The Impact of Demonstrations on Multilingual In-Context Learning: A Multidimensional Analysis

Miaoran Zhang[1]   Vagrant Gautam[1]   Mingyang Wang[2,3,4]   Jesujoba O. Alabi[1]

Xiaoyu Shen[5]* **Dietrich Klakow[1]   Marius Mosbach[6]**

[1]Saarland University, Saarland Informatic Campus

[2]Bosch Center for AI [3]LMU Munich [4]Munich Center for Machine Learning (MCML)

[5]Eastern Institute of Technology, Ningbo   [6]Mila, McGill University

{mzhang,vgautam,jalabi,dietrich.klakow}@lsv.uni-saarland.de

mingyang@cis.lmu.de   xyshen@eitech.edu.cn   marius.mosbach@mila.quebec

Abstract

In-context learning is a popular inference strategy where large language models solve a task using only a few labeled demonstrations without needing any parameter updates. Although there have been extensive studies on English in-context learning, multilingual in-context learning remains under-explored, and we lack an in-depth understanding of the role of demonstrations in this context. To address this gap, we conduct a multidimensional analysis of multilingual in-context learning, experimenting with 5 models from different model families, 9 datasets covering classification and generation tasks, and 56 typologically diverse languages. Our results reveal that the effectiveness of demonstrations varies significantly across models, tasks, and languages. We also find that strong instruction-following models including Llama 2-Chat, GPT-3.5, and GPT-4 are largely insensitive to the quality of demonstrations. Instead, a carefully crafted template often eliminates the benefits of demonstrations for some tasks and languages altogether. These findings show that the importance of demonstrations might be overestimated. Our work highlights the need for granular evaluation across multiple axes towards a better understanding of in-context learning.[1]

1 Introduction

An intriguing property of large language models (LLMs) is their ability to perform in-context learning (Brown et al., 2020), i.e., solve a task conditioned on a few demonstrations at inference time, without updating the model parameters. It has been shown to be an efficient alternative to fine-tuning when adapting models to diverse tasks and domains (Dong et al., 2022; Min et al., 2022b; Si et al., 2023, *inter alia*). In light of the success of in-context learning, there has been increased interest in better understanding the factors that influence its success, such as demonstration selection (Liu et al., 2022; Rubin et al., 2022; Wang et al., 2023c), prompt design (Min et al., 2022a; Wei et al., 2022), and more generally on understanding how and why in-context learning works (Xie et al., 2022; Bansal et al., 2023; Hendel et al., 2023; Pan et al., 2023; Wang et al., 2023b).

However, most recent work on in-context learning predominantly focuses on English, and the exploration of multilingual in-context learning generally lags behind. This is problematic, as results that apply to English might not hold for other languages, especially those that are less represented in LLM training data. While there have been a few studies on in-context learning that go beyond English, they either focus on benchmarking LLMs on multilingual tasks without in-depth exploration, e.g., MEGA (Ahuja et al., 2023) and BUFFET (Asai et al., 2023), or zoom in on specific capabilities such as mathematical reasoning (Shi et al., 2023b), machine translation (Zhu et al., 2023; Agrawal et al., 2023), or code-switching (Zhang et al., 2023).

In this work, we take a multidimensional approach (Ruder et al., 2022) that unifies these strands of research and comprehensively evaluate the multilingual in-context learning abilities of LLMs. We focus on dissecting the *actual* impact of in-context demonstrations, which is crucial for understanding model behaviour. Our research covers various models, tasks, and languages, and we seek to answer the following research questions:

1. Does multilingual performance benefit from demonstrations? (§4)

2. Does demonstration quality matter? (§5)

3. What is the interplay between demonstrations and templates? (§6)

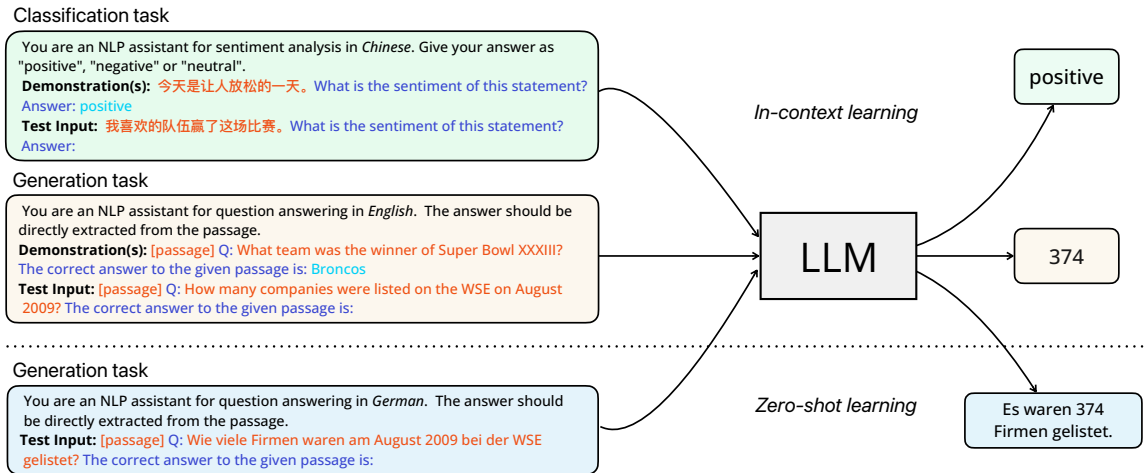4. How do the answers to these questions vary across languages and models? (§4, §5, §6)

Figure 1: An overview of the components of multilingual in-context learning (§2) with a comparison to zero-shot learning. Sources of variation include tasks, languages, models, and the template, i.e., the task instruction, patterns for formatting inputs, and verbalized labels.

Specifically, we address our research questions by evaluating 5 LLMs including base models that are only pre-trained on unlabeled text corpora (XGLM and Llama 2), and chat models that are further refined with instruction tuning and reinforcement learning (Llama 2-Chat, GPT-3.5, and GPT-4). We evaluate on 9 multilingual datasets that include both classification and generation tasks, covering 56 typologically different languages.

Our main findings are: (1) The effectiveness of demonstrations varies widely depending on the model, task, and language used. For base models, in-context learning barely outperforms zero-shot learning on many tasks. In general, in-context learning matters more for generation tasks with loosely-specified prompts; (2) Even with sophisticated demonstration selection methods, in-context learning is not always beneficial and can sometimes be worse than using no demonstrations at all; (3) Chat models are less sensitive to seeing correctly-labeled demonstrations than base models, suggesting that for the former, demonstrations primarily help the model understand the task format, while for the latter, demonstrations also impart task-specific knowledge; (4) Using a formatting-focused template can even eliminate the need for demonstrations with chat models. The relative significance of demonstrations versus prompt templates varies based on inherent model capabilities.

In sum, we suggest that the benefits of adding demonstrations may be overestimated. Future work on in-context learning should carefully compare their results with zero-shot learning and on multiple templates to faithfully represent its effectiveness. Given the vast variance across models, tasks, and languages, it is also important to cautiously frame claims about in-context learning.

2 Preliminaries

2.1 In-context learning

In-context learning (ICL) is a popular inference strategy where models solve[2] a task without any parameter updates (Brown et al., 2020). Instead, the model performs the task by conditioning on **labeled demonstrations**. Demonstrations are typically formatted using "pattern-verbalizer pairs," as this has been shown to be effective in eliciting good task performance (Schick and Schütze, 2021; Bach et al., 2022). Here, a *pattern* is used to format the input for the model, and a *verbalizer* maps the label to a textual representation. Additionally for instruction-tuned LLMs, a *task instruction* is often added to provide information about the task beyond individual demonstrations (Mishra et al., 2022b; Wang et al., 2022; Ouyang et al., 2022).

Formally, given a test sample x_t, k demonstrations $\{(x_i, y_i)\}_{i=1}^k$, a pattern $\mathcal{P}$, a verbalizer $\mathcal{V}$ and a task instruction $\mathcal{I}$, the model (parameterized by θ) makes its prediction as follows:

$$y_t \sim p_\theta(y|\mathcal{I}, \{(\mathcal{P}(x_i), \mathcal{V}(y_i))\}_{i=1}^k, \mathcal{P}(x_t)). \quad (1)$$

---

[2]The extent to which models actually "solve" tasks is an open question as ICL, similar to fine-tuning, has generalization issues despite its impressive results (Mosbach et al., 2023). Regardless, we use the word "solve" in the rest of this paper for simplicity.

Taken together, the pattern, the verbalizer and the optional task instruction comprise the **template** with which demonstrations and the test sample are formatted as the input prompt for model inference. The effectiveness of demonstrations is thus linked with the template used to present them to the model.

2.2 Multilingual prompting

Previous studies highlight that the selection of demonstrations and prompt templates can significantly influence model performance (Liu et al., 2022; Fu et al., 2023b; Sclar et al., 2024). In multilingual in-context learning, the variation in input prompts is further complicated by the *language* of demonstrations, templates and test samples, all of which are important design choices.

For the template language, Lin et al. (2022) and Ahuja et al. (2023) found that English templates generally perform better than native language templates, possibly due to superior instruction-following abilities on existing LLMs on English compared to other languages. Following this, we use English templates in our study.

For the language of few-shot demonstrations and test samples, there are three popular settings. Given a test sample in a certain language, the most straightforward approach is to use demonstrations in the same language (referred to as *in-language demonstrations*). This setting directly measures the model's inherent ability to solve problems in that language. Another choice is to use *English demonstrations* regardless of the language of the test sample. This is a cross-lingual transfer setup, where the goal is to transfer knowledge from a pivot language to a target language via in-context learning. As highlighted in Shi et al. (2023b) and Ahuja et al. (2023), in-language demonstrations often outperform English demonstrations on diverse multilingual tasks. Yet another option is to translate the test sample into English – an approach called *translate-test*, where the demonstrations are also in English. While translate-test leads to strong performance (Ahuja et al., 2023), this approach heavily relies on a translation system for data processing and centers the English proficiency of LLMs. In this work, we are interested in dissecting the intrinsic multilingual capabilities of LLMs, therefore we choose to use **in-language demonstrations**.

All these design choices are represented visually in Figure 1, which gives an overview of multilingual in-context learning. Detailed setup information is provided in the next section.

3 Experimental setup

Models. We evaluate two types of LLMs: pre-trained base models and chat models. Our base models include XGLM (Lin et al., 2022) and Llama 2 (Touvron et al., 2023). Our chat models are Llama 2-Chat, GPT-3.5 (Ouyang et al., 2022) and GPT-4 (OpenAI et al., 2023). Specifically, we use xglm-7.5B, Llama-2-13b, and Llama-2-13b-chat on Huggingface (Wolf et al., 2020), and we access gpt-3.5-turbo-16k and gpt-4-32k APIs via Microsoft Azure.[3]

Tasks and datasets. We experiment on a diverse range of multilingual classification and generation tasks, using 9 datasets covering 56 languages in total. Our dataset selection largely follows MEGA (Ahuja et al., 2023), but we add datasets for extremely under-represented African languages. Our classification tasks include natural language inference (NLI), paraphrase identification, commonsense reasoning and sentiment analysis, with the following datasets: XNLI (Conneau et al., 2018), IndicXNLI (Aggarwal et al., 2022), PAWS-X (Yang et al., 2019), XCOPA (Ponti et al., 2020), XStoryCloze (Lin et al., 2022) and AfriSenti (Muhammad et al., 2023). Our generation tasks are extractive question answering (QA) and machine translation (MT), for which we use XQuAD (Artetxe et al., 2020), TyDiQA-GoldP (Clark et al., 2020), and MAFAND (Adelani et al., 2022). See Appendix A.1 for more details.

In-context learning. For each test sample, we select $k \in \{0, 2, 4, 8\}$[4] different demonstrations, which are randomly sampled unless otherwise specified. All demonstrations are in the same language as the test sample, and all templates are in English. We employ appropriate task-specific templates for different model types. All templates and data splits are shown in Appendix A.2.

Metrics. For classification tasks, we report the rank classification accuracy[5] for open-source base models (Muennighoff et al., 2023; Lin et al., 2022).

---

[3]We also experiment with BLOOMZ and mT0 (Muennighoff et al., 2023). Results in Appendix B.1 show that their zero-shot performance significantly surpasses few-shot performance, which we ascribe to their training scheme.

[4]For QA datasets, we select a maximum of 4 demonstrations due to context size limitations.

[5]The scoring function is the average of per-token log probabilities (ignoring the common prefix of different candidates). The candidate with the highest score is chosen as the prediction.
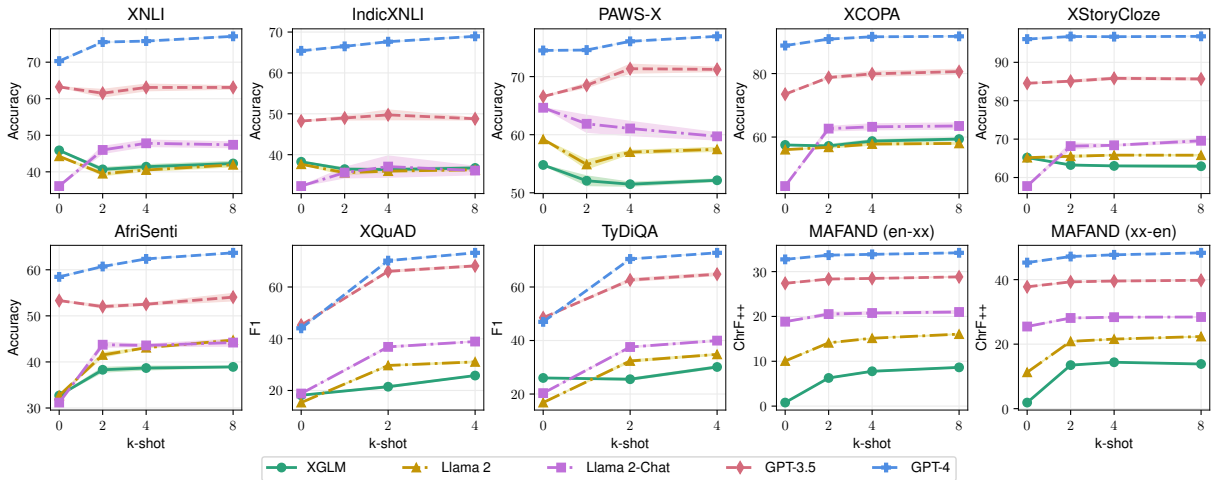
Figure 2: Average performance across languages with different numbers of demonstrations. We average and report standard deviations over 3 seeds for all models except GPT-4. Note that the standard deviations are relatively small, possibly because of averaging over languages. en-xx: translating from English to another language, xx-en: translating from another language to English.

For chat models, we measure the exact match between generated outputs[6] and verbalized labels (Ahuja et al., 2023). As for generation tasks, we use the F1 score for QA datasets and ChrF++ score (Popović, 2017) for MAFAND. Implementation details for our evaluation are provided in Appendix A.3.

4 Do (more) demonstrations benefit multilingual performance?

In this section, we systematically compare ICL and zero-shot learning as this question is under-explored in previous studies of multilingual ICL (Ahuja et al., 2023; Asai et al., 2023). We examine model performance on diverse multilingual tasks while varying the number of demonstrations, and show the results for classification tasks and generation tasks in Figure 2.

We begin with the overall trends across models and datasets. OpenAI's GPT-3.5 and GPT-4 models achieve the best multilingual in-context learning performance on all our datasets, which is unsurprising as they are currently the state-of-the-art on a large suite of NLP benchmarks. The next best models are Llama 2 and Llama 2-Chat, which demonstrate comparable or superior performance to the multilingual XGLM model despite being trained primarily on English corpora (Touvron et al., 2023). This indicates that their task-solving abilities can transfer across languages. Regardless of the model, however, performance on the AfriSenti
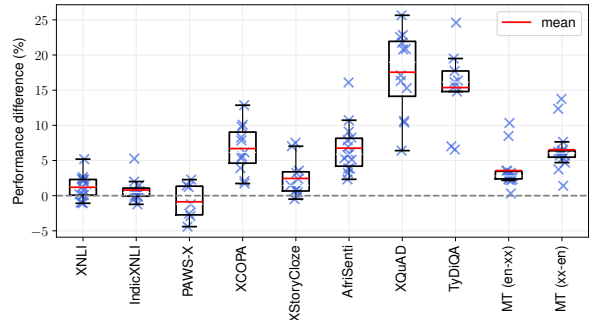

Figure 3: Performance difference between 4-shot and 0-shot. Each marker represents the average performance across models for each language in a given task. MT denotes the MAFAND dataset.

and MAFAND datasets, particularly when translating English to African languages, lags significantly behind other tasks, showing that language discrepancies remain even in the best models.

An important pattern across datasets and models is that **in-context learning does not always improve over zero-shot learning** – in particular, it helps with generation tasks, but results on classification tasks are mixed. For the AfriSenti dataset, many models show noticeable improvements with ICL. However, with other tasks such as IndicXNLI, XNLI and PAWS-X, the same models, especially base models, perform much worse compared to the zero-shot setting. We also see marginal improvements in some cases, e.g., XGLM and Llama 2 on XCOPA. In comparison to chat models, the addition of demonstrations typically reduces the performance of base models across many tasks.

---

[6] We extract verbalized labels from the generated outputs using regular expressions before calculating the exact match.

Model	XNLI	IndicXNLI	PAWS-X	XCOPA	XStoryCloze	AfriSenti	XQuAD	TyDiQA	MT (en-xx)	MT (xx-en)
XGLM	4.59	2.49	0.24▽	0.03	0.97▽	5.62	1.77	4.21	1.31	0.66
Llama 2	6.61	4.17	2.35	−0.11	0.33	4.17	1.32	0.54	2.15	1.35
Llama 2-Chat	−0.28	−1.36	−1.71▽	0.32	0.43	2.17	1.02	2.42	0.74	0.66
GPT-3.5	0.18	0.71	−2.07	0.86	−0.61	−0.66▽	−0.34	2.98	0.72	0.43
GPT-4	0.76	−0.19	0.07	−0.36	0.05	−0.68	−0.77	1.88	1.21	0.65

Table 1: Performance difference of 4-shot ICL with TOP-K vs. RANDOM selection. Positive numbers show that TOP-K is better than RANDOM (expected), and highlighted cells show where top-k is even worse than random selection. ▽: TOP-K performance is even worse than zero-shot learning. For RANDOM, we average over 3 seeds (except for GPT-4).

Model	XNLI	IndicXNLI	PAWS-X	XCOPA	XStoryCloze	AfriSenti	XQuAD	TyDiQA	MT (en-xx)	MT (xx-en)
XGLM	0.46	−0.05	0.44	0.51	0.62*	3.78*	24.56*	26.64*	3.18*	6.73*
Llama 2	0.96*	0.43	1.16	0.61*	1.12*	2.27*	26.68*	29.20*	4.79*	8.34*
Llama 2-Chat	−0.34	0.04	1.48	0.03	−0.23	0.77*	5.94*	4.37*	1.13*	1.53*
GPT-3.5	0.39	1.02	0.64	0.26	0.58*	−0.62	5.46*	5.61*	1.39*	0.48*
GPT-4	−0.86	−0.04	0.57	0.86	1.13	0.90	9.60	6.97	1.24	0.64

Table 2: Performance difference of 4-shot ICL with RANDOM vs. RANDOM-CORRUPTED demonstrations. Positive numbers show that RANDOM is better than RANDOM-CORRUPTED (expected), and highlighted cells show where corrupted labels perform even better than ground-truth labels. We average over 3 seeds (except for GPT-4). *: a significant difference ($p = 0.05$).
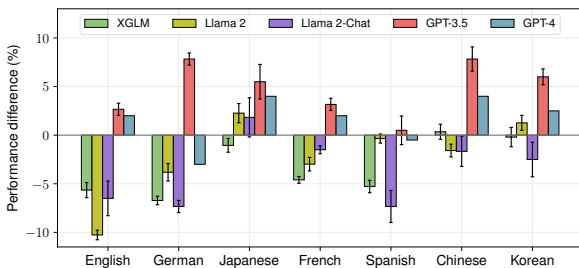

Figure 4: Performance difference between 4-shot and 0-shot for individual languages in PAWS-X. Error bars represent standard deviations calculated over 3 seeds.

When examining the cases where ICL improves performance, we see that **improvements saturate quickly with 2 to 4 demonstrations**. This aligns with Chen et al. (2023), who found that reducing the number of demonstrations to one does not significantly deteriorate chain-of-thought reasoning.

Looking at the improvements over zero-shot performance (for all models and languages combined) across tasks in Figure 3, we observe that there are large fluctuations between individual languages that are not captured by the average. The PAWS-X dataset in particular shows an average degradation, but in fact some languages benefit from ICL while others degrade. For a more nuanced understanding of language-specific differences within a task, we zoom into this dataset in Figure 4 to inspect these language-specific differences.[7] We see that

languages and models can behave very differently even on just one dataset, and a pattern which holds for one language with one model does not necessarily apply to a different language. For example, the ICL performance of Llama 2 outperforms its zero-shot performance by 2.3 points on Japanese and 1.3 points on Korean. However, demonstrations degrade performance for other languages, e.g., English performance degrades by 10.3 points. In sum, **the effectiveness of demonstrations varies widely depending on the model, task, and language.**

5 Does demonstration quality matter?

Our previous experiments evaluated ICL using randomly selected demonstrations. To ablate for the effects of demonstration quality, this section experiments with the choice of demonstrations as well as the importance of ground truth labels, i.e., the input-label mapping. Inspired by work on demonstration selection (Liu et al., 2022; Rubin et al., 2022) and input-label mapping (Min et al., 2022c; Yoo et al., 2022) in English, we compare the following three types of demonstrations:

- **RANDOM**: demonstrations are randomly selected from clean data

- **TOP-K**: the k most semantically similar[8] examples to a given test sample are selected (Liu

---
[7]Plots for other datasets are provided in Appendix B.2.

[8]We quantify semantic similarity using LaBSE (Feng et al., 2022), a multilingual sentence embedding model trained on 109+ languages.
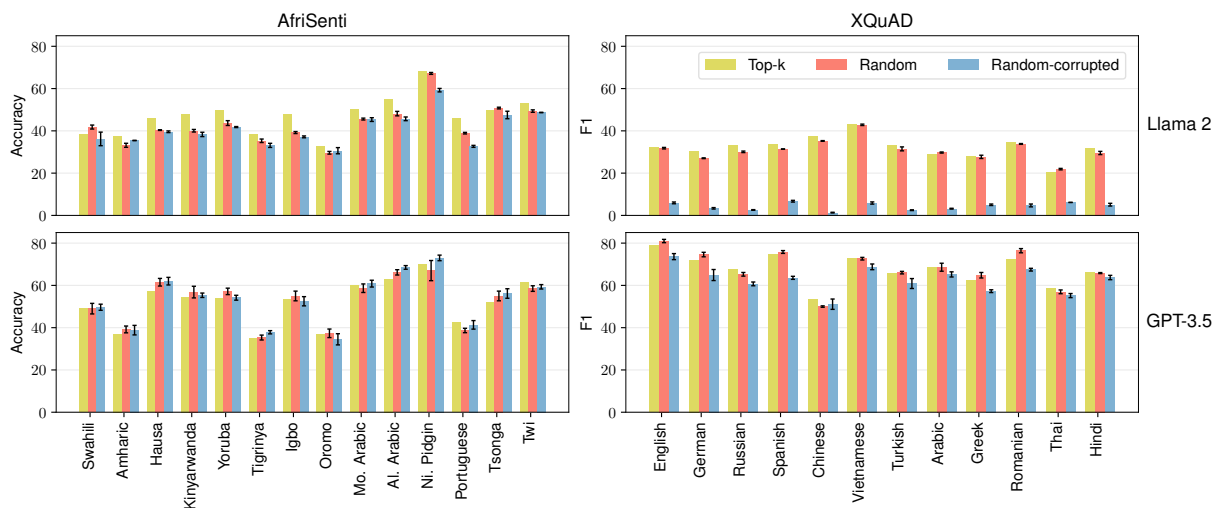
Figure 5: Performance of 4-shot ICL using different types of demonstrations for individual languages on AfriSenti and XQuAD. The top row shows Llama 2 results, and the bottom row shows GPT-3.5 results.

et al., 2022)

- **RANDOM-CORRUPTED**: demonstrations are randomly selected but the labels are corrupted by replacement with random labels[9] (Min et al., 2022c)

Table 1 shows that top-k selection performs better than random selection in many cases, especially for the base models XGLM and Llama 2. For chat models, the largest improvements are on generation tasks. For example, GPT-3.5 achieves a 2.98-point improvement on TyDiQA. Nevertheless, top-k selection often degrades performance on many other tasks, e.g., GPT-3.5 is 2.07 points worse on PAWS-X compared to random selection. When compared to zero-shot performance, ICL with top-k selection is even *worse* in some cases, such as XGLM on PAWS-X and XStoryCloze. In cases where random selection performs worse than zero-shot, even top-k selection gives only marginal improvements (see detailed numbers in Table 5 in Appendix C.1). These findings indicate that **sophisticated demonstration selection methods are not always beneficial and can sometimes be worse than using no demonstrations at all**.

Exploring this further, in Table 2, we compare randomly selected demonstrations with ground truth labels and corrupted labels. We find that using corrupted labels does not hurt performance on multilingual classification tasks much, which is consistent with previous research on English (Min

et al., 2022c). On generation tasks, however, all models perform worse with corrupted labels, but to vastly different extents. XGLM and Llama 2 perform significantly worse with corrupted labels, especially on the machine translation task, whereas **chat models do not rely as much on correct labels**. This might be explained by ICL helping the model understand the task format and activating prior knowledge acquired by the model, rather than the model learning the task from demonstrations. The observed model insensitivity to correct labels on certain tasks implies that random labels can serve as a strong baseline for demonstration generation before exploring more complex methods (Lyu et al., 2023; Wan et al., 2023).

To investigate how these patterns split up across languages, Figure 5 shows language-specific results on AfriSenti and XQuAD with Llama 2 and GPT-3.5.[10] On AfriSenti, top-k selection outperforms random selection with Llama 2 across most languages; however, in the case of Swahili and Tsonga, there is a performance drop of 3.2 and 1.2 points, respectively. With GPT-3.5, top-k selection does not help across most languages, but it does help with Mozambican Portuguese and Twi. Similarly, the impact of corrupted labels varies. Llama 2 is affected dramatically by corrupted labels on all languages in XQuAD, whereas GPT-3.5 is much less affected, although to varying degrees across different languages. We urge NLP practitioners to **attend to these discrepancies when creating language-specific applications**, and leave it to future work to explore where they come from.

---

[9]For classification tasks, we randomly choose a label from the fixed label set. For generation tasks, we randomly choose a label from the label space of the entire demonstration data.

[10]See Appendix C.2 for other models and datasets.

6 Better templates further reduce the benefits of demonstrations

In-context learning performance depends not only on the demonstrations, which we have varied so far, but also on how they are formatted using templates. Previous work (Gonen et al., 2023; Mizrahi et al., 2024) has shown that modifying the template changes task performance. This section thus seeks to examine the interplay between template choice and demonstrations.

Template design. In the zero-shot setting, we observe that chat models tend to generate verbose responses (e.g., "Sure! I can help you with that") or explanations (e.g., "The reason is that ...") that pose a challenge for automatic evaluation. We observe a reduction in this behaviour with ICL, which leads us to question whether demonstrations are merely a means to format model responses. To see if we can achieve the same effect with minor template engineering, we augment the original templates with instructions that focus on output formatting. We call these *formatting-focused templates* which are shown in Table 9.

In this section, we focus on XCOPA, AfriSenti, XQuAD, and TyDiQA, as these are the classification and generation tasks that seem to benefit most from in-context demonstrations (see Section 4). However, as Figure 6 shows, **the performance gap between zero-shot and in-context learning diminishes with formatting-focused templates**. The gap reduction is more substantial for QA datasets (i.e., the generation tasks) than for XCOPA and AfriSenti (i.e., the classification tasks). We speculate that it is simpler for the model to generate label words for classification tasks with a pre-defined label space than to answer questions in a way that is easy to evaluate automatically. In the latter case, formatting-focused templates can teach output styling, largely eliminating the benefits of demonstrations.

Compared to GPT-3.5 and GPT-4, Llama 2-Chat performs worse in both zero-shot and few-shot settings, and formatting-focused templates have a less pronounced impact. On QA datasets, GPT-3.5 and GPT-4 even achieve better zero-shot performance with formatting-focused templates than ICL with original templates, a pattern that is not observed with Llama 2-Chat. This suggests that **the relative significance of demonstrations and templates varies based on the inherent abilities of models** at solving tasks and following instructions.
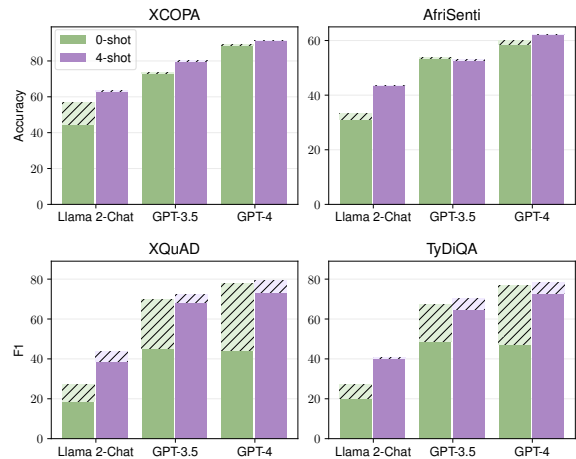

Figure 6: Effect of using different templates on 0-shot and 4-shot performance. Formatting-focused templates (with hatching) improve 0-shot performance over original templates (solid colours), and reduce the gap between 0-shot and 4-shot performance. Few-shot results are averaged across 3 seeds except for GPT-4.

Model	Demo. Label	XQuAD		TyDiQA	
		O	F	O	F
Llama 2-Chat	Original	$38.9_{\pm0.1}$	$43.8_{\pm0.7}$	$40.0_{\pm0.3}$	$40.6_{\pm0.8}$
	Corrupted	$33.0_{\pm0.4}$	$38.6_{\pm0.3}$	$35.6_{\pm0.1}$	$36.3_{\pm0.5}$
	Δ	5.9	5.2	4.4	4.3
GPT-3.5	Original	$68.2_{\pm0.4}$	$72.2_{\pm0.4}$	$64.8_{\pm0.5}$	$70.5_{\pm0.5}$
	Corrupted	$62.7_{\pm0.2}$	$69.9_{\pm0.2}$	$59.2_{\pm0.3}$	$67.1_{\pm0.7}$
	Δ	5.5	2.3	5.6	3.4
GPT-4	Original	73.2	79.3	72.8	78.3
	Corrupted	63.6	79.8	65.8	77.6
	Δ	9.6	-0.5	7.0	0.7

Table 3: Effect of using different templates on 4-shot performance with RANDOM and RANDOM-CORRUPTED demonstrations. When using formatting-focused templates (F) over the original templates (O), the performance gap (Δ) between original and corrupted labels decreases. We average and report standard deviations over 3 seeds for all models except GPT-4.

With our new formatting-focused templates, we revisit the impact of the input-label mapping discussed in Section 5. As Table 3 shows, all models perform worse with corrupted labels, but formatting-focused templates largely mitigate this degradation. Notably, **GPT-4 using corrupted labels performs on par with ground truth labels**. This strengthens our finding that the correct input-label mapping is not that important, while also highlighting the crucial role that templates play in in-context learning.

Figure 7 shows the language-specific effects of formatting-focused templates on XQuAD (results for other tasks are in Appendix D.1). For Llama 2-Chat, demonstrations remain essential even with
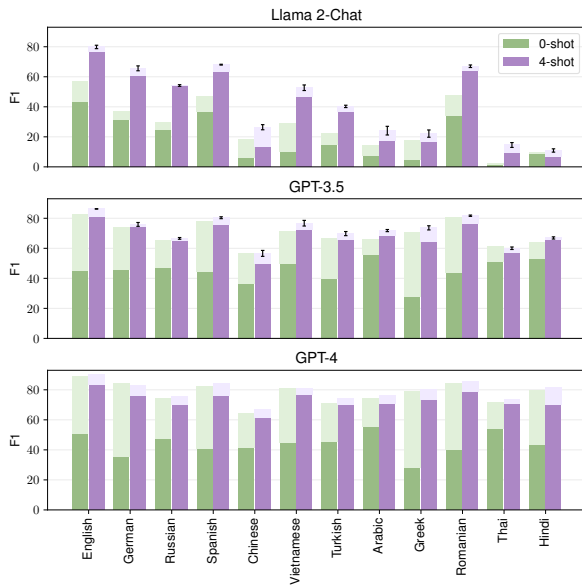
Figure 7: Effect of using different templates on 0-shot and 4-shot XQuAD performance. Formatting-focused templates (with hatching) improve 0-shot performance over original templates (solid colours), and reduce the gap between 0-shot and 4-shot performance. Few-shot results are averaged across 3 seeds except for GPT-4.

a formatting-focused template for most languages, but not Greek and Hindi. GPT-3.5 and GPT-4 also show variance across languages. Moreover, for most languages, zero-shot learning with minor template engineering can match and even exceed in-context learning performance, aligning with previous work on GPT-3 (Reynolds and McDonell, 2021). The fact that we can achieve the same effects through template engineering or demonstrations reinforces our hypothesis that models are not actually learning tasks on the fly. Instead, some combination of demonstrations and templates serves to activate prior knowledge of a task and encourage a consistent output format for automatic evaluation.

7 Discussion

Our systematic study provides strong evidence that the importance of in-context demonstrations on existing multilingual datasets might be overestimated, as it highly depends on the model, task, and language used. For strong instruction-following models, the effect of demonstrations is *superficial* and can be eliminated with minor template engineering. These findings open up new questions, which we discuss below.

Understanding the failures of ICL. There has been a surge of research interest in understanding

the underlying mechanisms of ICL (Xie et al., 2022; Von Oswald et al., 2023; Wang et al., 2023b; Hendel et al., 2023), motivated by its successes. Our results show that ICL is *not* always effective, and that its performance changes depending on multiple factors including the choice of model, task and language. The failures of ICL need as much scrutiny as its successes for a more fundamental understanding of the learning mechanisms of LLMs.

Optimizing demonstrations or templates. With the increasing popularity of research on demonstration selection (Liu et al., 2022; Rubin et al., 2022; Li et al., 2023b) and prompt engineering (Mishra et al., 2022a; White et al., 2023; Khattab et al., 2023), it is important to understand the interplay of the two. We show that good demonstrations help base models perform better on certain tasks, but that formatting-focused prompting has a much bigger impact on chat models. These results show that the impact of demonstrations cannot be fairly evaluated in isolation from the choice of prompt. These findings have implications both for researchers interested in fairly evaluating ICL, and for practitioners to choose to spend time optimizing demonstrations, templates or both.

Evaluating multilingual ICL. Compared to the extensive research on ICL in English (Zhao et al., 2021; Dong et al., 2022; Min et al., 2022b; Mosbach et al., 2023), multilingual ICL remains underexplored. There is no widely accepted setup to robustly evaluate the effectiveness of ICL across languages, since the choice of multilingual models and tasks is limited. Based on our findings, we have some recommendations for the nascent field of multilingual ICL. First, critical evaluation is important. We need to compare ICL strategies to zero-shot learning, and ablate them with multiple templates. Second, as there is so much variance across models, tasks and languages, it is important to carefully scope claims about ICL. Last but not least, every language is different, so granular per-language analysis is a must in multilingual research.

8 Related Work

Multilingual in-context learning. Most multilingual in-context learning studies focus on benchmarking LLMs on diverse tasks and comparing them with smaller fine-tuned models (Ahuja et al., 2023; Asai et al., 2023; Zhang et al., 2023; Zhu et al., 2023). As these works focus on benchmark-

ing, their analysis of the role of demonstrations is limited. Ahuja et al. (2023) explore different prompting strategies by adjusting the language of templates and demonstrations. Zhang et al. (2023) find that demonstrations sometimes do not contribute to or even degrade model performance on code-switching. Zhu et al. (2023) look at machine translation and analyze the effects of template and demonstration selection with XGLM. In the context of cross-lingual transfer, Shi et al. (2022), Tanwar et al. (2023), and Agrawal et al. (2023) investigate demonstration selection for specific applications. In contrast, we take a much broader perspective and investigate the actual impact of demonstrations across a wide range of models, tasks and languages.

English-centric demonstration analysis. Most of the current demonstration analysis literature focuses on English: Lu et al. (2022) analyze the sensitivity of ICL to the order of demonstrations, Min et al. (2022c) and Yoo et al. (2022) explore whether the ground truth labels matter for classification tasks, and Wei et al. (2023) investigate the sensitivity of various model families to different input-label mappings. Similarly, Pan et al. (2023) disentangle task recognition and task learning by manipulating the label space. Beyond this, Shi et al. (2023a) and Wang et al. (2023a) modify the validity of chain-of-thought (CoT) reasoning steps in demonstrations and explore the impact of this modification on mathematical reasoning. Also focusing on CoT, Chen et al. (2023) investigate how varying the number of demonstrations affects performance.

9 Conclusion

In this paper, we conduct an in-depth multidimensional analysis on the impact of demonstrations in multilingual in-context learning. We find that the use of demonstrations does not always provide benefits compared to zero-shot learning, and that there is a large variance in performance across models, datasets and languages. While the quality of demonstrations influences the performance of base LLMs on certain tasks, the impact is significantly reduced for LLMs tuned with alignment techniques. We also examine the interplay between demonstrations and templates, finding that a carefully crafted template can further decrease the benefits of demonstrations. Our granular analysis contributes novel insights with nuance and paves the way for a more thoughtful multilingual ICL evaluation.

Limitations

Data contamination. Since LLMs are trained with a vast amount of data scraped from the internet, this might result in data contamination, i.e., when the training data includes test datasets. Ahuja et al. (2023) suspect that many multilingual datasets appear in the training data of GPT-4, which might lead to an overestimation of the model's capabilities. In the context of our work, our prompt might just be reminding LLMs of a task they have already seen, whereas on an unseen task, the impact of demonstrations might be different. We do not examine the impact of potential data contamination in our paper and leave an exploration of this to future work.

Other demonstration choices. In this work, we choose to use demonstrations that are in the same languages as the test sample, due to our focus on evaluating inherent multilingual abilities of LLMs, as explained in Section 2.2. However, using English demonstrations for cross-lingual transfer or translating test samples into English has its own practical value for NLP applications. Additionally, it is worth exploring selecting demonstrations from a mixture of languages. Expanding our study to more setups would provide additional insights into multilingual and cross-lingual LLM abilities.

Other prompting methods. In Section 6, we only experiment with manually augmented templates to illustrate how the choice of template can reduce the effectiveness of demonstrations. There is a broad literature on prompt engineering and prompt sensitivity (White et al., 2023; Gonen et al., 2023), suggesting that it is plausible that another prompt could reduce the gap between few-shot and zero-shot performance even further. Chain-of-thought (CoT) prompting is another approach with promising multilingual abilities (Shi et al., 2023b; Huang et al., 2023) that might affect our findings. Our manually-augmented templates are intended only as a starting point for further analysis, which we leave to future work.

Beyond automatic evaluation. When examining model responses, we noticed some cases where a correct answer as evaluated by a human was not fully captured by automatic evaluation metrics. Human evaluation is time-consuming, expensive, and hard to source for the wide range of languages that we explore in our work. Another option is LLM

evaluation, which is becoming increasingly popular (Fu et al., 2023a; Chan et al., 2024), but is also an expensive approach. More importantly, we have no guarantees about LLMs' multilingual capabilities. As a trade-off between cost and evaluation quality, we stick to automatic evaluation in our work for all tasks and languages.

Acknowledgments

References

David Adelani, Jesujoba Alabi, Angela Fan, Julia Kreutzer, Xiaoyu Shen, Machel Reid, Dana Ruiter, Dietrich Klakow, Peter Nabende, Ernie Chang, Tajuddeen Gwadabe, Freshia Sackey, Bonaventure F. P. Dossou, Chris Emezue, Colin Leong, Michael Beukman, Shamsuddeen Muhammad, Guyo Jarso, Oreen Yousuf, Andre Niyongabo Rubungo, Gilles Hacheme, Eric Peter Wairagala, Muhammad Umair Nasir, Benjamin Ajibade, Tunde Ajayi, Yvonne Gitau, Jade Abbott, Mohamed Ahmed, Millicent Ochieng, Anuoluwapo Aremu, Perez Ogayo, Jonathan Mukiibi, Fatoumata Ouoba Kabore, Godson Kalipe, Derguene Mbaye, Allahsera Auguste Tapo, Victoire Memdjokam Koagne, Edwin Munkoh-Buabeng, Valencia Wagner, Idris Abdulmumin, Ayodele Awokoya, Happy Buzaaba, Blessing Sibanda, Andiswa Bukula, and Sam Manthalu. 2022. A few thousand translations go a long way! leveraging pre-trained models for African news translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3053–3070, Seattle, United States. Association for Computational Linguistics.

Divyanshu Aggarwal, Vivek Gupta, and Anoop Kunchukuttan. 2022. IndicXNLI: Evaluating multilingual inference for Indian languages. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10994–11006, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sweta Agrawal, Chunting Zhou, Mike Lewis, Luke Zettlemoyer, and Marjan Ghazvininejad. 2023. In-context examples selection for machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8857–8873, Toronto, Canada. Association for Computational Linguistics.

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267, Singapore. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Akari Asai, Sneha Kudugunta, Xinyan Velocity Yu, Terra Blevins, Hila Gonen, Machel Reid, Yulia Tsvetkov, Sebastian Ruder, and Hannaneh Hajishirzi. 2023. BUFFET: Benchmarking large language models for few-shot cross-lingual transfer. *arXiv*.

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.

Hritik Bansal, Karthik Gopalakrishnan, Saket Dingliwal, Sravan Bodapati, Katrin Kirchhoff, and Dan Roth. 2023. Rethinking the role of scale for in-context learning: An interpretability-based case study at 66 billion scale. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11833–11856, Toronto, Canada. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2024. ChatEval: Towards better LLM-based evaluators through multi-agent debate. In *The Twelfth International Conference on Learning Representations*.

Jiuhai Chen, Lichang Chen, Chen Zhu, and Tianyi Zhou. 2023. How many demonstrations do you need for in-context learning? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11149–11159, Singapore. Association for Computational Linguistics.

Yanda Chen, Ruiqi Zhong, Sheng Zha, George Karypis, and He He. 2022. Meta-learning via language model in-context tuning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 719–730, Dublin, Ireland. Association for Computational Linguistics.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey for in-context learning. *arXiv*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023a. GPTScore: Evaluate as you desire. *arXiv*.

Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2023b. Complexity-based prompting for multi-step reasoning. In *The Eleventh International Conference on Learning Representations*.

Hila Gonen, Srini Iyer, Terra Blevins, Noah Smith, and Luke Zettlemoyer. 2023. Demystifying prompts in language models via perplexity estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10136–10148, Singapore. Association for Computational Linguistics.

Roee Hendel, Mor Geva, and Amir Globerson. 2023. In-context learning creates task vectors. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9318–9333, Singapore. Association for Computational Linguistics.

Haoyang Huang, Tianyi Tang, Dongdong Zhang, Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023. Not all languages are created equal in LLMs: Improving multilingual capability by cross-lingual-thought prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12365–12394, Singapore. Association for Computational Linguistics.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. 2023. DSPy: Compiling declarative language model calls into self-improving pipelines. *arXiv*.

Viet Lai, Nghia Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Nguyen. 2023. ChatGPT beyond English: Towards a comprehensive evaluation of large language models in multilingual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13171–13189, Singapore. Association for Computational Linguistics.

Cheng Li, Jindong Wang, Yixuan Zhang, Kaijie Zhu, Wenxin Hou, Jianxun Lian, Fang Luo, Qiang Yang, and Xing Xie. 2023a. Large language models understand and can be enhanced by emotional stimuli. *arXiv*.

Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. 2023b. Unified demonstration retriever for in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4644–4668, Toronto, Canada. Association for Computational Linguistics.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. 2022. Few-shot learning with multilingual generative language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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. Association for Computational Linguistics.

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.

Xinxi Lyu, Sewon Min, Iz Beltagy, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Z-ICL: Zero-shot in-context learning with pseudo-demonstrations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2304–2317, Toronto, Canada. Association for Computational Linguistics.

Sewon Min, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022a. Noisy channel language model prompting for few-shot text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5316–5330, Dublin, Ireland. Association for Computational Linguistics.

Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2022b. 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. 2022c. 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. 2022a. 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.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022b. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487, Dublin, Ireland. Association for Computational Linguistics.

Moran Mizrahi, Guy Kaplan, Dan Malkin, Rotem Dror, Dafna Shahaf, and Gabriel Stanovsky. 2024. State of what art? A call for multi-prompt LLM evaluation. *arXiv*.

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. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12284–12314, Toronto, Canada. Association for Computational Linguistics.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

Shamsuddeen Muhammad, Idris Abdulmumin, Abinew Ayele, Nedjma Ousidhoum, David Adelani, Seid Yimam, Ibrahim Ahmad, Meriem Beloucif, Saif Mohammad, Sebastian Ruder, Oumaima Hourrane, Alipio Jorge, Pavel Brazdil, Felermino Ali, Davis David, Salomey Osei, Bello Shehu-Bello, Falalu Lawan, Tajuddeen Gwadabe, Samuel Rutunda, Tadesse Belay, Wendimu Messelle, Hailu Balcha, Sisay Chala, Hagos Gebremichael, Bernard Opoku, and Stephen Arthur. 2023. AfriSenti: A Twitter sentiment analysis benchmark for African languages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13968–13981, Singapore. Association for Computational Linguistics.

Jessica Ojo, Kelechi Ogueji, Pontus Stenetorp, and David I. Adelani. 2023. How good are large language models on African languages? *arXiv*.

OpenAI, :, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, et al. 2023. GPT-4 technical report. *arXiv*.

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.

Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. 2023. What in-context learning "learns" in-context: Disentangling task recognition and task learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8298–8319, Toronto, Canada. Association for Computational Linguistics.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online. Association for Computational Linguistics.

Maja Popović. 2017. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark. Association for Computational Linguistics.

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–7.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671, Seattle, United States. Association for Computational Linguistics.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2022. Square one bias in NLP: Towards a multi-dimensional exploration of the research manifold. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2340–2354, Dublin, Ireland. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023a. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. 2023b. Language models are multilingual chain-of-thought reasoners. In *The Eleventh International Conference on Learning Representations*.

Peng Shi, Rui Zhang, He Bai, and Jimmy Lin. 2022. XRICL: Cross-lingual retrieval-augmented in-context learning for cross-lingual text-to-SQL semantic parsing. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5248–5259, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chenglei Si, Zhe Gan, Zhengyuan Yang, Shuohang Wang, Jianfeng Wang, Jordan Lee Boyd-Graber, and Lijuan Wang. 2023. Prompting GPT-3 to be reliable. In *The Eleventh International Conference on Learning Representations*.

Eshaan Tanwar, Subhabrata Dutta, Manish Borthakur, and Tanmoy Chakraborty. 2023. Multilingual LLMs are better cross-lingual in-context learners with alignment. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6292–6307, Toronto, Canada. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv*.

Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, Joao Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. 2023. Transformers learn in-context by gradient descent. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 35151–35174. PMLR.

Xingchen Wan, Ruoxi Sun, Hootan Nakhost, Hanjun Dai, Julian Eisenschlos, Sercan Arik, and Tomas Pfister. 2023. Universal self-adaptive prompting. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7437–7462, Singapore. Association for Computational Linguistics.

Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. 2023a. Towards understanding chain-of-thought prompting: An empirical study of what matters. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2717–2739, Toronto, Canada. Association for Computational Linguistics.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023b. Label words are anchors: An information flow perspective for understanding in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9840–9855, Singapore. Association for Computational Linguistics.

Xinyi Wang, Wanrong Zhu, Michael Saxon, Mark Steyvers, and William Yang Wang. 2023c. Large language models are latent variable models: Explaining and finding good demonstrations for in-context learning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

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 on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed 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*, volume 35, pages 24824–24837. Curran Associates, Inc.

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*.

Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. 2023. A prompt pattern catalog to enhance prompt engineering with chatgpt. *arXiv*.

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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhenyu Wu, Yaoxiang Wang, Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Jingjing Xu, and Yu Qiao. 2023. OpenICL: An open-source framework for in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 489–498, Toronto, Canada. Association for Computational Linguistics.

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*.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. 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 3687–3692, Hong Kong, China. Association for Computational Linguistics.

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.

Ruochen Zhang, Samuel Cahyawijaya, Jan Christian Blaise Cruz, Genta Winata, and Alham Aji. 2023. Multilingual large language models are not (yet) code-switchers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12567–12582, Singapore. 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 *International Conference on Machine Learning*, pages 12697–12706. PMLR.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Multilingual machine translation with large language models: Empirical results and analysis. *arXiv*.

A Experimental details

A.1 Tasks and datasets

We conduct experiments on 9 multilingual datasets with a wide coverage of tasks and languages, as shown in Table 4. All datasets are public research datasets and our experiments are consistent with their intended use, i.e., NLP evaluation. For the machine translation dataset MAFAND, English serves as the pivot language and there are two translation directions: en-xx (i.e., translating from English to another language) and xx-en (i.e., translating from another language to English). As the black-box training data of OpenAI APIs that we used is up to September 2021, we include the dataset release date in the table which can be taken as a clue to the severity of dataset contamination.

Dataset	Task	Languages	|Lang.|	Release Date
XNLI	natural language inference	English, German, Russian, French, Spanish, Chinese, Vietnamese, Turkish, Arabic, Greek, Thai, Bulgarian, Hindi, Urdu, Swahili	15	2019.09
IndicXNLI	natural language inference	Hindi, Bengali, Tamil, Marathi, Malayalam, Telugu, Kannada, Punjabi, Oriya, Assamese, Gujarati	11	2022.04
PAWS-X	paraphrase identification	English, German, Japanese, French, Spanish, Chinese, Korean	7	2019.08
XCOPA	commonsense reasoning	Chinese, Italian, Vietnamese, Indonesian, Turkish, Thai, Estonian, Tamil, Swahili, Haitian, Quechua	11	2020.04
XStoryCloze	commonsense reasoning	English, Russian, Spanish, Chinese, Indonesian, Arabic, Hindi, Basque, Telugu, Burmese, Swahili	11	2023.05
AfriSenti	sentiment analysis	Swahili, Amharic, Hausa, Kinyarwanda, Yoruba, Tigrinya, Igbo, Oromo, Moroccan Arabic, Algerian Arabic, Nigerian Pidgin, Mozambican Portuguese, Tsonga, Twi	14	2023.05
XQuAD	extractive QA	English, German, Russian, Spanish, Chinese, Vietnamese, Turkish, Greek, Romanian, Thai, Hindi	12	2019.10
TyDiQA-GoldP	extractive QA	English, Russian, Indonesian, Korean, Arabic, Finnish, Bengali, Telugu, Swahili	9	2020.02
MAFAND	machine translation	Amharic, Hausa, Kinyarwanda, Luganda, Luo, Chichewa, Nigerian Pidgin, Shona, Swahili, Setswana, Twi, Xhosa, Yoruba, Zulu	14	2022.06

Table 4: Multilingual benchmarking datasets.

A.2 In-context learning

We sample few-shot demonstrations from the validation set and evaluate the test set. For datasets without a test data split (XStoryCloze and TyDiQA), we sample few-shots from the train set and evaluate the validation set. Since XQuAD only has a validation data split, we utilize it for both demonstration sampling and evaluation, ensuring that the test sample itself is not included in its demonstrations. For chat models (Llama 2-Chat, GPT-3.5, and GPT-4), we limit the test sample size to a maximum of 200 in order to reduce inference expenses and ensure a fair comparison.

We use GPT-3 style prompting templates for XGLM and Llama 2 as shown in Table 6. The templates for BLOOMZ and mT0 are shown in Table 7. For Llama 2-Chat, GPT-3.5 and GPT-4, default templates are shown in Table 8 and task instructions are used to assign a system role to the model. Inspired by Lai et al. (2023) and Li et al. (2023a), where emotional stimuli are able to enhance LLM understanding, we design formatting-focused templates (discussed in Section 6) to reinforce LLM to generate formatted outputs that are easier to evaluate automatically, as shown in Table 9.

A.3 Implementation

Our codebase is adapted from OpenICL (Wu et al., 2023). We use int8bit model quantization[11] for all models except OpenAI models. Experiments are conducted using a single NVIDIA A100-80GB GPU. As models have a maximum context length, we preserve complete demonstrations that can fit within the context window. We employ greedy decoding for model generation. For chat models, the maximum new token is set to 50, while for machine translation, it is set to 100. For other models, the maximum

---

[11]In our preliminary experiments, we found that int8 quantization led to a performance degradation of 1-2% on a few classification datasets with Llama 2 and XGLM. Since this degradation is consistent across different setups, we believe that it would not affect our overall findings.

new token is set to 20, while for machine translation, it is set to 50. We use three seeds $(0, 33, 42)$ in our experiments, and the single-seed results for BLOOMZ and mT0 are obtained with the seed 0.

B   More results for varying numbers of demonstrations

In this section, we provide supplemental results for Section 4.

B.1   Results for BLOOMZ and mT0

In addition to the 5 models (base models and chat models) we discussed in the main content, we also experiment with two instruction-tuned models: BLOOMZ and mT0 (Muennighoff et al., 2023). The results for varying numbers of random demonstrations are shown in Figure 8. In line with findings from Asai et al. (2023), we observe significant performance degradation when using demonstrations compared to zero-shot learning in all cases. This decline can be attributed to their training scheme, where models are fine-tuned on a large collection of existing datasets in a zero-shot manner. In contrast, several studies (Chen et al., 2022; Wang et al., 2022) focus on enhancing the in-context learning ability of LLMs by incorporating demonstrations into their training process. This suggests that we should be careful in model selection for in-context learning research and take the model training process into consideration.
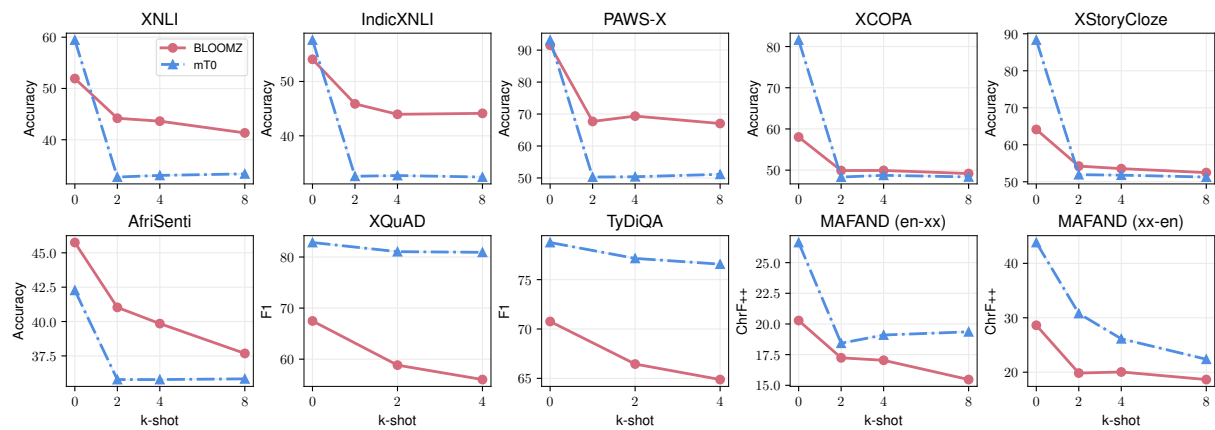
Figure 8: Average performance across languages for BLOOMZ and mT0 with different numbers of demonstrations. The results are obtained with a single random seed. Note that PAWS-X, XQuAD and TyDiQA are included in the instruction-tuning datasets of BLOOMZ and mT0.

B.2   Results for individual languages

The language-specific results for each task are shown in Figure 9. The order of languages follows their data ratio in the CommonCrawl corpus[12] from high-resource to low-resource. We observe large variations in model performance across different languages. For instance, there exists a large performance disparity between English and Urdu in XNLI. In XCOPA, the performance of Quechua is significantly worse compared to other languages.

C   More results for ablating the quality of demonstrations

In this section, we provide supplemental results for Section 5.

C.1   Performance of different types of demonstrations

In Table 5, we show the model performance of three types of demonstrations, as well as the zero-shot performance for comparative analysis. As we notice, top-k selection may not always be the optimal choice, given the considerable effort in optimizing demonstrations. For QA, XGLM and Llama 2's abilities in solving this task almost collapse with corrupted labels. However, for chat models, demonstrations with corrupted labels can achieve comparable performance with ground truth labels and largely improve the

---

[12]http://commoncrawl.org

Model	Demonstration	XNLI	IndicXNLI	PAWS-X	XCOPA	XStoryCloze	AfriSenti	XQuAD	TyDiQA	MT (en-xx)	MT (xx-en)
XGLM	ZERO-SHOT	45.87	38.27	**54.79**	57.51	**65.19**	32.71	18.16	26.01	0.79	1.89
	TOP-K	**45.99**	**38.85**	51.72	**58.76**	63.99	**44.30**	**27.54**	**34.32**	**9.08**	**15.05**
	RANDOM	$41.40_{0.50}$	$36.36_{0.33}$	$51.48_{0.33}$	$58.73_{0.43}$	$63.02_{0.09}$	$38.68_{0.39}$	$25.77_{0.06}$	$30.11_{0.36}$	$7.77_{0.09}$	$14.39_{0.02}$
	RANDOM-CORRUPTED	$40.94_{0.42}$	$36.41_{0.35}$	$51.04_{0.28}$	$58.21_{0.30}$	$62.40_{0.21}$	$34.90_{0.42}$	$1.21_{0.03}$	$3.47_{0.07}$	$4.59_{0.01}$	$7.66_{0.04}$
Llama 2	ZERO-SHOT	44.25	37.66	59.21	56.02	65.17	32.71	15.33	16.81	10.06	11.27
	TOP-K	**47.10**	**40.15**	**59.35**	57.69	**66.16**	**47.25**	**32.37**	**35.36**	**17.29**	**22.92**
	RANDOM	$40.49_{0.35}$	$35.98_{0.24}$	$57.00_{0.29}$	$\mathbf{57.80}_{0.32}$	$65.83_{0.08}$	$43.08_{0.02}$	$31.05_{0.28}$	$34.82_{0.21}$	$15.14_{0.01}$	$21.57_{0.02}$
	RANDOM-CORRUPTED	$39.53_{0.20}$	$35.55_{0.44}$	$55.85_{0.92}$	$57.19_{0.06}$	$64.71_{0.25}$	$40.81_{0.36}$	$4.36_{0.25}$	$5.62_{0.27}$	$10.35_{0.04}$	$13.23_{0.04}$
Llama 2-Chat	ZERO-SHOT	36.10	32.32	**64.64**	44.55	57.77	31.18	18.82	20.33	18.83	25.46
	TOP-K	47.53	35.73	59.36	**63.55**	68.82	45.75	39.94	42.38	**21.50**	**29.02**
	RANDOM	$47.81_{0.85}$	$\mathbf{37.09}_{2.57}$	$61.07_{1.2}$	$63.23_{0.91}$	$68.39_{0.14}$	$43.58_{0.23}$	$38.92_{0.09}$	$39.96_{0.30}$	$20.76_{0.23}$	$28.36_{0.12}$
	RANDOM-CORRUPTED	$\mathbf{48.15}_{1.22}$	$37.05_{3.09}$	$59.59_{1.13}$	$63.20_{0.33}$	$68.62_{0.93}$	$42.81_{0.11}$	$32.98_{0.39}$	$35.59_{0.08}$	$19.63_{0.04}$	$26.82_{0.11}$
GPT-3.5	ZERO-SHOT	63.23	48.23	66.57	73.50	84.55	**53.32**	45.25	48.52	27.39	37.77
	TOP-K	**63.27**	**50.45**	69.29	**80.77**	85.23	51.86	67.82	**67.76**	**29.20**	**39.99**
	RANDOM	$63.09_{0.88}$	$49.74_{1.17}$	$\mathbf{71.36}_{0.75}$	$79.91_{0.75}$	$\mathbf{85.84}_{0.30}$	$52.52_{0.21}$	$\mathbf{68.16}_{0.36}$	$64.78_{0.47}$	$28.48_{0.01}$	$39.56_{0.03}$
	RANDOM-CORRUPTED	$62.70_{1.05}$	$48.73_{0.51}$	$70.71_{0.66}$	$79.65_{0.74}$	$85.26_{0.07}$	$53.14_{0.47}$	$62.70_{0.19}$	$59.17_{0.27}$	$27.09_{0.18}$	$39.08_{0.08}$
GPT-4	ZERO-SHOT	70.30	65.41	74.50	88.82	96.05	58.46	44.03	46.97	32.73	45.28
	TOP-K	76.53	67.45	**76.14**	91.23	**96.73**	61.68	72.44	**74.65**	**35.06**	**48.34**
	RANDOM	75.77	67.64	76.07	**91.59**	96.68	**62.36**	**73.21**	72.77	33.85	47.69
	RANDOM-CORRUPTED	**76.63**	**67.68**	75.50	90.73	95.55	61.46	63.61	65.80	32.61	47.05

Table 5: Performance of different types of demonstrations. For RANDOM and RANDOM-CORRUPTED, we report the mean and standard deviation across 3 seeds except for GPT-4. Best results for each model and dataset are boldfaced.

zero-shot performance. Overall, the base models are more sensitive to the type of demonstrations than chat models.

C.2 Results for individual languages

In Figure 10, we show the language-specific results for each task, in which we can see language discrepancies with different types of demonstrations.

D The interplay between demonstrations and templates

In this section, we provide supplemental results for Section 6.

D.1 Results for individual languages

We examine the effect of templates and show language-specific results for XCOPA, AfriSenti, XQuAD and TyDiQA in Figure 11. In a few cases, we found that formatting-focused templates lead to a decline in performance compared to original templates (e.g., Igbo and Mozambican Portuguese in AfriSenti with GPT-3.5). This can be attributed to the model's sensitivity to prompts, highlighting the potential of automatic prompt engineering. Still, formatting-focused template can largely narrow the performance gap between 0-shot and 4-shot in a broad context.

Task	Pattern	Verbalizer
NLI	{premise}, right? {label}, {hypothesis}	Yes ‖ Also ‖ No
PAWS-X	{sentence1}, right? {label}, {sentence2}	No ‖ Yes
XCOPA	{premise} {% if question == "cause" %}because{% else %} so{% endif %} {label}	{choice1} ‖ {choice2}
XStoryCloze	{input_sentence_1} {input_sentence_2} {input_sentence_3} {input_sentence_4} {label}	{sentence_quiz_1} ‖ {sentence_quiz_2}
AfriSenti	{tweet} The sentiment of the previous sentence is {label}	positive ‖ neutral ‖ negative
QA	{context}\nQ:{question}\nA:{answer}	{answer}
MT	{source_sentence} = {target_sentence}	{target_sentence}

Table 6: Prompting templates for XGLM and Llama 2 following Brown et al. (2020) and Lin et al. (2022).

Task	Pattern	Verbalizer
NLI	{premise} Based on the previous passage, is it true that {hypothesis}? Yes, No, or Maybe? {label}	Yes ‖ Maybe ‖ No
PAWS-X	Sentence 1: {sentence1}\n Sentence 2: {sentence2}\n Question: Can we rewrite Sentence 1 to Sentence 2? Yes or No? {label}	No ‖ Yes
XCOPA	{premise} {% if question == "cause" %}This happened because...{% else %} As a consequence...{% endif %}\n Help me pick the more plausible option:\n - {choice1}\n - {choice2}\n {label}	{choice1} ‖ {choice2}
XStoryCloze	{input_sentence_1} {input_sentence_2} {input_sentence_3} {input_sentence_4}\n What is a possible continuation for the story given the following options?\n - {sentence_quiz_1}\n - {sentence_quiz_2}\n {label}	{sentence_quiz_1} ‖ {sentence_quiz_2}
AfriSenti	{tweet} Would you rate the previous sentence as positive, neutral or negative? {label}	positive ‖ neutral ‖ negative
QA	{context}\nQ:{question}\nReferring to the passage above, the correct answer to the given question is:{answer}	{answer}
MT	Translate the following {src_language} text to {tgt_language}:\n {src_sentence}\n{tgt_sentence}	{tgt_sentence}

Table 7: Prompting templates for BLOOMZ and mT0 following Muennighoff et al. (2023) and Bach et al. (2022).

(a) XNLI

(b) IndicXNLI

7360
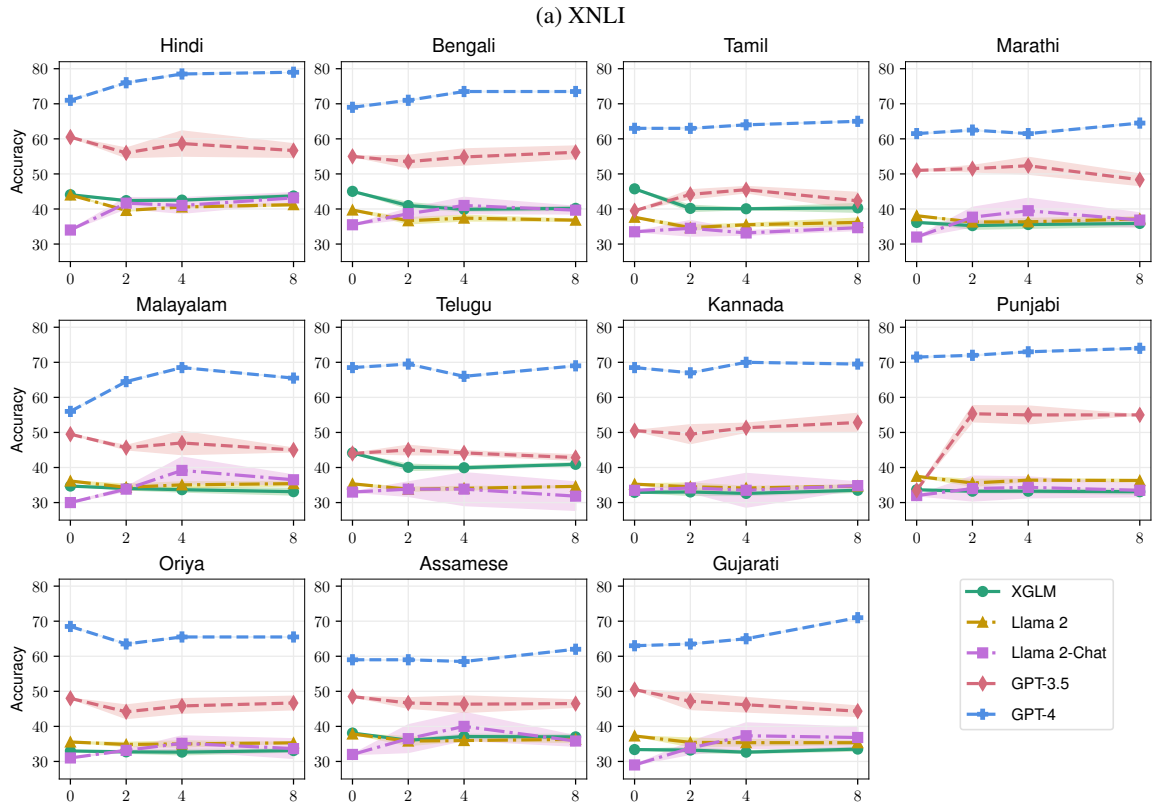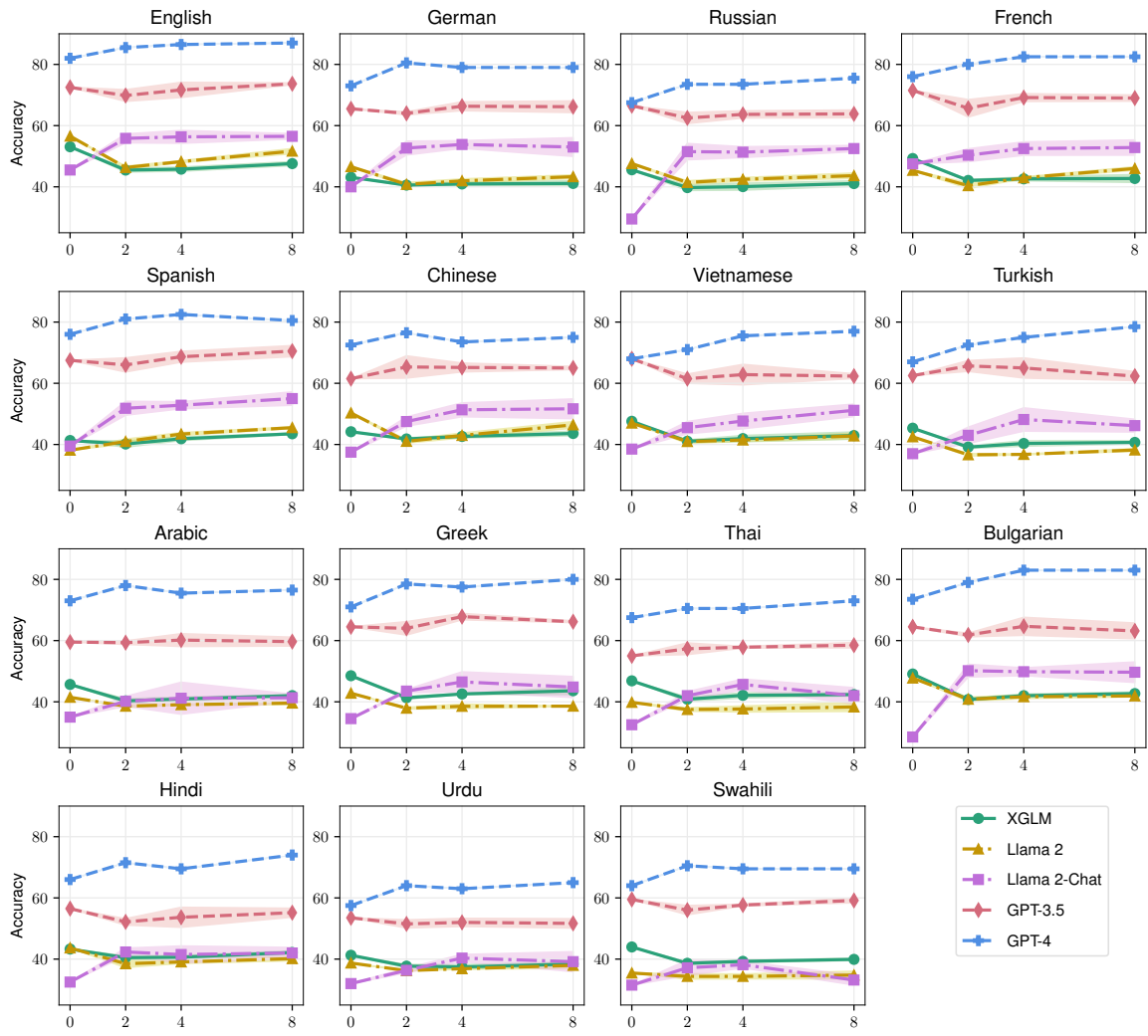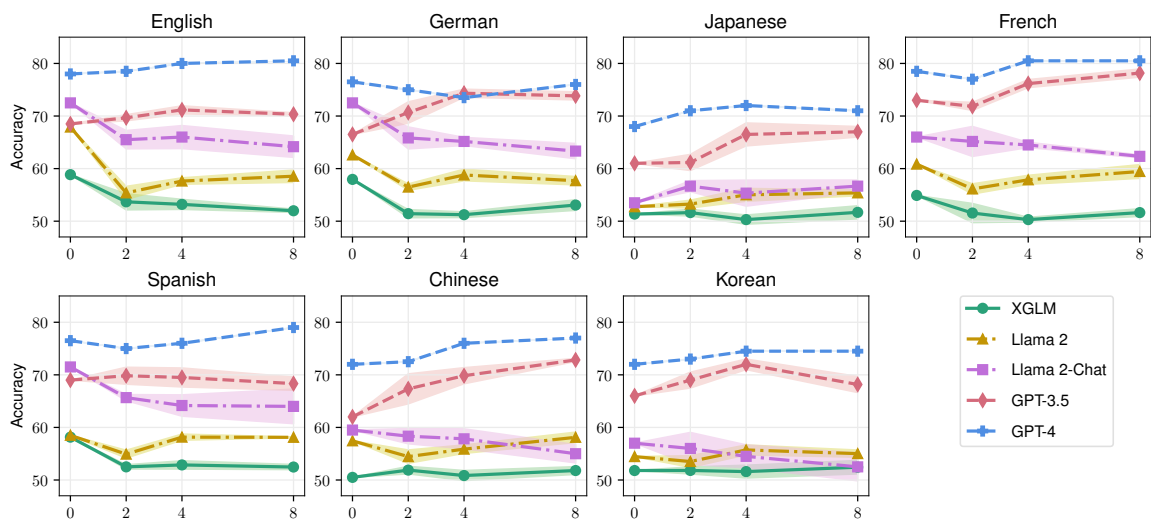
(c) PAWS-X
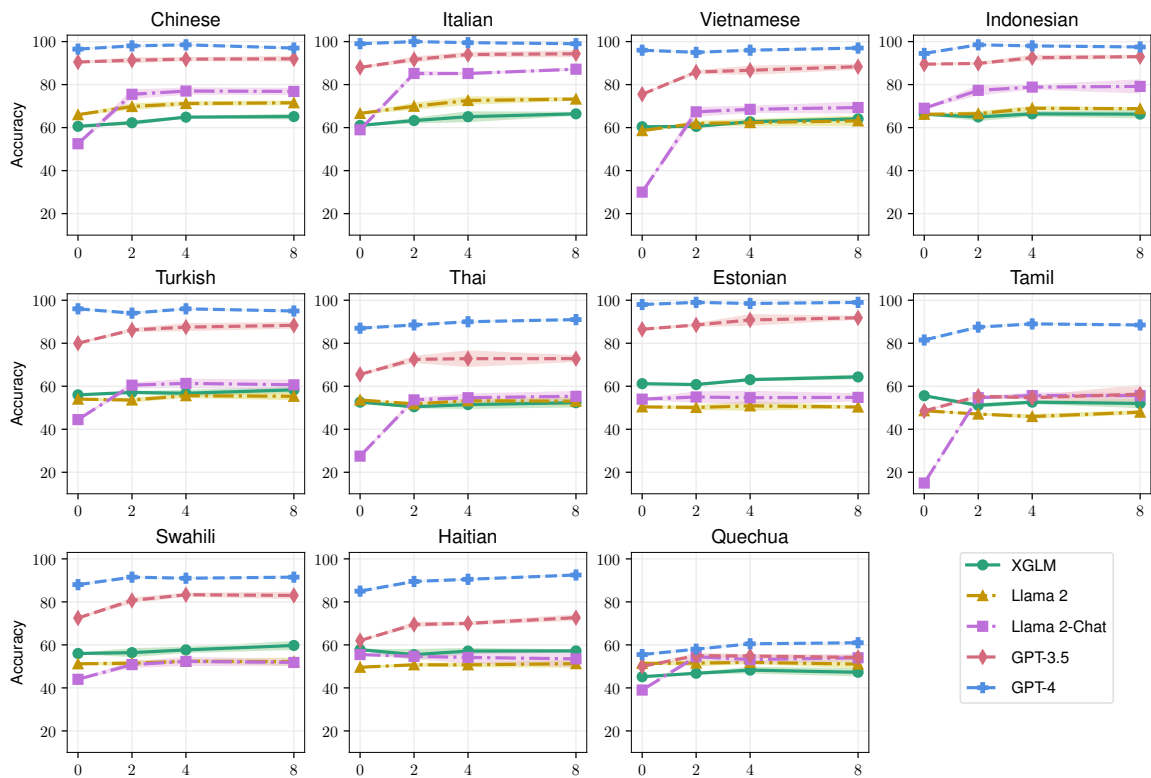
(d) XCOPA
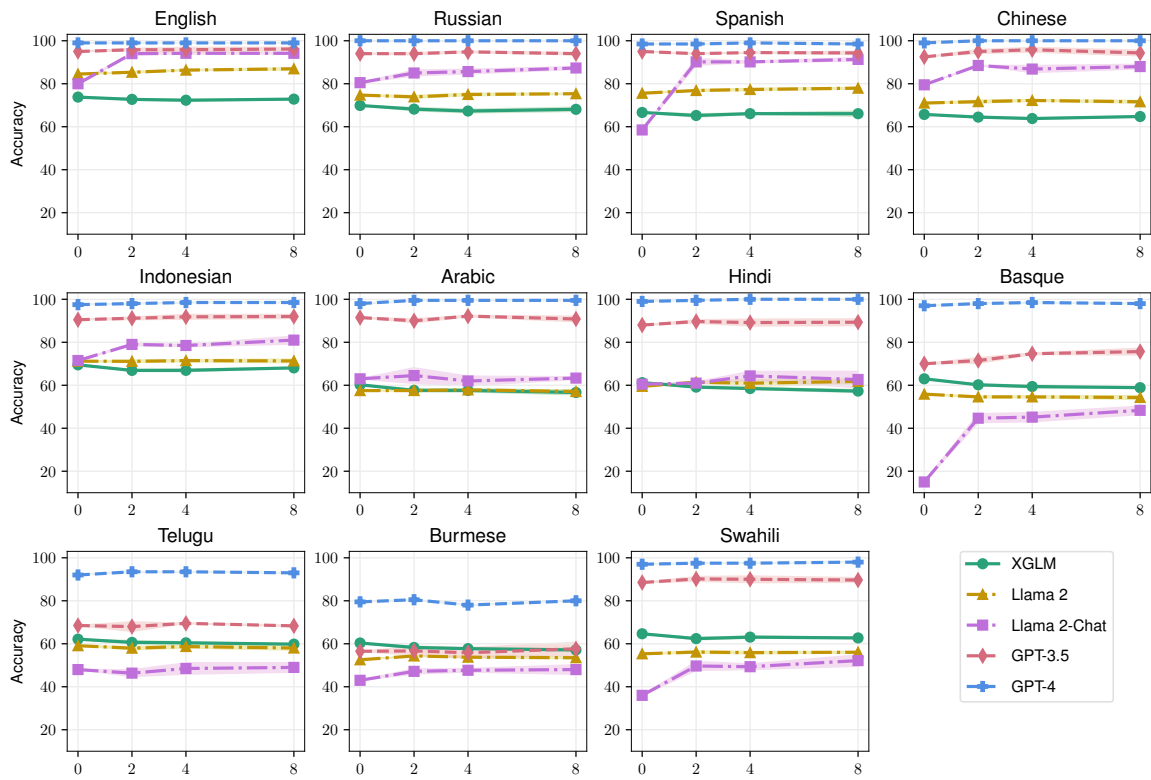
(e) XStoryCloze
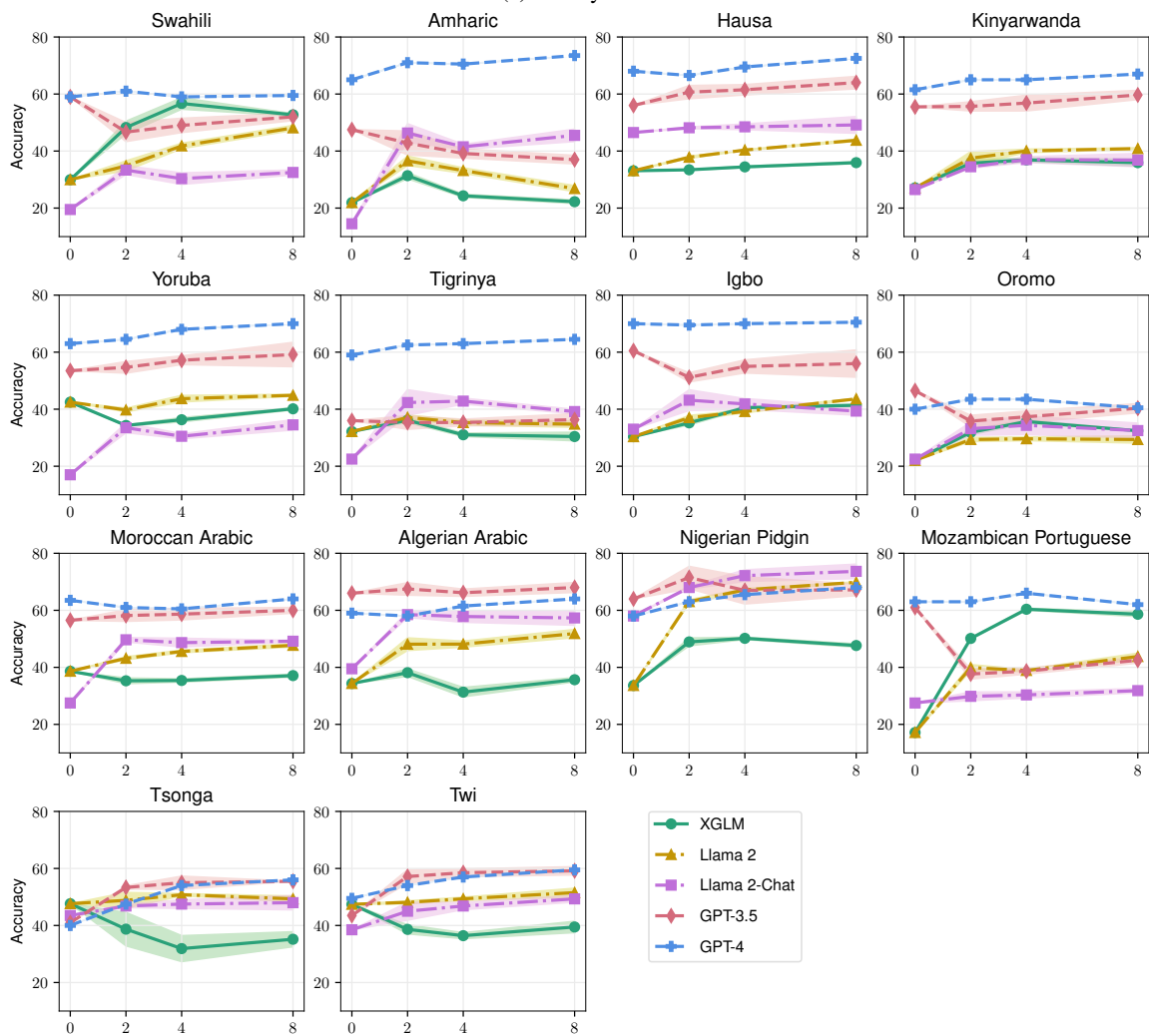
(f) AfriSenti

7362
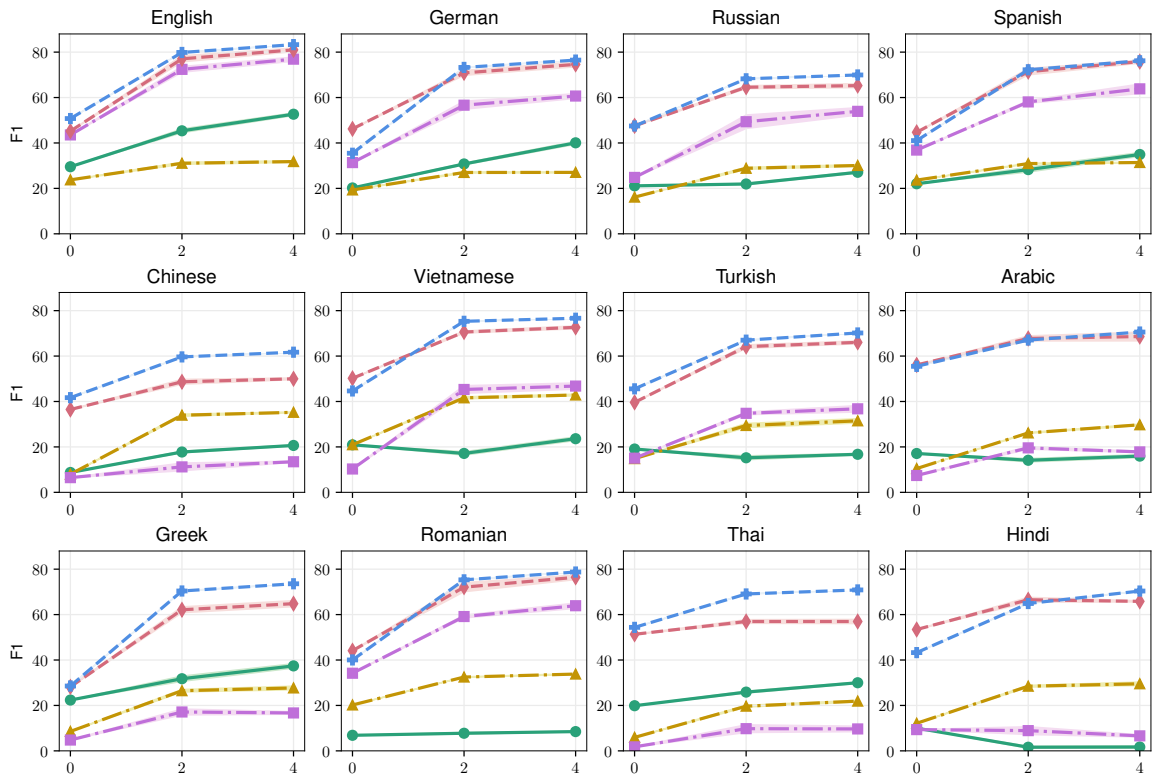
(g) XQuAD
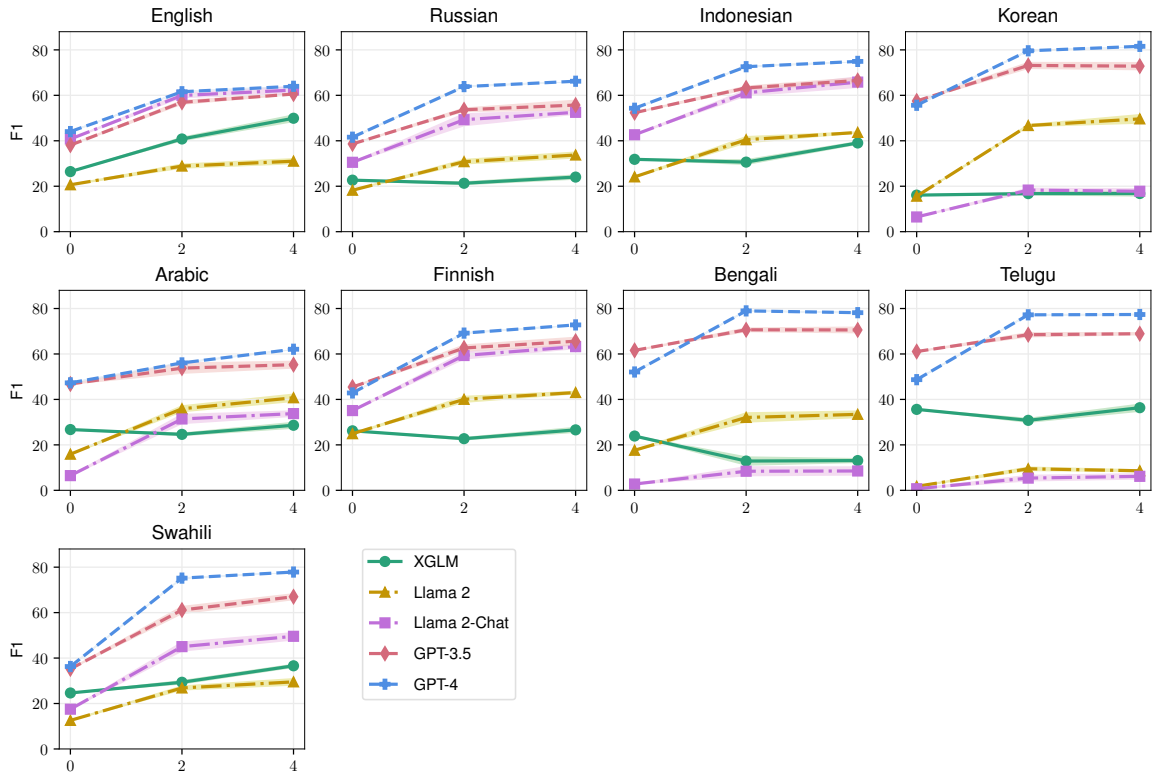
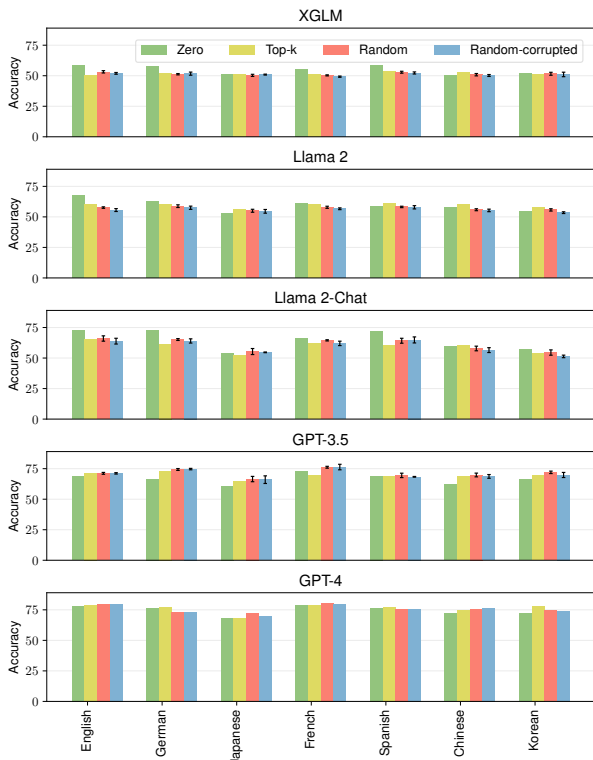(h) TyDiQA
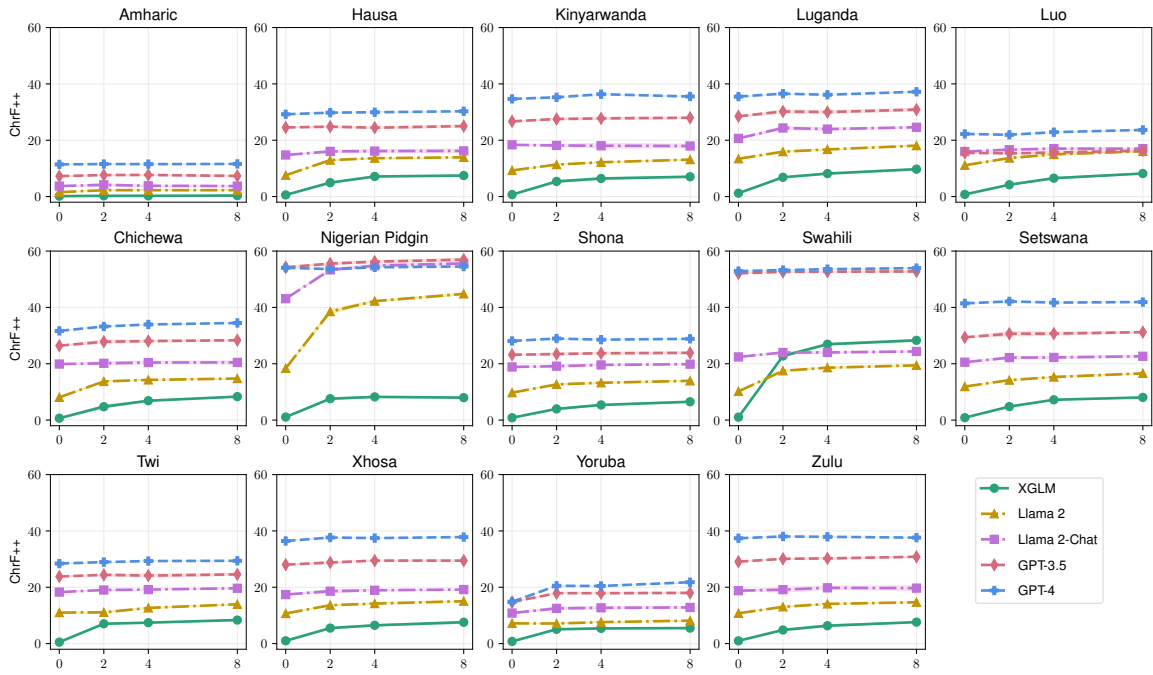
(i) MAFAND (en-xx)
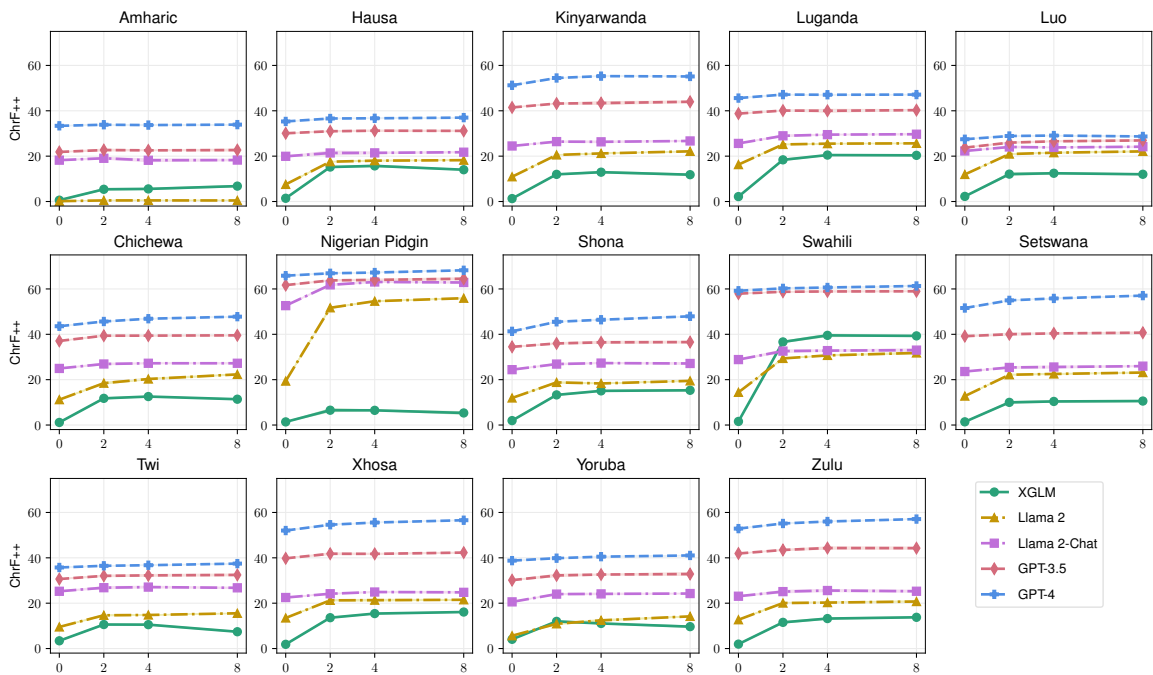
(j) MAFAND (xx-en)

Figure 9: Language-specific performance for both classification and generation tasks with different numbers of demonstrations. We average and report standard deviations over 3 seeds for all models except GPT-4.
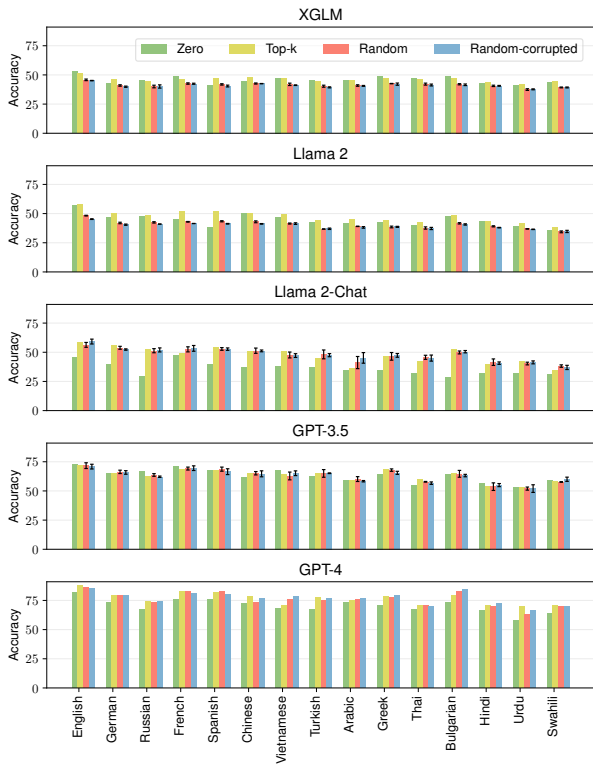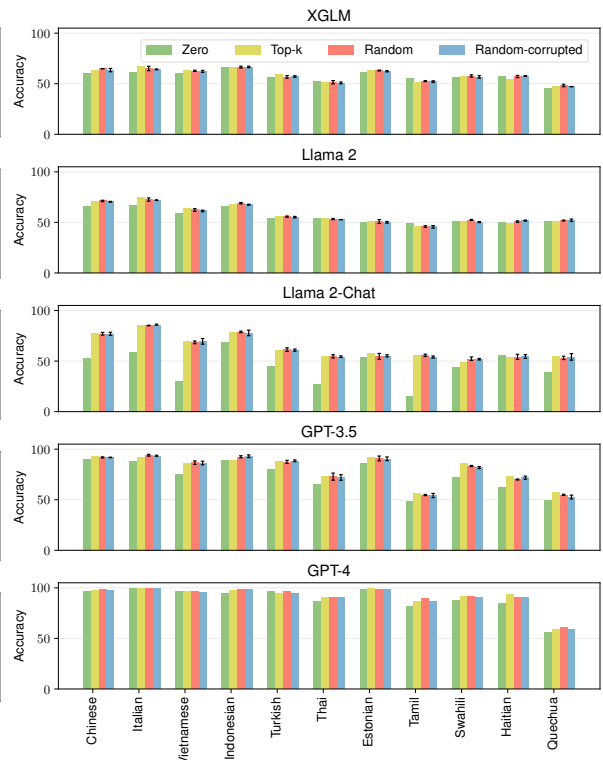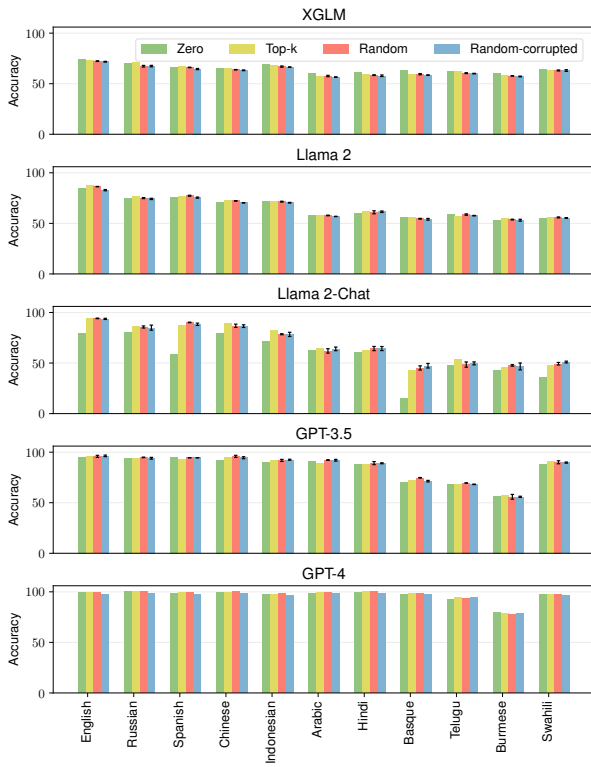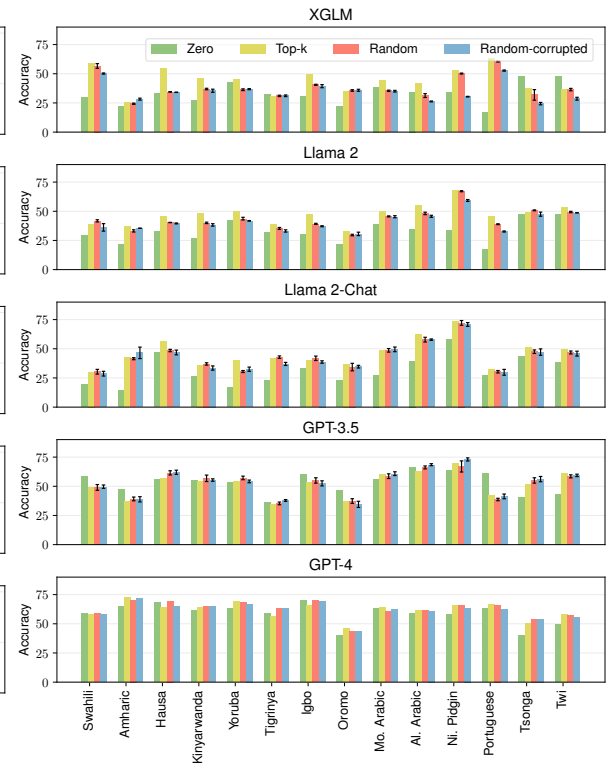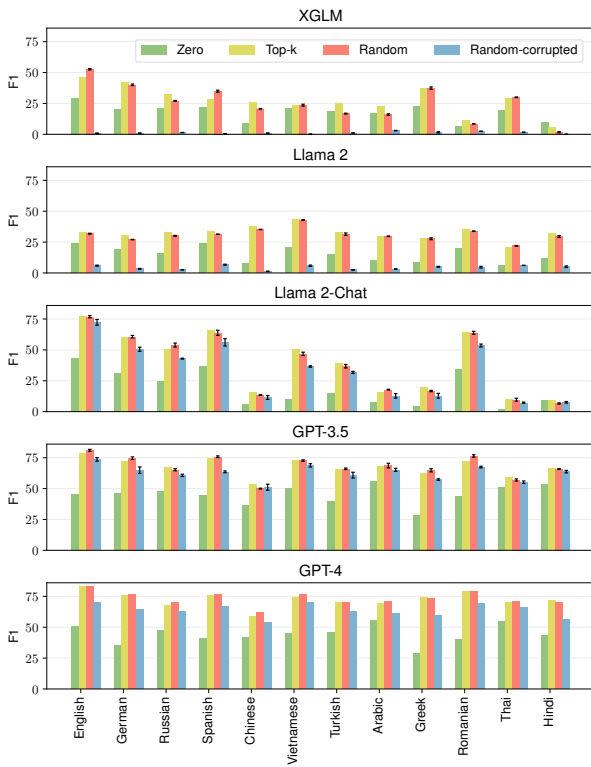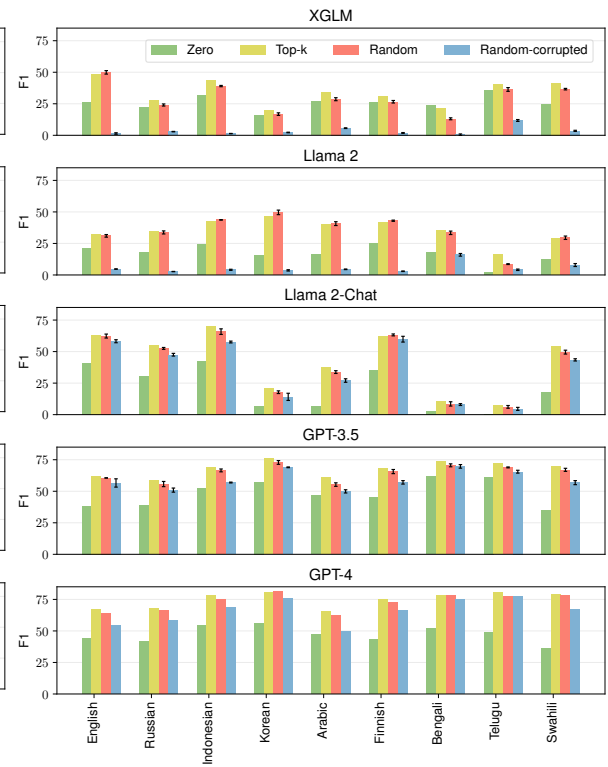
(a) XNLI
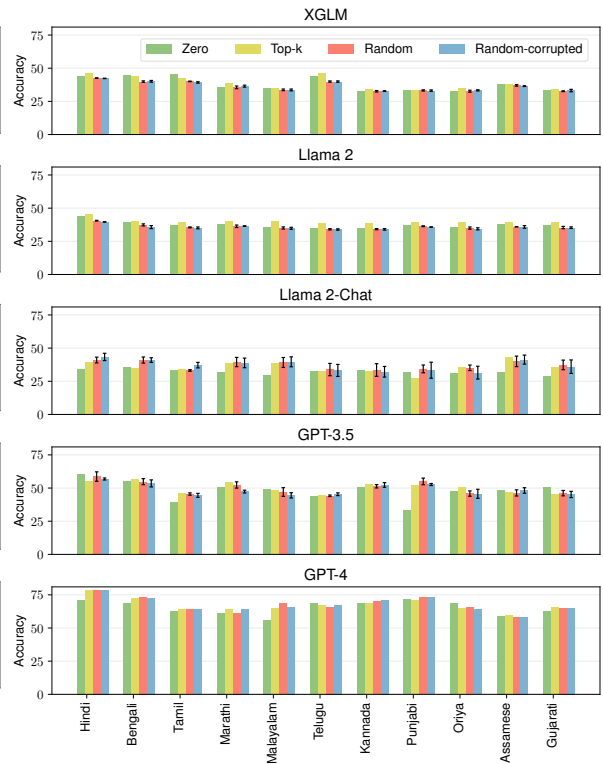
(b) IndicXNLI

(c) PAWS-X

(d) XCOPA
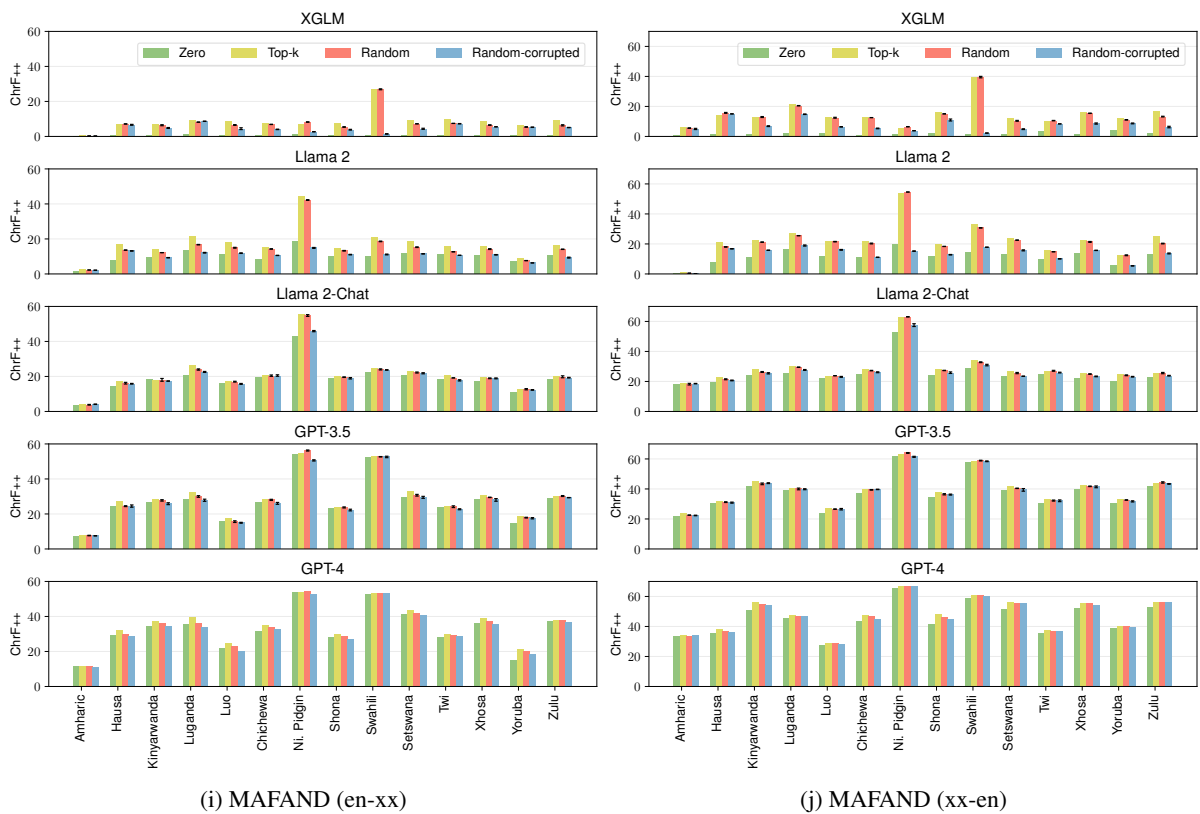
(e) XStoryCloze

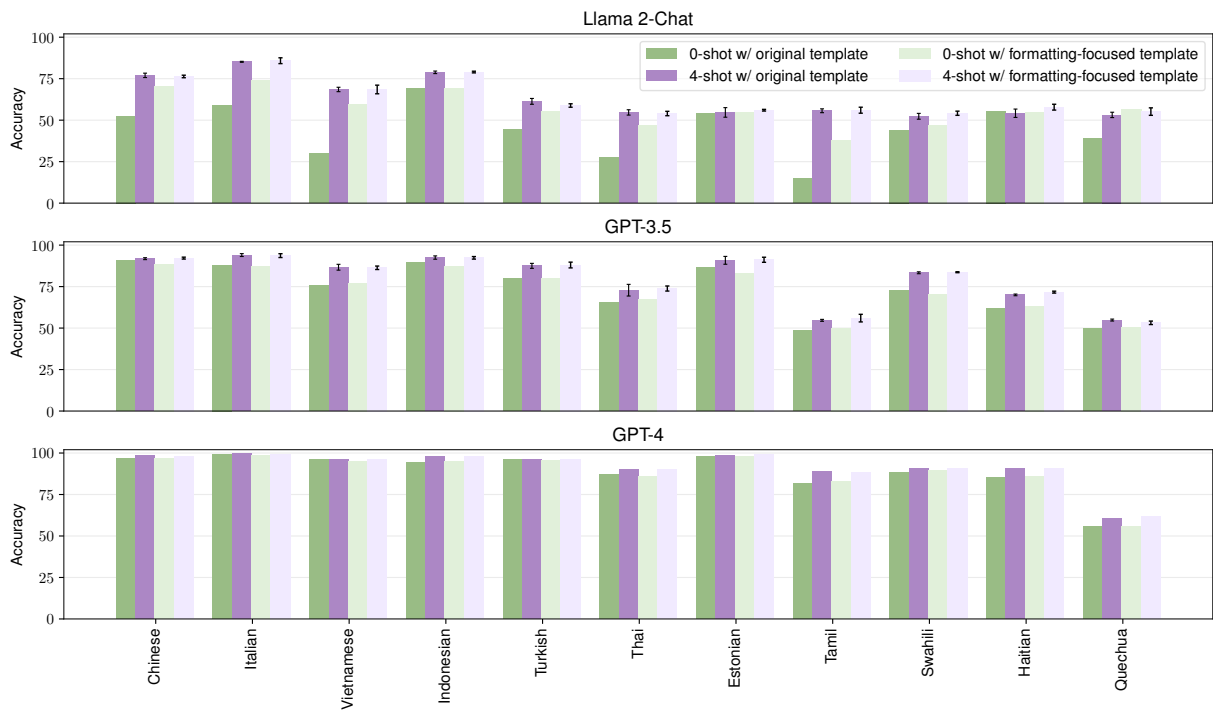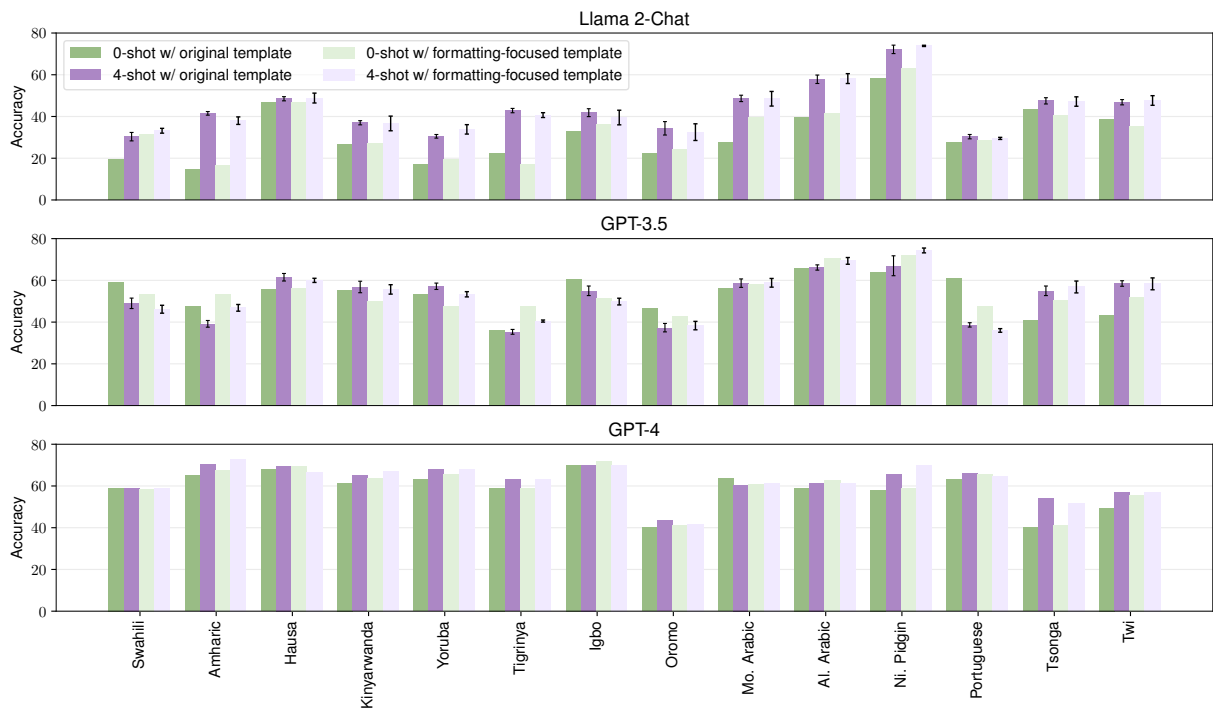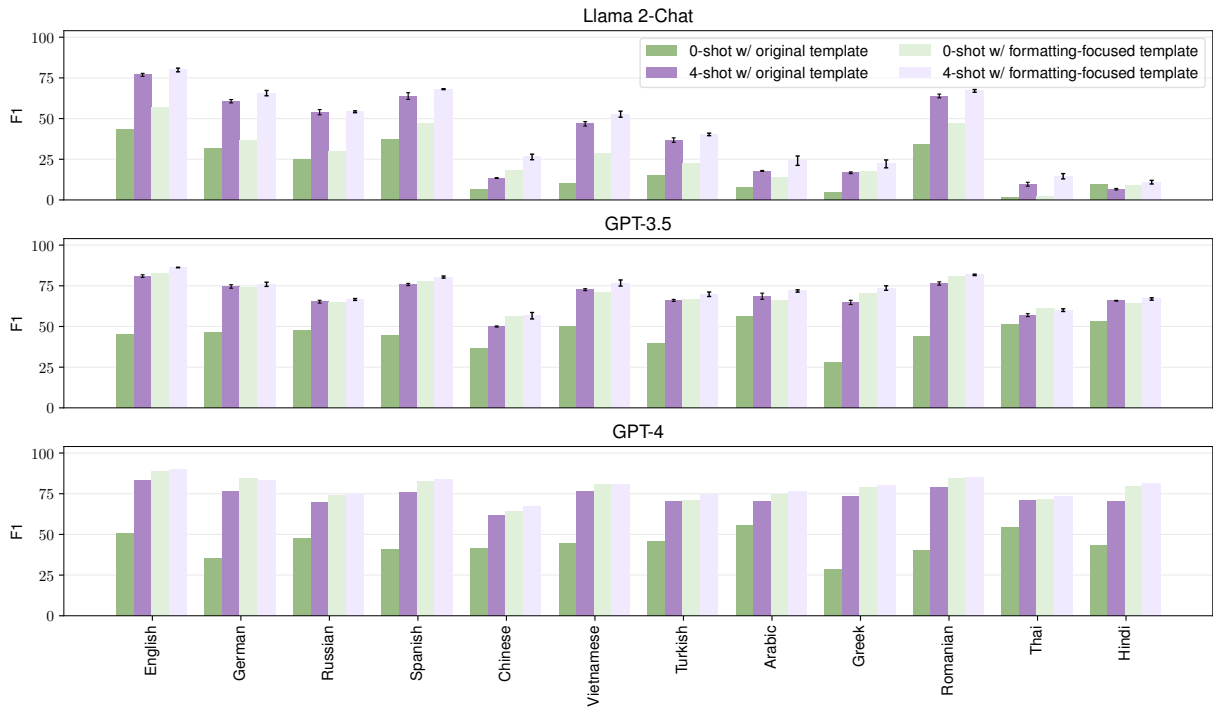(f) AfriSenti

(g) XQuAD

(h) TyDiQA

Figure 10: Language-specific performance of 4-shot ICL using different types of demonstrations. We average and report standard deviations over 3 seeds for all models except GPT-4.
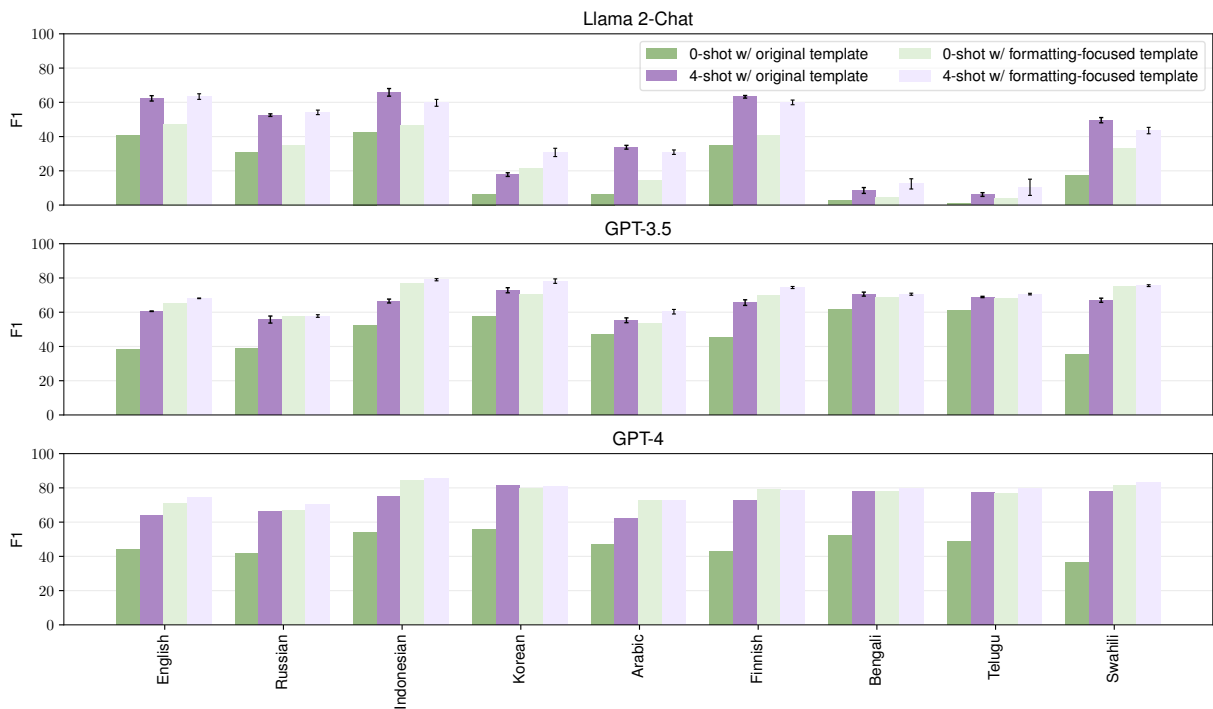
(a) XCOPA

(b) AfriSenti

7368

(c) XQuAD

(d) TyDiQA

Figure 11: Effect of using different templates on 0-shot and 4-shot performance for XCOPA, AfriSenti, and TyDiQA. Few-shot results are averaged across 3 seeds except for GPT-4.

Task	Template
NLI	**task instruction**: You are an NLP assistant whose purpose is to solve Natural Language Inference (NLI) problems in <EVALUATION_LANGUAGE>. NLI is the task of determining the inference relation between two (short, ordered) texts: entailment, contradiction, or neutral. Answer as concisely as possible in the same format as the examples below: **pattern**: {premise}\nQuestion: {hypothesis}\nTrue, False, or Neither? **verbalizer**: True ‖ Neither ‖ False
PAWS-X	**task instruction**: You are an NLP assistant whose purpose is to perform Paraphrase Identification in <EVALUATION_LANGUAGE>. The goal of Paraphrase Identification is to determine whether a pair of sentences have the same meaning. Answer as concisely as possible in the same format as the examples below: **pattern**: {sentence1}\nQuestion: {sentence2}\nTrue or False? **verbalizer**: False ‖ True
XCOPA	**task instruction**: You are an NLP assistant whose purpose is to perform open-domain commonsense causal reasoning in <EVALUATION_LANGUAGE>. You will be provided a premise and two alternatives, where the task is to select the alternative that more plausibly has a causal relation with the premise. Answer as concisely as possible in the same format as the examples below: **pattern**: Premise: {premise}\nWhat is the {question}? Pick the more plausible option:\n 1: {choice1}\n2: {choice2}\n You should tell me the choice number in this format 'Choice number:' **verbalizer**: Choice number: 1 ‖ Choice number: 2
XStoryCloze	**task instruction**: You are an NLP assistant whose purpose is to perform open-domain commonsense causal reasoning in <EVALUATION_LANGUAGE>. You will be provided a four-sentence story and two continuations, where the task is to select the correct ending. Answer as concisely as possible in the same format as the examples below: **pattern**: Story: {input_sentence_1} {input_sentence_2} {input_sentence_3} {input_sentence_4}\n What is a possible continuation for the story? Pick the more plausible option:\n 1: {sentence_quiz1}\n2: {sentence_quiz2}\n You should tell me the choice number in this format 'Choice number:' **verbalizer**: Choice number: 1 ‖ Choice number: 2
AfriSenti	**task instruction**: You are an NLP assistant whose purpose is to perform Sentiment Analysis in <EVALUATION_LANGUAGE>. Sentiment Analysis is the task of determining the sentiment, opinion or emotion expressed in a textual data. Give your answer as a single word, "positive", "neutral" or "negative". **pattern**: Does this statement "{tweet}" have a {positive neutral or negative} sentiment? Labels only **verbalizer**: positive ‖ neutral ‖ negative
QA	**task instruction**: You are an NLP assistant whose purpose is to solve reading comprehension problems in <EVALUATION_LANGUAGE>. You will be provided questions on a set of passages and you will need to provide the answer as it appears in the passage. The answer should be in the same language as the question and the passage. **pattern**: {context}\nQ: {question}\nReferring to the passage above, the correct answer to the given question is: **verbalizer**: {answer}
MT	**pattern**: Translate the following {src_language} text to {tgt_language}: {src_sentence} **verbalizer**: {tgt_sentence}

Table 8: Prompting templates for chat models following Ahuja et al. (2023) and Ojo et al. (2023). We add language identifiers in task instructions as it is an effective strategy for improving multilingual prompting (Huang et al., 2023).

Task	Template
XCOPA	**task instruction**: You are an NLP assistant whose purpose is to perform open-domain commonsense causal reasoning in <EVALUATION_LANGUAGE>. You will be provided a premise and two alternatives, where the task is to select the alternative that more plausibly has a causal relation with the premise. Answer as concisely as possible in the same format as the examples below: **pattern**: Premise: {premise}\nWhat is the {question}? Pick the more plausible option:\n 1: {choice1}\n2: {choice2}\n This is very important: Do not repeat the question and no explanation. You should tell me the choice number in this format 'Choice number:' **verbalizer**: Choice number: 1 ‖ Choice number: 2
AfriSenti	**task instruction**: You are an NLP assistant whose purpose is to perform Sentiment Analysis in <EVALUATION_LANGUAGE>. Sentiment Analysis is the task of determining the sentiment, opinion or emotion expressed in a textual data. Give your answer as a single word, "positive", "neutral" or "negative". **pattern**: Does this statement "{tweet}" have a {positive neutral or negative} sentiment? This is very important: Do not repeat the question and no explanation. Labels only **verbalizer**: positive ‖ neutral ‖ negative
QA	**task instruction**: You are an NLP assistant whose purpose is to solve reading comprehension problems in <EVALUATION_LANGUAGE>. Answer the question from the given passage. Your answer should be directly extracted from the passage and be a single entity, name, or number, not a sentence. **pattern**: {context}\nQ: {question}\nThis is very important: Your answer should be directly extracted from the passage and be a single entity, name, or number, not a sentence. **verbalizer**: {answer}

Table 9: Formatting-focused templates for chat models. We augmented the original templates in Table 8 with formatting-focused instructions.