B`, but performance deteriorates when paired with `C`. `A`'s overall main effect might be close to zero even though it influences certain settings. To better understand the impact of each factor, we will have to investigate its interactions.

We determine interactions following the procedure described in Section 4.1.2. Figure 5 shows the number of interactions that each factor maintains. A general observation is that large models tend to have simpler 2-way interactions, while smaller models tend to have more complex 3-way interactions.

Highly interactive factors The most important factor of `instruction quality` maintains many interactions. Hence, many other factors change predictions depending on the used instruction template. We find a similar effect for the `instructions` factor[5]. This demonstrates the intricacy of the formulation of instructions: the `instruction quality` has the largest positive impact on prediction outcomes, but at the same time, the instructions are highly interactive and volatile, with their the effects of many other factors depending on it. Otherwise, we observe that `calibration` is the most volatile, with eight significant interactions with other factors. The previously observed main effect has to be seen in this perspective: `calibration` is not generally detrimental, but its effects depend very much on the setup in which it is used. For example, we find on closer inspection that `calibration` leads to the highest overall accuracies for the 7B parameter models when presented with specific `instructions` and paraphrase identification in-context examples (`cross-task`).

Low interactive factors On the other end of the spectrum, we find that factors like the number of in-context examples (`n-shots`), the `balancing` of in-context labels or using just `one label` have little to no interactions at all. Conveniently, there are no ambiguities for these factors and we can therefore interpret their main effects directly, as they are most likely to be stable across setups. For example, suppose it is possible to increase the number of examples in the context. In that case, we can reliably expect small gains in accuracy without the danger of otherwise interfering with the learning process. Similarly, balancing labels leads to reliable small improvements and having just a single label in the context reliably reduces accuracy for large models.

5 Discussion

We will first summarise the findings of this paper and then discuss their implications.

Findings We saw in Section 3 how spurious correlations do not influence predictions in ICL in a relevant manner as they did previously in TT. This, however, does not resolve the problem of robustness: depending on the setup, ICL accuracy in our experiments differs up to 40%, as other factors in the setup become pivotally important. We here conducted a comprehensive analysis of the influence

[5]We fit another ANOVA excluding `instruction quality` while keeping `instructions` as a factor to ensure that the effect is not only due to large performance differences between the two realisations of instruction quality. We find similarly strong interactions for the `instructions` factor (see Appendix H).

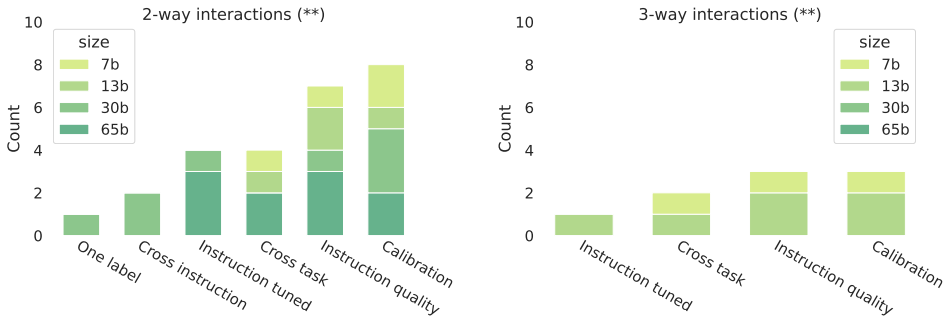


Figure 5: The number of interactions per factor with other factors. A large number of interactions means that the outcome of a change in these factors depends on a lot of other variables.

of different setups on the consistency of predictions in ICL models. Considering different setups, well-chosen `instructions` promise the largest performance gains across many setups. At the same time, they are among the most volatile factors of all and highly sensitive to the setting in which they are used. On the other hand, factors that relate to the exact organisation of the in-context examples, such as the label distribution or in-context instructions (`cross-instructions`), have surprisingly small impacts. Other factors like `n-shots` – among others – are not interactive, which makes them much easier to handle: their expected gain or loss should, in most cases, correspond to our observed main effects. Across all of our experiments, we also find the general tendency that larger numbers of model parameters and instruction tuning are beneficial for model consistency across many settings.

Implications and future research What do these findings imply? As we have seen, inconsistency is a severe concern in ICL, and we here contribute to narrowing down its sources. Unlike previously in TT, concentrating on spurious correlations is not vital for ICL robustness and investigating design choices concerned with in-context examples (i.e. the exact few-shot setting) promises to be less impactful or mostly dependent on other setup factors. Instead, our findings suggest that the exact phrasing of instruction templates is pivotally important. To get hold of inconsistent predictions in ICL, finding the exact properties of instructions that so strongly influence model predictions is a sensible next step (potentially with a similar methodology as it is presented here). Insights into the impact of instruction properties can help us to find the source of inconsistencies and avoid them in production, while they can also contribute to the theoretical understanding of in-context learning which is currently still under investigation. While our analysis focused on the few-shot setting, it also significantly impacts the increasingly popular zero-shot learning, as instructions are central in that setting. For model deployment, our findings demand caution as minor changes to certain parts of prompts (e.g. the instructions) can change the performance of the general setup. This is especially true for employing smaller, untuned models. A consistent finding across all our experiments is that instruction tuning improves consistency and robustness to irrelevant factors across all setups. Therefore, we advocate for the use of tuned models to improve robustness. Finally, recent research has suggested that dynamics in ICL are, to a certain degree, chaotic (Khashabi et al., 2022). It might be advised to use more diverse evaluation setups and a rigorous statistical analysis of the results to guarantee the generality of results and avoid Type-I errors in publications (Ioannidis, 2005).

6 Conclusion

We here analysed robustness and variability in the recent learning paradigm of ICL, showing that they are generally different from in task-tuning. By using a methodology that covers a wide range of potential prompt design decisions, we show which factors actually matter in prompt design and how these factors influence each other.

Limitations and Acknowledgements

For a discussion of the limitations of our work and the acknowledgements, we refer to Appendix I and Appendix J, respectively.

References

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. 2022. What learning algorithm is in-context learning? investigations with linear models. *ArXiv preprint*, abs/2211.15661.

Anas Awadalla, Mitchell Wortsman, Gabriel Ilharco, Sewon Min, Ian Magnusson, Hannaneh Hajishirzi, and Ludwig Schmidt. 2022. Exploring the landscape of distributional robustness for question answering models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5971–5987, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Stephen 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 Fries, Maged Alshaibani, Shanya Sharma, Urmish Thakker, Khalid Almubarak, Xiangru Tang, Dragomir Radev, Mike Tian-jian Jiang, and Alexander Rush. 2022. PromptSource: An integrated development environment and repository for natural language prompts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 93–104, Dublin, Ireland. Association for Computational Linguistics.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *ArXiv preprint*, abs/2302.04023.

Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. 2020. Beat the AI: Investigating adversarial human annotation for reading comprehension. *Transactions of the Association for Computational Linguistics*, 8:662–678.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher 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 Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. 2022. Llm. int8 (): 8-bit matrix multiplication for transformers at scale. *ArXiv preprint*, abs/2208.07339.

Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. 2022. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1161–1166, Hong Kong, China. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. 2019. Using pre-training can improve model robustness and uncertainty. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2712–2721. PMLR.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2744–2751, Online. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

D. Hupkes, M. Giulianelli, V. Dankers, et al. 2023. A taxonomy and review of generalization research in nlp. *Nature Machine Intelligence*, 5:1161–1174.

John PA Ioannidis. 2005. Why most published research findings are false. *PLoS medicine*, 2(8):e124.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Divyansh Kaushik, Eduard H. Hovy, and Zachary Chase Lipton. 2020. Learning the difference that makes A difference with counterfactually-augmented data. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Pride Kavumba, Naoya Inoue, Benjamin Heinzerling, Keshav Singh, Paul Reisert, and Kentaro Inui. 2019. When choosing plausible alternatives, clever hans can be clever. In *Proceedings of the First Workshop on Commonsense Inference in Natural Language Processing*, pages 33–42, Hong Kong, China. Association for Computational Linguistics.

Pride Kavumba, Ryo Takahashi, and Yusuke Oda. 2022. Are prompt-based models clueless? In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2333–2352, Dublin, Ireland. Association for Computational Linguistics.

Daniel Khashabi, Xinxi Lyu, Sewon Min, Lianhui Qin, Kyle Richardson, Sean Welleck, Hannaneh Hajishirzi, Tushar Khot, Ashish Sabharwal, Sameer Singh, et al. 2022. Prompt waywardness: The curious case of discretized interpretation of continuous prompts. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3631–3643.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *ArXiv preprint*, abs/2211.09110.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv preprint*, abs/1907.11692.

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 Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

John Miller, Karl Krauth, Benjamin Recht, and Ludwig Schmidt. 2020. The effect of natural distribution shift on question answering models. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6905–6916. PMLR.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11048–11064, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. 2022. Reframing instructional prompts to GPTk's language. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 589–612, Dublin, Ireland. Association for Computational Linguistics.

Marius Mosbach, Tiago Pimentel, Shauli Ravfogel, Dietrich Klakow, and Yanai Elazar. 2023. Few-shot fine-tuning vs. in-context learning: A fair comparison and evaluation. *ArXiv preprint*, abs/2305.16938.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4658–4664, Florence, Italy. Association for Computational Linguistics.

Xenia Ohmer, Elia Bruni, and Dieuwke Hupkes. 2023. Evaluating task understanding through multilingual consistency: A chatgpt case study. *ArXiv preprint*, abs/2305.11662.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris M. Mooij. 2012. On causal and anticausal learning. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress.

Chenglei Si, Zhe Gan, Zhengyuan Yang, Shuohang Wang, Jianfeng Wang, Jordan Boyd-Graber, and Lijuan Wang. 2023. Prompting gpt-3 to be reliable.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *ArXiv preprint*, abs/2302.13971.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions. *ArXiv preprint*, abs/2212.10560.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 4144–4150. ijcai.org.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, et al. 2023. Larger language models do in-context learning differently. *ArXiv preprint*, abs/2303.03846.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv preprint*, abs/1910.03771.

Kang Min Yoo, Junyeob Kim, Hyuhng Joon Kim, Hyunsoo Cho, Hwiyeol Jo, Sang-Woo Lee, Sang-goo Lee, and Taeuk Kim. 2022. Ground-truth labels matter: A deeper look into input-label demonstrations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2422–2437, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. 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 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

Ruiqi Zhong, Kristy Lee, Zheng Zhang, and Dan Klein. 2021. Adapting language models for zero-shot learning by meta-tuning on dataset and prompt collections. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2856–2878, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment.

A Experiment 1: List of TT models

We compare the sensitivity to spurious correlations of ICL models with TT models. The following table contains all TT models we used during these experiments, providing the respective handle for the huggingface hub or indicating with 'own' that we fine-tuned the respective model ourselves.

		Models	
		$\text{RoBERTa}_{\text{BASE}}$	$\text{RoBERTa}_{\text{LARGE}}$
Base datasets	MNLI	textattack/roberta-base-MNLI	roberta-large-mnli
	SQuAD	deepset/roberta-base-squad2	deepset/roberta-large-squad2
	QQP	own	own
Adv. datasets	HANS	own	own
	ANLI	own	own
	PAWS	own	own
	SQuAD adversarial	own	own
	adversarial QA	own	own
	SQuAD shifts	own	own

B Experiment 1: Finetuning details of own models

We finetuned all RoBERTa models using the same set of hyperparameters, based on the literature and experience.

Hyperparameters We train using the ADAM Optimizer with γ = 1e-05, inverse square root decay and $\beta_{1/2}$ = (0.9, 0.999), no weight decay, 250 warmup steps and a batch size of 8. We stop training if the model does not show improvement on the validation set for 1 epoch of training.

Data For adversarially tuned models, we mixed the training set of the base data with 70% of the adversarial data (30% retained for evaluation). We ensured a mixing ratio of 20%/80% adversarial/base data.

C Experiment 1: Datasets details

We here provide additional information about the datasets we use in Experiment 1:

C.1 Base datasets

MNLI (Multi Natural Language inference; Williams et al. 2018)
A large-scale natural language inference dataset. It contains sentence pairs annotated with three categories: entailment, contradiction, and neutral. The dataset is sourced from a variety of genres, like fiction, government documents, and telephone conversations, thus encouraging models to learn domain-agnostic representations.

QQP (Quora Question Pairs; Wang et al. 2017)
A collection of question pairs from the Quora platform, labelled as either duplicates or non-duplicates. The aim is to identify semantically equivalent questions, addressing challenges such as paraphrasing and varying levels of detail.

SQuAD (Stanford Question Answering Dataset; Rajpurkar et al. 2016)
A reading comprehension dataset consisting of questions about passages from Wikipedia. The questions are human-annotated, and the answer to each question is a segment (or span) of the passage. The goal of models is to identify and extract the correct span from the passage that answers the question.

C.2 Adversarial datasets

HANS (Heuristic Analysis for NLI Systems; McCoy et al. 2019)
Constructed to evaluate models on non-entailment cases that appear entailed due to spurious biases. Built upon common NLI datasets like SNLI and MultiNLI, it dissects three heuristic strategies that a model might utilise: lexical overlap, subsequence, and syntactic structure.

ANLI (Adversarial Natural Language Inference; Nie et al. 2020)
Generated by first training models on existing datasets (e.g., SNLI and MultiNLI) and then having human annotators produce examples that the models predict incorrectly. Generation of additional examples was done in multiple rounds with respectively improved models, accordingly each round increases the adversarial difficulty.

PAWS (Paraphrase Adversaries from Word Scrambling; Zhang et al. 2019)
Comprises sentence pairs with high lexical overlap but differing semantics, challenging models that heavily weigh word overlap. An adversarial expansion to datasets like the Quora Question Pairs dataset (QQP).

SQuAD Adversarial (Jia and Liang, 2017)
A derivative of the Stanford Question Answering Dataset (SQuAD) where adversarial sentences are introduced into the context paragraphs, aiming to mislead models into selecting incorrect answers while the correct answers remain unchanged.

Adversarial QA (Bartolo et al., 2020)
A reading comprehension dataset, where each question is tied to a Wikipedia passage. Distinctively, answer annotations are freeform human responses rather than extracts from the passage, testing the extractive capability boundaries of SQuAD-inspired models.

SQuAD Shifts (Miller et al., 2020)
Formed by perturbing the original SQuAD distribution in terms of linguistic and stylistic attributes. This dataset gauges model robustness against unseen data distributions, such as domain shifts or synthetic noise.

D Experiment 1: Impact of spurious correlations in ICL

We conducted an additional analysis of the results in Section 3.2. The goal of this additional analysis is to understand the impact of the type of adaptation data (adversarial vs. base) on the prediction outcomes in comparison with *other* factors that we varied in our experiments (such as the type of `instruction template`, whether the model was `instruction tuned` or the `size` of the model). `Type data` is a binary factor indicating whether the model was adapted on base or adversarial data; `Size` is a quarternary factor indicating model size; `Type instructions` is a binary factor indicating the type of template that was used; `Instruction tuned` is a binary factor indicating whether the tested model was instruction tuned or not.

Table 4 shows the summary statistics of an ANOVA that we apply to these factors and their impact on the model accuracy. We can see from Table 4 that adaptation data is the only factor that does not significantly impact prediction outcomes.

	df	sum_sq	mean_sq	F	PR(>F)
Type data	1.0	8.67	8.67	0.12	0.72
Size	3.0	6626.73	2208.91	31.26	5.71e-18
Type instruction	1.0	95.32	95.32	1.34	0.024
Instruction tuned	1.0	900.55	900.55	12.74	4.05e-04
Residual	357.0	25220.11	70.64	NaN	NaN

Table 4: Results of ANOVA

E Experiment 1 & 2:Prompt template examples

E.1 FLAN instructions

Input:

Does the Hypothesis in the input entail (True) or contradict (False) the Premise or is it independent (Neither)?
Premise: Kirklees Stadium (known as the John Smith's Stadium due to sponsorship), is a multi-use sports stadium in Huddersfield in West Yorkshire, England. Since 1994, it has been the home ground of football club Huddersfield Town and rugby league side Huddersfield Giants, both of whom moved from Leeds Road.
Hypothesis: Kirklees Stadium is in Scotland.

OPTIONS:
- True
- Neither
- False

ANSWER: False.

[...]

Does the Hypothesis in the input entail (True) or contradict (False) the Premise or is it independent (Neither)?
Premise: Jonathan Smith (born January 17, 1971), better known by his stage name Lil Jon, is an American rapper, record producer, and DJ. He was the frontman of the group Lil Jon & The East Side Boyz, which he formed in 1997, and they released several albums until 2004.
Hypothesis: Jonathan Smith spent much of his time in China.

OPTIONS:
- True
- Neither
- False

ANSWER:

Target:

Neither.

E.2 P3 details

In the following, we provide more details on the instruction templates (Bach et al., 2022), as used in Experiments II.

E.2.1 P3 details – Names

Names of all available P3-instructions, following the ordering of Figure 2:

1. 'MNLI Crowdsource'
2. 'Guaranteed Possible Impossible'
3. 'Always Sometimes Never'
4. 'Consider Always Sometimes Never'
5. 'Does This Imply'
6. 'Guaranteed True'
7. 'GPT 3 Style'
8. 'Take the Following as Truth'
9. 'Must Be True'
10. 'Based on the Previous Passage'
11. 'Should Assume'
12. 'Can We Infer'
13. 'Justified in Saying'
14. 'Does It Follow That'
15. 'Claim True False Inconclusive'

E.2.2 P3 details – Examples

We here show examples of P3 prompt templates as they are used in Experiment 2: The prompt templates wrap the respective ANLI data point and provide natural language instructions about the task to the model.

High-performing templates 'Claim true false inconclusive'

[...]

Jonathan Smith (born January 17, 1971), better known by his stage name Lil Jon, is an American rapper, record producer, and DJ. He was the frontman of the group Lil Jon & The East Side Boyz, which he formed in 1997, and they released several albums until 2004. Based on that information, is the claim: "Jonathan Smith spent much of his time in China." true, false, or inconclusive?

ANSWER:

High-performing templates 'Does it follow that'

[...]
Given that Jonathan Smith (born January 17, 1971), better known by his stage name Lil Jon, is an American rapper, record producer, and DJ. He was the frontman of the group Lil Jon & The East Side Boyz, which he formed in 1997, and they released several albums until 2004. Does it follow that Jonathan Smith spent much of his time in China. Yes, no, or maybe?

ANSWER:

Low-performing templates 'MNLI crowdsource'

[...]
Jonathan Smith (born January 17, 1971), better known by his stage name Lil Jon, is an American rapper, record producer, and DJ. He was the frontman of the group Lil Jon & The East Side Boyz, which he formed in 1997, and they released several albums until 2004. Using only the above description and what you know about the world, "Jonathan Smith spent much of his time in China." is definitely correct, incorrect, or inconclusive?
ANSWER:

Low-performing templates 'Guaranteed possible impossible'

[...]
Assume it is true that Jonathan Smith (born January 17, 1971), better known by his stage name Lil Jon, is an American rapper, record producer, and DJ. He was the frontman of the group Lil Jon & The East Side Boyz, which he formed in 1997, and they released several albums until 2004.

Therefore, "Jonathan Smith spent much of his time in China." is guaranteed, possible, or impossible?

ANSWER:

F Experiment 2: Factors details

In the following, we provide a more detailed description of the factors used in Section 4 and also provide our motivation to include these factors.

F.1 Invariance factors

Size We consider models of different sizes. Model size has been shown to be an important moderating factor in probably all previous studies on in-context learning.

Instruction tuning We have seen previously that instruction tuning improves the consistency of a model across templates (see Section 4.1.4). We introduce it as a factor to show which other invariance factors it may affect.

Calibration Previous research has shown how small models are especially biased towards single labels when prompted. We find similar tendencies for our model: We exploratively calculate the entropy of a model's predictions across all data points in a dataset. This allows us to estimate whether a model is biased toward predicting a single label (low entropy). Optimally, a model's prediction should be close to the entropy of the target distribution $\mathcal{H}(Y)$. We find that smaller models have a larger bias towards predicting a single label (lower prediction entropy), while larger and IT models get closer to $\mathcal{H}(Y)$ (see Figure 6).

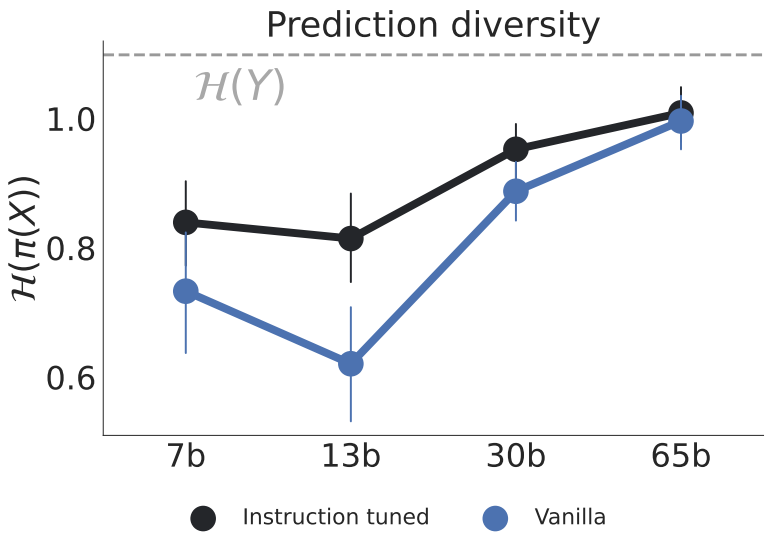


Figure 6

Zhao et al. (2021) suggests solving this issue by calibrating the model probabilities using 'content-free' prompts. We add the factor of calibration to assess its effects systematically.

n-shots The number of in-context examples has been shown to interact with other factors (e.g. according to Zhao et al., 2021, `calibration` has a more significant effect for fewer in-context examples). We would also expect that `n-shots` interacts with many other in-context factors such as `one label`, in which we show the model just examples with the same label in-context, is modulated by the number of in-context examples. We introduce 'few' ($k = 2$) and 'many' ($k = 5$) examples as a factor.

Instruction quality Ultimately, we have seen how some instructions produce consistent and relatively well-performing responses across different models while others do not (see Section 4.1.4. We add this last factor to see which other types of factors help the in-context learner cope with varying `instruction quality`. We chose the two best and two worst-performing templates[6] from our previous analysis.

[6]See Appendix E for an example of the instructions

F.2 Invariance factors

The following briefly describes each of the tested λ_{inv}.

Balanced labels Zhao et al. (2021) additionally showed how a majority label among the in-context example can influence the distribution of model outputs. Therefore, we compare contexts with balanced in-context label distribution with randomly sampled labels and an extreme case with only a single in-context label.

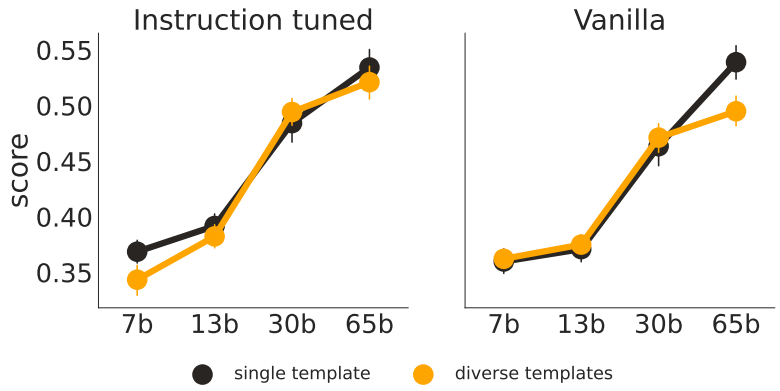


Figure 7

Cross-instruction We include `cross-templates` as a factor to assess model robustness to shifts in label space and surface form of instruction formulation. Previous research has shown how in-context learners are sensitive to the instructions (Mishra et al., 2022) as well as the label distribution $\mathcal{C}$ (Min et al., 2022). The experiments of Min et al. (2022) represent an extreme case in which $\mathcal{C}$ is resampled to be random tokens. While these edge cases are theoretically attractive, we here change this scenario to a practically common one, where instructions and labels are semantically equivalent but have different surface forms by randomly sampling from the available p3 instructions for the in-context examples. We test the impact of in-context instructions in a single setting with results shown in Figure 7 Surprisingly, almost all models are robust to semantic-invariant changes to instructions of the in-context examples despite changes in the label space and substantial changes in surface form and format across different instructions.

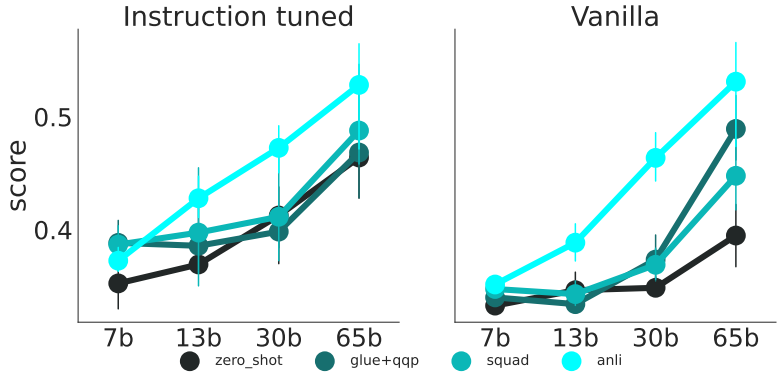


Figure 8: Accuracy scores of all models in all possible setups, with vanilla models on the left and instruction-tuned models on the right.

Cross-task In `cross-task`, we exchange the task of the in-context examples such that the only consistency between in-context and target examples is the general format (x followed by y) and the truthfulness

of the x to y mapping. To see whether conditioning on a fixed label space matters, we add tasks with a discriminative (QQP) and a generative (SQuAD) objective as different factors. Compared to a zero-shot baseline, we can see that large models can benefit from conditioning on other tasks (Figure 8). For our principal analysis, we only include QQP as an in-context task, as SQuAD is incompatible with many other factors (such as `balanced labels`, `one label` aso...)

Instructions Besides the quality of the instructions, we are also interested in how consistent model behaviour is across `instructions` that are of similar quality. To get an insight into this, we bin the high-quality instructions respectively into a new factor.

G Experiment 2: Accuracy distribution

We here show the distribution of accuracy scores for all setups in experiment 2, separated by model size (hue) and whether the model is instruction tuned or not (i.e. vanilla).

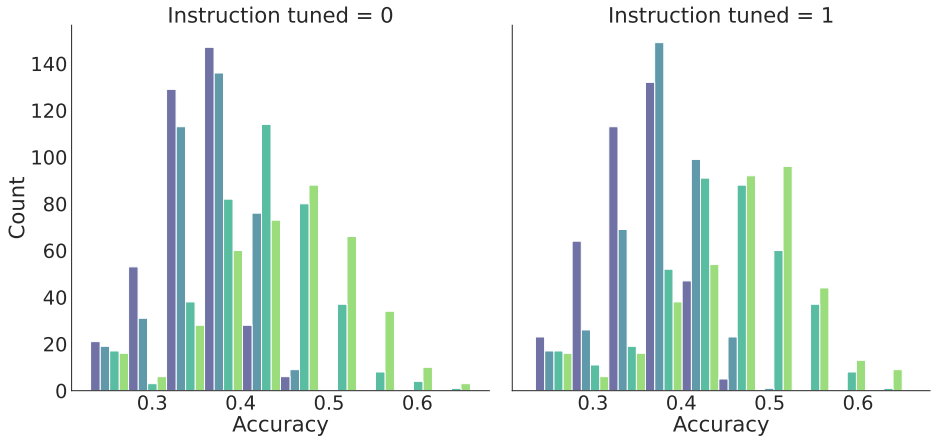


Figure 9

H Experiment 2: Interactions details

H.1 ANOVA using `instructions` factor

We fit an ANOVA using the factor `instructions` instead of `instruction quality`. In that case, we find a similar pattern of interactions, showing that the size of the main effect can not merely explain the number of interactions.

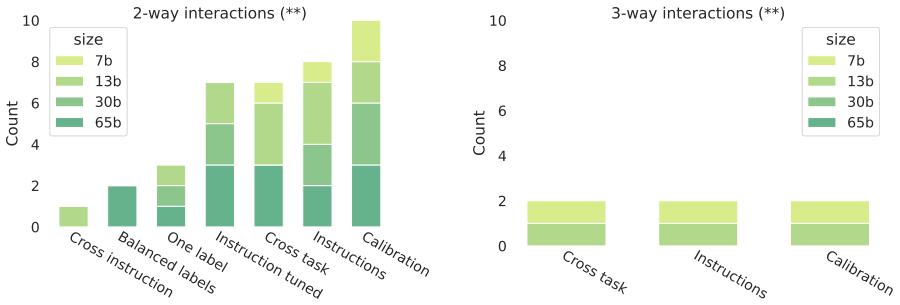


Figure 10: Interactions when excluding `Instruction quality` and keeping `Instructions` instead. We find similar patterns.

H.2 Interaction mappings and effect sizes

The following shows the exact mapping of the interacting factors as well as the size of the corresponding effect size, measured by $\beta_{\lambda_1 \times \lambda_2}$ values from a post hoc regression analysis.

Figure 11: The exact mappings of all 2-way interactions in our experiments.

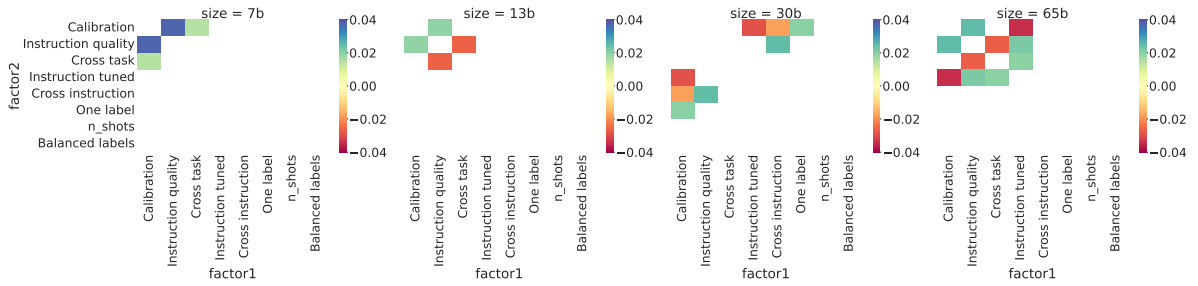


Table 5: The exact mappings of all 3-way interactions in our experiments.

Model	λ_1	λ_2	λ_3	$\beta_{\lambda_1 \times \lambda_2 \times \lambda_3}$
7B	Instruction quality	Calibration	Cross task	0.037106
13B	Instruction tuned	Calibration	Instruction quality	0.002102
13B	Instruction quality	Cross task	Calibration	-0.013176

I Limitations

For the first set of experiments in Section 3, the comparison between TT models and ICL is not 'fair'. Model sizes are not comparable, the amount of adaptation data differs significantly (thousand for task-tuning compared to 5 for ICL) and some of the adversarial datasets were created with some of the TT models 'in-the-loop' (e.g. ANLI). However, our motivation here is not to be fair, but to show practically relevant effects in either type of task adaptation. For a fair comparison, see Mosbach et al. (2023).

For the second set of experiments in Section 4, we only consider a subset of factors that we deemed the most relevant or interesting. Adding more factors would enrich the analysis. However, the number of model inferences to compute grows exponentially with the number of considered factors, which sets soft limits for the number of analysed factors. For potential follow-ups, we suggest a more fine-grained investigation of different instruction designs for the target example, as this potentially yields exciting insights on what exactly leads to the large performance gains and high volatility. Our study is coarse in this aspect.

Our analysis would have been more expressive if we chose an 'easier' task than the relatively 'hard' ANLI dataset to run our evaluation: our smaller models perform relatively poorly across many factors on challenging datasets like ANLI and provide less variance for a meaningful analysis.

J Acknowledgements

We would like to thank the anonymous reviewers for their time and productive feedback. Beyond that, we thank the whole COLT group at UPF for their helpful feedback and other support during the project – especially Emily Cheng and Xixian Liao. Further, LW thanks the Department of Translation and Language Sciences at the University Pompeu Fabra for funding.