Key ingredients for effective zero-shot cross-lingual knowledge transfer in generative tasks

Nadezhda Chirkova
Naver Labs Europe
Grenoble, France
nadia.chirkova@naverlabs.com

Vassilina Nikoulina
Naver Labs Europe
Grenoble, France
vassilina.nikoulina@naverlabs.com

Abstract

Zero-shot cross-lingual transfer, which implies finetuning of the multilingual pretrained language model on input-output pairs in one language and using it to make task predictions for inputs in other languages, was widely studied for natural language understanding but is understudied for generation. Previous works notice a frequent problem of generation in a wrong language and propose approaches to address it, usually using mT5 as a backbone model. In this work we compare various approaches proposed from the literature in unified settings, also including alternative backbone models, namely mBART and NLLB-200. We first underline the importance of tuning learning rate used for finetuning, which helps to substantially alleviate the problem of generation in the wrong language. Then, we show that with careful learning rate tuning, the simple full finetuning of the model acts as a very strong baseline and alternative approaches bring only marginal improvements. Finally, we find that mBART performs similarly to mT5 of the same size, and NLLB-200 can be competitive in some cases. Our final zero-shot models reach the performance of the approach based on data translation which is usually considered as an upper baseline for zero-shot cross-lingual transfer in generation.

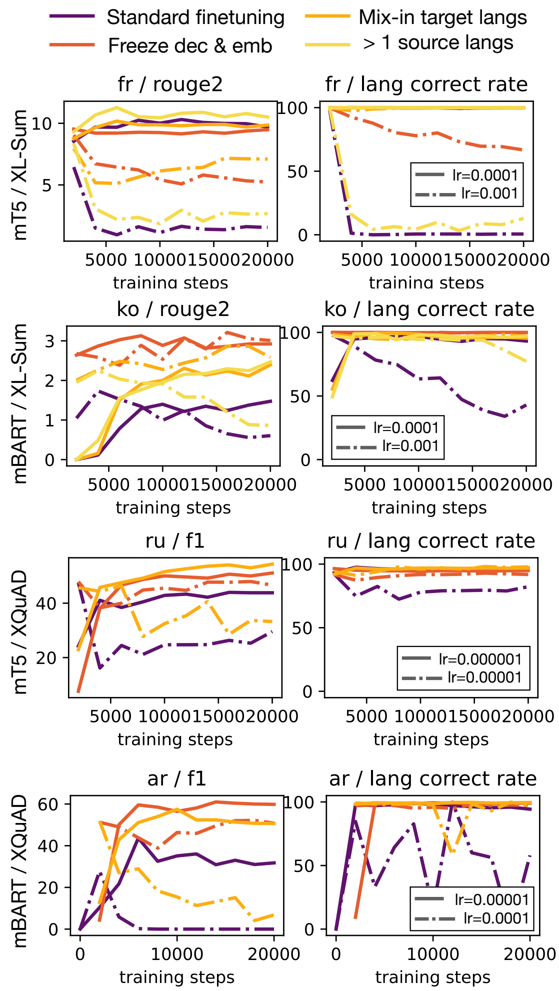

Figure 1: Learning rate plays a key role in cross-lingual transfer: decreasing LR almost completely eliminates generation in the wrong language with standard full finetuning, and often brings larger improvements that using complex adaptation methods developed to overcome this problem. Full results in Fig. 9–12 in Appendix.

1 Introduction

Multilingual pretrained language models (mPLMs) such as mBERT (Devlin et al., 2019), mBART (Liu et al., 2020), and mT5 (Xue et al., 2021) provide high-quality representations for texts in various languages and serve as a a universal backbone for finetuning on language-specific task data. The latter, however, is not always available for a language of interest, providing motivation for studying *zero-shot cross-lingual* capabilities of mPLMs. In this setting, the model is adapted, e.g. finetuned, on input-output pairs in a *source* language, usually English, and then applied in a zero-shot manner to make predictions for inputs in another *target* language, seen only at the pretraining stage.

While the described setting was broadly studied for natural language understanding tasks (Xue et al., 2021; Conneau et al., 2020; Artetxe et al., 2020a;

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 7222–7238
June 16-21, 2024 ©2024 Association for Computational Linguistics

Pires et al., 2019; Wu and Dredze, 2019; Pfeiffer et al., 2020), work on zero-shot cross-lingual transfer in *generation* is more limited (Vu et al., 2022; Pfeiffer et al., 2023; Maurya et al., 2021; Li and Murray, 2023). Previous work highlight two main problems arising in this scenario: producing incoherent or irrelevant answers, and generating text in a wrong language. A series of potential solutions were proposed, such as freezing the parts of the weights during finetuning, utilizing parameter-efficient finetuning methods, mixing-in the unsupervised target language data together with the supervised source language data, or using more than one source language. A common strategy is also to perform an intermediate tuning of the model on the language generation task in a self-supervised manner (as opposed to denoising tasks used for pretraining).

However, despite listed efforts, the state of zero-shot cross-lingual generation still remains unclear and poses open questions:

- *Which adaptation method is most effective?* Methods proposed for mitigating generation in the wrong language, were all tested on different tasks and benchmarks, and not compared to methods from other works, making it hard to establish the best performing one.

- *What makes a better mPLM for zero-shot cross-lingual transfer?* Different models have different pretraining objectives, training and architectural choices. How do these factors impact the quality of the cross-lingual transfer in generation?

- *Importance of hyperparameters in downstream task adaptation.* None of the previous work studied the impact of hyper-parameters used during downstream task adaptation for zero-shot cross-lingual transfer in generation.

- *Finally, if we pick the best solutions from all of the three listed dimensions, how far in performance can we get?* Can we reach the performance of a strong baseline, data translation, consisting in translating the train data into target languages? Previous studies either did not reach its performance or did not compare to this baseline.

The contribution of this work is conducting a deep empirical study addressing the listed questions. We consider most commonly used multilingual encoder-decoder mPLMs, namely mT5 and mBART, as well as the translation model NLLB-200. We systematically study six adaptation methods, investigate the effect of intermediate tuning, pay attention to adaptation hyperparameters, and compare models and adaptation methods *in a unified setting*. We consider two tasks: summarization and questions answering (QA). Our main findings are as follows:

- Hyperparameter tuning plays a very important role in cross-lingual transfer in generation: while related works report a severe problem of generation in a wrong language after full finetuning, we find that simply reducing the learning rate helps to alleviate this problem almost completely, without hurting performance;

- Intermediate tuning substantially improves performance in the majority of cases;

- With carefully chosen learning rates and intermediate tuning when necessary, simple full finetuning is a very strong baseline for zero-shot cross-lingual transfer in generation. Improvements brought by more advanced methods are quite modest, and none of the methods consistently outperform full finetuning in all cases. The notable methods are freezing the model decoder and embeddings, which performs consistently well with mBART (but not with mT5), and using more than one source language, which performs consistently well with mT5 (but not with mBART);

- mBART and mT5 of similar sizes lead to comparable performance. Qualitatively, due to the specifics of the masking pretraining objective, mBART is better suited for tasks with long outputs while mT5 is for tasks with short outputs;

- NLLB-200 is surprisingly competitive in summarization, reaching performance of mT5 and mBART for high-resource Latin-alphabet languages, but lags behind in QA;

- The final performance of the zero-shot approach is the same or superior to the performance of the data translation approach, often considered as an upper baseline for cross-lingual transfer in generation. Notably, careful learning rate tuning coupled with intermediate tuning allows the zero-shot approach closely approach the performance of data translation simply with the full finetuning adaptation.

2 Related Work

All works on zero-shot cross-lingual transfer in generation underline (and try to address) the severe problem of generating in a wrong language at the test time. This problem is also referred to under terms catastrophic forgetting (of languages not participating in finetuning, Vu et al., 2022), source language hallucination (Pfeiffer et al., 2023), or accidential translation problem (Li and Murray, 2023). Vu et al. (2022) propose to overcome generation in a wrong language by using parameter-efficient finetuning instantiated by prompt-tuning (Lester et al., 2021). They also mix-in the unsupervised target language task together with the supervised source language task, and factorize learnable prompts into language and task components.

Pfeiffer et al. (2023) propose mmT5 (modular mT5), allocating a small amount of language-specific parameters in the model during pretraining and freezing them during task-specific finetuning. To alleviate generation in a wrong language, they freeze some additional mmT5 parameters during finetuning, e. g. the embedding layer and feed forward layers in Transformer decoder. Li and Murray (2023) argue that learning language-invariant representations during finetuning is harmful for cross-lingual generation and propose finetuning on the data from more than one source language to avoid generation in a wrong language, with mT5 as a backbone model. ZMBART (Maurya et al., 2021) is the only work which considers the other backbone model than mT5: they perform an intermediate tuning of mBART on an auxiliary unsupervised task on Hindi, Japanese and English. To avoid generation in a wrong language, they freeze embeddings and the Transformer decoder, and mix-in the data from auxiliary pretraining during finetuning.

In our work we are interested to compare all previously proposed approaches in a unified setting, to better assess the impact of different factors on the zero-shot cross-lingual transfer in generation.

Alternative approaches to zero-shot cross-lingual transfer include data translation approaches, often referred as *translate-train* and *translate-test* paradigms. The former one implies translating the train task data to target languages and finetuning the model on this translated data, and the latter one implies translating test input examples into the source language, generating outputs in the source language and translating them back into the target language. The drawbacks of these approaches include a high computational cost either at training or testing time, lack of high-quality translation models for low-resource languages, and potential inconsistencies between sentences in translation (Vu et al., 2022). Despite its computational cost, data translation is a strong approach which is usually considered as an upper baseline for zero-shot approaches. Another related field is *few-shot cross-lingual transfer in generation* which assumes access to a small amount of labeled examples in the target language (Schmidt et al., 2022; Lauscher et al., 2020; Zhao et al., 2021). This setting is out of scope of this study, but could be considered in the future work.

3 Methodology and experimental setup

Adaptation methods. We investigate the following adaptation methods:

- *Full finetuning*: all weights of the model are finetuned on the source language data;
- *Prompt tuning* (Vu et al., 2022): comprises prepending several learnable vectors ("prompt") to the list of embeddings of the text input and freezing all other model weights during finetuning. Parameter-efficient approaches were shown in the literature to be better suited for transfer learning than full finetuning.
- *Adapters* (Houlsby et al., 2019; Bapna and Firat, 2019): lightweight tuned modules inserted after each fully-connected and attention block of Transformer, when the rest of (pretrained) model weights are frozen. We consider adapters as the most widely used parameter-efficient adaptation approach in the literature;
- *Freezing of the decoder and embeddings* (Maurya et al., 2021): only weights in the encoder are finetuned. The motivation behind this approach is that the decoder should preserve capabilities of generating in various languages while the encoder will adapt the model to the task;
- *Mixing-in self-supervised data for target languages* (Lester et al., 2021; Maurya et al., 2021): during finetuning, task data instances in the source language will be alternated with self-supervised data instances in target languages. The motivation is that such a mixing will preserve model's capability of generation in target languages;
- *Using several source languages* (Li and Murray, 2023): performing finetuning on more than one

source language to better decouple task knowledge from language knowledge.

In the rest of the text term "full finetuning" refers to the finetuning of all weights on the English task data only, even though two last described methods also finetune all weights. We do not consider mmT5 as it was not publicly released and requires substantial resources for pretraining.

We also experiment with *intermediate tuning* (IT) of the model, used in several works and performed before finetuning on the task data. Standard encoder-decoder mPLMs rely on a self-supervised denoising training, where often the input corresponds to corrupted text (eg. with masked tokens or permuted sentences), and the output can follow some very specific structure (eg. a masked span rather than a full sentence, output containing special tokens, etc.). Therefore, in their raw form, these mPLMs are not necessarily well suited to receive well-formed text as an input and generate clean text as an output. IT performs finetuning on a language modeling-like task, e.g. predicting the continuation of a paragraph based on its beginning, to compensate for this gap. IT was shown to be necessary in Vu et al. (2022) for prompt tuning of mT5 and in Maurya et al. (2021) for full or partial finetuning of mBART. We systematically test the necessity of IT for all methods and models.

Models. We focus on encoder-decoder mPLMs as they are well suited for generation purposes, as opposed to encoder-only mPLMs such as mBERT or XLM-R. We leave the investigation of decoder-only mPLMs such as BLOOM (Scao et al., 2022) for future work. We consider mT5 and mBART as two most widely used mPLMs and NLLB-200 as a high-quality translation model:

- *mT5*: pretrained using the masked language modeling objective where parts of the input sequence are masked and the missing fragments act as targets[1]. mT5 is pretrained on the mC4 corpora, supports 101 languages, and does not use any language codes. Among released sizes from 300M to 13B we experiment with mT5-base (580M, most of the experiments) and mT5-Large (1.2B, additional experiment).
- *mBART (pt)*: pretrained using the denoising objective where parts of the input sequence are masked and the entire original sequence acts as a target (Liu et al., 2020; Tang et al., 2021). mBART is pretrained on Common Crawl (Conneau et al., 2020) corpora, supports 50 languages, has 680M parameters in total and uses language codes in both encoder and decoder sides. Both input sequence `X` and target sequence `Y` are prepended with the language code: `[lang_code, X]` and `[lang_code, Y]`, and at the inference time `lang_code` is forced as a first generated token. Our hypothesis is that the use of the language code in the decoder can help to alleviate the problem of generation in a wrong language.
- *mBART (tr)*: In addition to the *pretrained* version, we also consider mBART finetuned for *translation* (Tang et al., 2021).
- *NLLB-200*: translation model supporting 200 languages, pretrained on sentence-level data mined from the web and automatically paired using multilingual embeddings. NLLB-200 uses the same language code scheme as mBART and is released in various sizes from 600M to 54.5B, among them we consider 600M (distilled version). Our hypothesis is that translation-based pretraining may provide good representations for cross-lingual transfer as suggested by (Reid and Artetxe, 2023).

Evaluation. We select two generative tasks to evaluate cross-lingual zero-shot knowledge transfer:

- *XL-Sum*: news summarization on the XL-Sum dataset (Hasan et al., 2021). The model needs to generate a 1–2 sentences summary based on a 1–2 news paragraphs. The evaluation is performed with ROUGE-2 metric (Lin, 2004) computed on the test sets (first 2k examples per language).
- *XQuAD*: question answering dataset (Artetxe et al., 2020b), the model needs to generate a short phrase answer based on a paragraph and a question about it appended in the end of the paragraph. The evaluation is performed with F-measure comparing tokens in the gold answer and the model-generated answer, computed on publicly available development sets. For better metrics interpretability, we only consider questions for which groundtruth answers do not contain numbers and are correctly identified to be written in the target language.

[1]In contrast to English-centric T5, mT5 did not include supervised tasks in pretraining.

We select a representative subset of languages for each task[2], covering Latin- and non-Latin scripts, and report how do task-specific metrics evolve during adaptation. For better interpretability, in addition to the task metrics, we also consider (1) *lang. correct rate* metric (the percentage of outputs generated in the correct target language) and (2) *average sequence length* metric that allows us to spot some edge behaviour of the models.

Adaptation settings. For all adaptation methods we train models on English data for 20k steps with a batch size of 4000 tokens on a single A100 GPU, and run evaluation each 2k steps. We crop input sequences to the maximum length supported by models, which equals to 512 (mT5, NLLB-200) or 1024 tokens (mBART). We grid search the learning rate (LR) for each task-model-adaptation method combination, details are given below.

For *Intermediate tuning* (IT) we finetune models for 100k steps on the CommonCrawl data uniformly sampled across all target languages and English, with the batch size of 5k tokens and the LR chosen to optimize fluency of model generations, inspected manually. We use PrefixLM-inspired self-supervision from (Vu et al., 2022), where the continuation of the text needs to be predicted based on its beginning. It has shown more promising results in our preliminary experiments compared to self-supervised objective from (Maurya et al., 2021) (see details in Appendix B).

- *Prompt tuning*: we use the prompt dimension of 100 and initialize the prompt with randomly selected rows of the embedding matrix, following Vu et al. (2022).

- *Adapters*: we use the adapter dimension of 64 and insert adapters after each attention and fully-connected layer, following Bapna and Firat (2019).

- *Mixing-in target languages*: we use the same self-supervised objective as in IT and sample the corresponding data with probability 1% (all languages represented uniformly within this 1%), following Vu et al. (2022). We experimented with higher portions in Appendix C, as well as with mixing-in the pretraining task of the base model, and found that they lead to worse results.

- *Using several source languages*: we test this approach only on XL-Sum, because for XQuAD only English training data is available; for XL-Sum we use English, Japanese and Arabic, selecting them uniformly when forming mini-batches.

Hyperparameter tuning. We tune LR and decide on the necessity of IT, for each considered task-model-adaptation method combination. We initially grid searched LR for full finetuning, adapters and prompt tuning, for each task and model, without IT. The result of this step is the preliminary LR (PLR), and we utilize the PLR of full finetuning for other adaptation methods since they are also based on full finetuning. PLR usually corresponds to the highest LR which still enables generation in the correct language. After finding PLR, for each task-model-adaptation method combination, we select the best of four hyperparameter combinations: two options for LR (PLR and PLR $\times 10$) and two options for IT (used or not). Our intuition is that the use of advanced adaptation methodology or IT could potentially increase the LR which still does not lead to generation in the wrong language. In practice, this happened only once, for freezing of mBART in the summarization task.

For XL-Sum, we perform the described tuning on the validation sets, looking at the performance averaged over considered target languages, while the main evaluation is performed on the test sets. For XQuAD, only validation sets are publicly available so we perform tuning using held-out languages (Thai, Romanian, and Vietnamese). In Appendix D we show that performance on validation sets in target languages correlates with performance on validation sets in held-out languages and validation sets translated from English into target languages. This demonstrates that having validation sets in target languages is not necessary in practice which is important to enable fully zero-shot setting.

We report the resulting optimal settings in Table 3 in Appendix. We could not find information on the used LR in (Pfeiffer et al., 2023) and (Vu et al., 2022), to compare our chosen LRs with theirs. Maurya et al. (2021) and Li and Murray (2023) use a constant LR for all tasks, which are hard to compare to ours because of different data[3].

More details on the experimental setting are given in Appendix A.

[2]XL-Sum: Chinese, French, Korean, Russian, and Spanish. XQuAD: Arabic, Chinese, German, Russian, and Spanish

[3]Maurya et al. (2021) use LR=3e-5 larger than ours 1e-6, Li and Murray (2023) use LR=7e-5 close to ours 1e-4.

4 Experiments

First, we investigate the effect of the learning rate, intermediate tuning and the adaptation method for two most commonly used models, mT5 and mBART. Second, we compare them with other models and consider larger models. Finally, we present some qualitative examples and observations from manual inspection of predictions. In general, model predictions reaching highest metric values in our plots, form quite meaningful and reasonable responses to the considered tasks; more details in Section 5.

Effect of learning rate. We begin our study with analysing the effect of LR on the full finetuning on the English task data. With too small or too large LR the model does not learn even the English task because of too short steps or divergence. For the range of LRs when the English task is learned well, we observe that larger LRs lead to the effect reported in other works, when the model overfits to the source English language and generates answers in English when applied for inputs in other languages. However, *with the reduced LR, this effect almost completely eliminates and the model mostly generates in the target language*. This effect is demonstrated in Figure 1 on a subset of languages and in Fig. 9–12 in Appendix on all considered languages.

Figure 1 also shows a comparison of enhancements of full finetuning proposed in the literature, such as mixing-in target languages or freezing the decoder and embeddings. Even though these enhancements improve performance and percentage of outputs in the correct language with fixed LR, we find that *reducing LR in full finetuning often brings larger improvements*. Reducing LR for other methods makes them even stronger.

We note that performance in English is usually a little higher with larger LR. This may raise a hypothesis that for non-English languages, outputs generated with larger LR in English may be of higher semantic quality than the ones generated in the correct target language with smaller LR. In Appendix E we test this hypothesis and demonstrate that this is not the case.

Effect of intermediate tuning. For each combination of a task and an adaptation method, we compare the mT5-base/mBART task adaptation with and without intermediate tuning (IT).

We choose the best LR between PLR and PLR

	XL-Sum		XQuAD	
Method	mT5	mBART	mT5	mBART
Full finetuning	+0.1	+2.5	+6.3	+9.0
Ft + mix tgt langs	0	+0.6	+3.1	-8.3
Ft + >1 src langs	0	+1	n/a	n/a
Freeze emb & dec	+4.3	+4.1	+11.2	+1.3
Adapters	0	0	+1.0	+3.9
Prompt tuning	+7.5	+7.2	+26.8	+25.1

Table 1: Difference in performance between task adaptation with and without intermediate tuning, for various methods. Rouge-2 for XL-Sum, F-measure for XQuAD. *Main conclusion:* intermediate tuning brings performance improvements in the majority of cases, in almost all the rest cases it does not affect performance.

$\times 10$ (section 3). Results are presented in Table 1. We observe that *intermediate tuning substantially increases performance in the majority of cases*. In particular, IT appears to be essential for mBART with almost all adaptation methods and in all tasks, and important for mT5 in question answering. For mT5 in summarization, the use of IT does not increase performance, except with prompt tuning and freezing methods. We believe this is because these two approaches do not modify the decoder, which was trained only on masked spans targets during mT5 pretraining and was never exposed to realistic text targets, and IT closes this gap. This result is consistent with (Vu et al., 2022) and (Maurya et al., 2021).

Comparison of adaptation methods. Figure 2 shows results (averaged over target languages) comparing adaptation methods for mT5-base and mBART models. Detailed per-language results are presented in Figure 8 in Appendix.

We observe that *with carefully chosen learning rates and intermediate tuning, simple full finetuning is a very strong baseline for zero-shot cross-lingual transfer in generation*. Improvements brought by the use of more advanced adaptation methods are rather modest, and *none of the adaptation methods consistently outperform full finetuning in all cases*. *The notable approach for mBART is freezing the decoder and embeddings*, proposed by Maurya et al. (2021) for this base model: freezing consistently outperforms full finetuning in all target languages in both tasks. However, this approach does not show such improvements for mT5. *For XL-Sum, using more than one source language proposed in (Li and Murray, 2023) for mT5, brings consistent improvement over target*

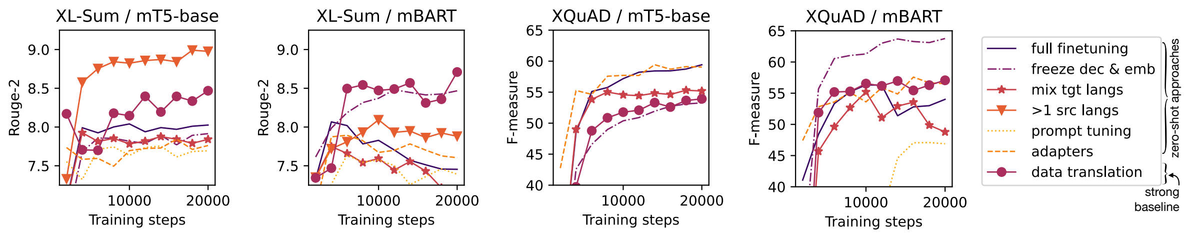

Figure 2: Comparison of adaptation methods, with tuned learning rates and intermediate tuning when it is needed. Results averaged across target languages and 2 runs. Language correct rate is close to 100% in almost all cases, due to hyperparameter tuning. The exception is prompt tuning of mT5 in the XQuAD task which is not shown because of too low performance. *Main conclusions*: (1) Straightforward full finetuning is a strong approach which reaches or approaches the performance of data translation in all cases. (2) None of other approaches outperform full finetuning *consistently* in all cases: using several sources languages works well for mT5 but not for mBART and freezing decoder works well for mBART but not mT5. (3) One of zero-shot approaches reaches or outperforms a strong and computationally expensive baseline, data translation, in all cases.

	XL-Sum		XQuAD	
Method	R2	LCR	F1	LCR
Large / IT + ft	9.9	99.8%	69.8	94.7%
Large / IT + ft >1 src lg	10.9	99.8%	n/a	n/a
Large / Data translation	10.8	99.8%	63.6	96.7%
Base / IT + ft	8.0	99.7%	59.4	92.9%
Base / IT + ft >1 src lg	9.0	99.8%	n/a	n/a
Base / Data translation	8.5	99.6%	53.9	95.3%

Table 2: Results for mT5-large model, averaged over target languages. Metrics: Rouge-2 for XL-Sum, F-measure for XQuAD, LCR: language correct rate. LCR is lower than 100% on XQuAD (partly) because of language identification errors for short sequences.

languages when used with mT5. For mBART this approach performs on par with using one source language. The obvious drawback of this approach is that multi-lingual data may be not available, e.g. this is the case for XQuAD.

Mixing-in unsupervised tasks for target languages often degrades performance and increases the length of predictions, see Appendix C. Prompt tuning often has difficulties learning an English task and substantially underperforms other adaptation methods on XQuAD. Adapters usually perform on par or slightly worse than full finetuning.

Comparison of models. Figure 2 allows us to compare mT5-base and mBART after tuning of hyperparameters and adaptation methods. These models incorporate comparable numbers of parameters. We observe that *mT5 and mBART reach the close level of performance in both tasks*. The same conclusion holds if we simply compare full finetuning runs of both models.

In Figure 3 we compare all four models we consider, adapted using full finetuning. We compare models without intermediate tuning, to avoid hindering model capabilities behind this additional step. We find that *translation-pretrained NLLB-200 performs well in summarization*, achieving performance of mT5 and mBART in Latin-language high-resource languages, French and Spanish, and performing on par with mBART without intermediate tuning in other languages[4]. We selectively inspected the predictions of NLLB and found that they indeed form meaningful summaries. However, in QA, NLLB-200 performs poorly, often (but not always) generating non-relevant answers. Translation-finetuned version of mBART performs poorly in all tasks, generating a lot of wrong language predictions.

Interestingly, the results do not support our initial hypothesis that the translation pretraining objective used in NLLB and architectural choices such as the use of language codes in mBART, could improve zero-shot knowledge transfer in generation.

Comparison versus data translation. Figure 2 also shows comparison versus the data translation[5] approach, when English training data is translated into target languages using the NLLB-3.3B model. We translate data sentence-by-sentence and grid search the LR for finetuning. The results show that after careful tuning, *the zero-shot approach reaches or outperforms the data translation approach in both considered tasks*. If we consider a simpler setting when only LR and the use of IT are tuned, i.e. comparing full finetuning and data translation

[4]Expect Chinese, for which NLLB-200 generates a lot of empty predictions. NLLB-200 was noticed previously in the literature to have issues with processing Chinese.

[5]Data translation is often referred as translate-train method.

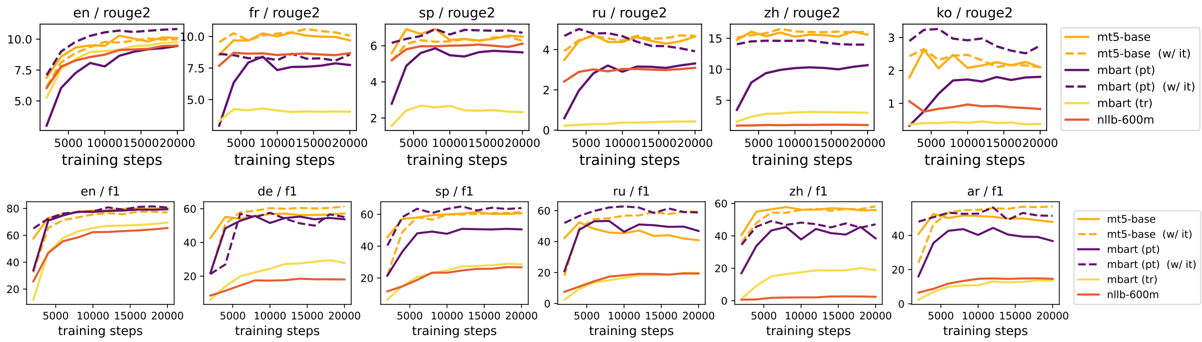

Figure 3: Comparison of base models with full finetuning. Each plot averaged over 3 runs. Correct language rate is close to 100%, due to hyperparameter tuning, in almost all cases except the translation-tuned version of mBART. pt: pretrained version of mBART, tr: translation-finetuned version of mBART. *Main conclusion:* mBART and mT5 of similar sizes perform on par; NLLB performs well in summarization for Latin-alphabet languages.

	Model	Model outputs in French	Translation into English	Avg len
1	Grountruth	Pr Hulda Swai, professeure tanzanienne de sciences de la vie et de bio-ingénierie, a été désignée lauréate 2020 du prix continental Kwame Nkrumah de l'Union africaine (UA) pour l'excellence scientifique.	Pr. Hulda Swai, a Tanzanian female professor of life sciences and bioengineering, has been named the 2020 winner of the African Union (AU) Kwame Nkrumah Continental Prize for Scientific Excellence.	151
2	mT5-base, full ft w/o IT	<extra_id_0> femmes sont très motivées à entreprendre une carrière scientifique dans le domaine de la nanotechnologie.	<extra_id_0> women are highly motivated to pursue a scientific career in the field of nanotechnology.	101
3	mT5-base, full ft with IT	La professeure de nanotechnologie de l'Union africaine a réussi à réunir 7,5 millions de dollars.	The nanotechnology female professor from the African Union managed to raise $7.5 million.	91
4	mT5-base, >1 src langs w/o IT	<extra_id_0> une scientifique africaine a reçu une récompense pour son travail dans le domaine de la nanotechnologie.	<extra_id_0> an African female scientist has received an award for her work in the field of nanotechnology.	107
5	mT5-base, train data translation	Un professeur de nanotechnologie a reçu une récompense continentale pour son travail dans le domaine des nanotechnologies.	A nanotechnology professor has received a continental award for her work in the field of nanotechnology.	112
6	mBART, full ft w/o IT	Ancienne professeure de l'université de Durban a reçu un prix de la part de la Banque mondiale.	Former professor at the University of Durban received an award from the World Bank.	117
7	mBART, full ft with IT	A ne pas manquer sur BBC Afrique : Une femme motivée et concentrée	Not to be missed on BBC Africa: A motivated and focused woman	111
8	mBART, freeze dec & emb, with IT	La professeure africaine de nanotechnologie a été lauréate du prix Kwame Nkrumah de l'Union africaine.	The African nanotechnology female professor was the recipient of the African Union Kwame Nkrumah Prize.	115
9	mBART, train data translation	Un scientifique africain a été lauréat du prix Kwame Nkrumah de l'Union africaine.	An African scientist has been awarded the African Union Kwame Nkrumah Prize.	108

Figure 4: Example predictions for a selection of models. Avg. len. over evaluation corpora in French, in characters. Red highlights errors or extra tokens.

runs in Figure 2, we observe that the zero-shot approach closely approaches the data translation approach in summarization and performs on par in question answering. The XQuAD dataset is harder to automatically translate than XL-Sum, e.g. single words often present in targets may be translated into short full sentences.

Experiments with larger models. Table 2 reports results for the mT5-large model where we compare performance achieved with full finetuning after intermediate tuning versus training on translated data. We also include the leader approach of using several source languages for XL-Sum. We consider only mT5 because mBART is released in one size. We reduce LR to 0.00001 for the larger model, as the LR of 0.0001 used for the base model was sometimes producing English outputs. We also list mT5-base results for reference.

We find that the same conclusions hold for the mT5-large model as for mT5-base: reducing LR eliminates generation in the wrong language, and the zero-shot approach is on par or better than the data translation approach.

5 Inspection of predictions

We inspected a subset of predictions in French and Russian and found that models achieving highest scores in both tasks generate fluent, meaningful and reasonable predictions in a lot of cases, but sometimes have issues with factualness, grammaticality or hallucinations. Examples are shown in Figure 4. Analyzing effects of LR, we observe that increasing LR leads first to increase in code switching and then to wrong language generation, while *reducing LR leads to producing rudiments of pretraining in generation*. For example, models sometimes generate extra tokens used in pretraining, such as <extra_id_{N}> for mT5 or <sep> for mBART, see rows 2 and 4 in Figure 4. *In most cases this does not affect meaningfulness of predictions*, but in *rare* cases leads to mT5 producing incomplete sentences, which may look unreasonable in summarization, e.g. "<extra_id_0> Guinea-Bissau President Alberto Dabo said." (translated from French). The reason is that in mT5 pretraining tokens <extra_id_{N}> were followed by fragments of input sentences. *The described effect is eliminated by intermediate tuning* (row 3 in Fig. 4).

In the same fashion, *mBART average lengths are closer to groundtruth average lengths than mT5 in summarization, and the reverse effect takes place in QA*. The reason is that in mT5 pretraining, the targets are only fragments masked in the input, which are shorter than targets in mBART pretraining represented by full sequences (they need to be reconstructed from the masked inputs).

Notably, data translation can produce translation-related errors, e.g. in rows 5 and 9 models generate a wrong male article "Un", probably because this was a dominating article in the translated data.

6 Conclusion

In this work, we conducted a deep systematic study of how to achieve high-performing zero-shot cross-lingual transfer in generation. Our study highlights the *high importance of careful learning rate tuning and the usefilness of the intermediate tuning*. We show that with these two ingredients, mT5 and mBART achieve strong results with simple full finetuning, i.e. closely approach the performance of translate-train in summarization and reach it in question answering. The performance gap in summarization is closed by using several source languages for mT5 and freezing decoder and embeddings for mBART. Translation-pretrained NLLB-200 shows surprisingly good performance in summarization but lags behind in question answering. We urge future works to pay more attention to hyperparameter tuning and to report more rigorously their experimental setup, as well as consider a wider spectrum of models and baselines in the experiments.

7 Limitations and broader impact

We aim at conducting a deep, thoughtful study of various design choices in zero-shot cross-lingual generation, but acknowledge the impossibility of considering all possible options, given the resource constraints. In particular, we could not perform full fine-grained grid search of LR for each task-model-adaptation method combination. Instead, we use a well-designed simplified strategy described in Section 3, which already gave strong results. In the same fashion, we had to limit our study to three models (we picked most commonly used models) and adaptation methods which do not require model pretraining, e.g. we do not consider mmT5 model. Nonetheless, we hope our study provides helpful insights on zero-shot cross-lingual transfer in generative tasks and shows that it can achieve the performance of the data translation method, which is usually considered as an unreachable upper baseline.

We do not anticipate any negative impact of our work and on the reverse hope that it will help to make higher-quality language technologies accessible to a broader set of languages.

8 Acknowledgments

We gratefully appreciate Alexandre Bérard's help in the project.

References

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020a. 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.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020b. 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.

Ankur Bapna and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. 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 1538–1548, Hong Kong, China. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. mbert blog post. https://github.com/google-research/bert/blob/master/multilingual.md.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. 2021. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4693–4703, Online. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online. Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tianjian Li and Kenton Murray. 2023. Why does zero-shot cross-lingual generation fail? an explanation and a solution. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12461–12476, Toronto, Canada. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Kaushal Kumar Maurya, Maunendra Sankar Desarkar, Yoshinobu Kano, and Kumari Deepshikha. 2021. Zm-BART: An unsupervised cross-lingual transfer framework for language generation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2804–2818, Online. Association for Computational Linguistics.

Jonas Pfeiffer, Francesco Piccinno, Massimo Nicosia, Xinyi Wang, Machel Reid, and Sebastian Ruder. 2023. mmt5: Modular multilingual pre-training solves source language hallucinations.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. 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.

Machel Reid and Mikel Artetxe. 2023. On the role of parallel data in cross-lingual transfer learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5999–6006, Toronto, Canada. Association for Computational Linguistics.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model.

Fabian David Schmidt, Ivan Vulić, and Goran Glavaš. 2022. Don't stop fine-tuning: On training regimes for few-shot cross-lingual transfer with multilingual language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10725–10742, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2021. Multilingual translation from denoising pre-training. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3450–3466, Online. Association for Computational Linguistics.

Tu Vu, Aditya Barua, Brian Lester, Daniel Cer, Mohit Iyyer, and Noah Constant. 2022. Overcoming catastrophic forgetting in zero-shot cross-lingual generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9279–9300, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. 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 833–844, Hong Kong, China. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Mengjie Zhao, Yi Zhu, Ehsan Shareghi, Ivan Vulić, Roi Reichart, Anna Korhonen, and Hinrich Schütze. 2021. A closer look at few-shot crosslingual transfer: The choice of shots matters. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5751–5767, Online. Association for Computational Linguistics.

A Experimental setup

Data. We experiment with news summarization on the XL-Sum dataset (Hasan et al., 2021) (released under the CC BY-NC-SA 4.0 license) and question answering on the XQuAD dataset (Artetxe et al., 2020b) (released under the CC BY-SA 4.0 license). Both datasets were released for research puposes. The XL-Sum dataset was obtained by crawling BBC news in 44 languages, with corpus size per language varying from 1K (Scottish Gaelic) to 300K (English) article-summary pairs. Inputs are composed of 1–2 paragraphs and targets are usually 2–3 sentences. We evaluate on test sets and crop test sets larger than 2K samples, to 2K. The XQuAD dataset was obtained by translating SQuAD validation set (Rajpurkar et al., 2016) into 11 languages, thus all language corpora are parallel. We use this dataset for evaluation and train on the training set of SQuAD (80K training instances). Each input is composed of a paragraph and a question about this paragraph appended in the end of the paragraph. Each output is an answer to a question, a short segment copied from the paragraph.

Preprocessing and postprocessing. We tokenize data using each model's tokenizer. We crop model inputs and outputs to the maximum lengths supported by models, which equal to 1024 tokens for mBART and 512 tokens for mT5-base and NLLB-600M. Due to the design of pretraining, models may generate extra tokens such as <extra_id_{N}> for or <sep> for mBART. We remove such extra tokens from predictions before computing metrics.

Models and training. We consider three models: mT5 (base and large, released under the Apache License 2.0 license), mBART (MIT license), and NLLB-200 (cc-by-nc-4.0 license). All models allow use for research purposes. We train models on English data for 20k steps with batch size of 4000 tokens on a single A100 GPU, and conduct validation on considered target languages each 2k steps. We use Adam optimizer with standard inverse square root LR schedule and warm up of 4k steps, and update model parameters after each minibatch. We estimated the total computational budget of our experiments to be 4K GPU hours.

Hyperparameter search. For full finetuning, adapters and prompt tuning, we run a search over a range of LR. For each task and model (without intermediate tuning), we search the LR best for non English languages on average, looking at ROUGE-2 for summarization and F-measure for QA. We start with the set of three LRs: $\{10^{-k}, k = 3, 4, 5\}$. If the optimal $k^* \neq 4$ then we extend search correspondingly to $k = 2, 1$ or $k = 6, 7$ until performance stops improving. For full finetuning, after we find optimal k^* we also consider $3 \cdot 10^{-k^*}$. The motivation is that the optimal k^* usually corresponds to the maximal k that still allows generation in the correct language, and considering $3 \cdot 10^{-k^*}$ enables more accurate search for this maximum. We report chosen LRs for full finetuning and adapters in Table 3. For prompt tuning we chose LR of 0.01 for both tasks.

Model	Method	XL-Sum LR	XL-Sum IT?	XQuAD LR	XQuAD IT?
mT5 (base)	Ft w/o IT	1e-4		1e-4	
	Ft	1e-4		1e-4	✓
	+ Mix tgt langs	1e-4		1e-4	✓
	+ >1 src langs	1e-4		n/a	
	Freeze	1e-4	✓	1e-4	✓
	Adapters	1e-3		1e-3	
	Prompt tuning	1e-2	✓	1e-2	✓
mBART	Ft w/o IT	1e-6		1e-5	
	Ft	1e-6	✓	1e-5	✓
	+ Mix tgt langs	1e-6	✓	1e-5	
	+ >1 src langs	1e-6	✓	n/a	
	Freeze	1e-5	✓	1e-5	✓
	Adapters	1e-5	✓	1e-3	✓
	Prompt tuning	1e-2	✓	1e-3	✓
NLLB	Ft w/o IT	1e-5		3e-5	
mBART (tr)	Ft w/o IT	1e-6		1e-3	

Table 3: Best hyperparameter configurations for non-English languages: chosen learning rates and whether intermediate tuning (IT) is used. n/a: not applicable.

Evaluation. For summarization, we report the ROUGE-2 metric (Lin, 2004), and for QA, we report F-measure. In QA, a lot of answers contain numbers or English words which could inflate metrics even if the model does not generate in the correct language. Moreover, the accuracy of language identification decreases on short answers, resulting in false indication of generation in wrong language. To avoid these issues, we compute metrics in QA only over questions for which groundtruth answers do not contain numbers and are correctly identified to be written in the target language (∼50% of 1190 questions satisfy this criteria).

For ROUGE metric, we use the `gem-metrics` package. For F1 metric in XQuAD, we use the script provided by the dataset authors. To identify

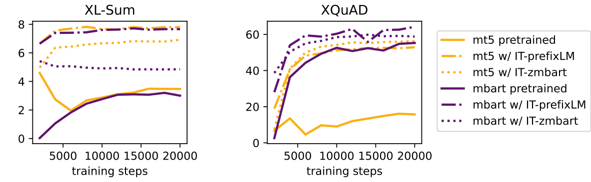

Figure 5: Comparison of self-supervised objectives for intermediate tuning, with freezing decoder and embeddings as an adaptation method. Task metric: Rouge-2 for XL-Sum, F1 for XQuAD. Correct language rate is close to 100% in all cases except pretrained mBART on XL-Sum.

language, we use `fasttext` library (Joulin et al., 2017, 2016) and its `lid.176.bin` model[6].

B Preliminary experiments with intermediate tuning

Figure 5 reports comparison of two self-supervised objectives for intermediate tuning: Prefix-LM and ZmBART-like objective. PrefixLM objective implies predicting the continuation of the chuck of text based on its beginning, while ZmBART-like objective implies citing random sentences from the input chunk of text. We compare two objectives using the freezing of the decoder and embeddings as an adaptation method, applied after intermediate tuning with the chosen objective, because we found intermediate tuning to be essential for this adaptation method in the preliminary experiments. Finetuning LR equals to the PLR defined in Section 4, intermediate tuning LR was chosen to optimize fluency of model generations, inspected manually. Intermediate tuning is performed on the CommonCrawl dataset.

We observe that for XL-Sum, the Prefix-LM objective leads to substantially higher Rouge-2 values, while for XQuAD both objectives lead to close results. Based on these results, we decided to use the Prefix-LM objective in all experiments.

C Preliminary experiments with mixing-in target languages

Figure 6 reports results of preliminary experiments with mixing-in a self-supervised task in target languages. For each base model, namely mT5-base and mBART, we consider its pretraining task and a Prefix-LM task used for intermediate tuning. We

[6] `https://fasttext.cc/docs/en/language-identification.html`

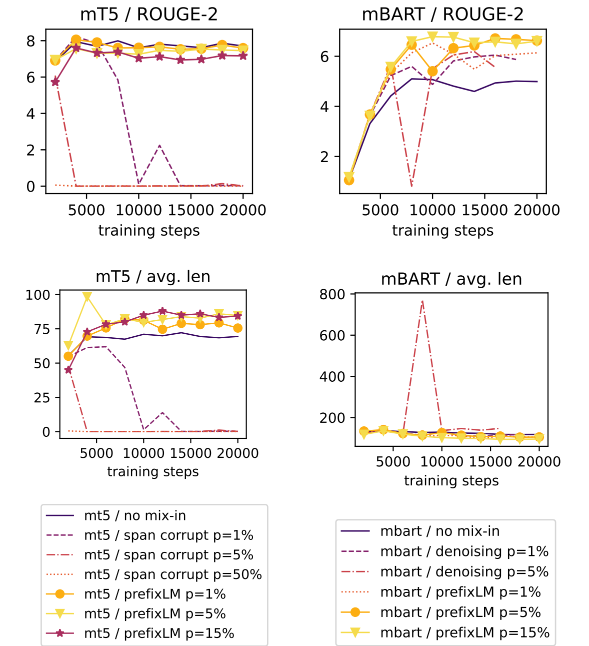

Figure 6: Preliminary experiments with mixing-in a self-supervised task for target languages. The probability in the legend denotes the probability of sampling target language examples when forming mini-batches. Two self-supervised tasks considered: Prefix-LM and the pretraining task of the model. Correct language rate is close to 100% in all cases

consider several options for the probability of sampling target language examples when forming mini-batches. CommonCrawl data is used for the self-supervised task. The experiment is conducted for the XL-Sum task, with LR being equal to the PLR defined in Section 4, without intermediate tuning.

For mt5, we observe that using the span corruption pretraining task leads to empty outputs with any mixing-in probability (with smaller probabilities this effect happens later in the training). This is because task examples do not contain any mask tokens, and empty generation is a default response of the pretrained mT5 to such inputs. Mixing-in PrefixLM task examples performs similarly to the standard finetuning of mT5, with mixing-in probability of 1% performing best, same as in (Vu et al., 2022). Qualitatively, mixing-in self-supervised task increases the length of generated outputs in the tasks of interest.

For mBART, all mixing-in strategies lead to modest improvements in performance, with PrefixLM task performing slightly better. All considered mixing-in probabilities lead to similar results. Based on these observations, we decided to use the

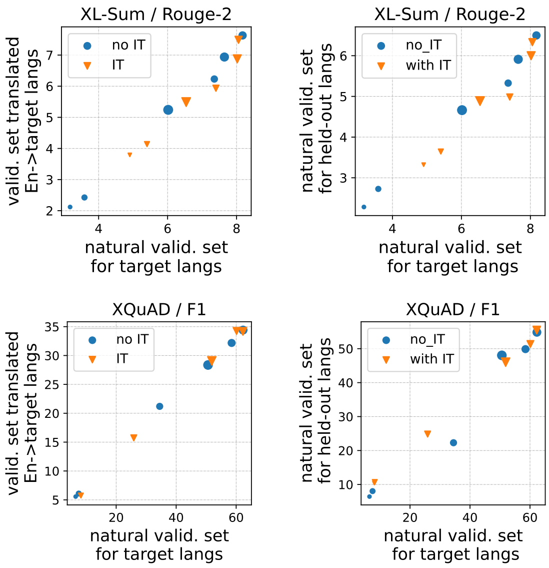

Figure 7: Correlation in performance on various validation sets. Each dot represents mT5-base finetuned on English data, with or without intermediate tuning (reflected with the shape and color of the point), and the size of the point reflects the learning rate. Natural valid. set means the validation set provided by the authors of the dataset.

PrefixLM task with mixing-in probability of 1% in our experiments.

D Comparing validation sets

Figure 7 demonstrates the correlation between performance measured on various validation sets. Performance on validation sets in target languages correlates with performance on validation sets in held-out languages and validation sets translated from English into target languages. This shows that having validation sets in target languages is not necessary in practice which is important to enable fully zero-shot setting.

E Additional experiment with translating English outputs into target languages

When reducing the LR for preserving generation in correct language, a reasonable question could be whether predictions of higher LR models are higher quality answers, but just in the wrong language, or simply hallucinations caused by data distribution shift. The premise for the former scenario is that on English data, performance with our chosen LR is usually slightly lower than with a larger LR.

We find that actually the later scenario takes place, by comparing performance of our chosen LR (best for non-English) and of the best LR for English with model predictions being translated into target languages using NLLB-3.3B[7], for last checkpoints of full models finetuning. According to Table 4, translated predictions of the higher LR model score lower than the (non-translated) predictions of the lower LR model. This result further advocates for the importance of careful LR tuning for full finetuning in zero-shot cross-lingual transfer in generation.

		Best-En LR + Tr.		**Best-non-En LR**	
		LR	**Score**	**LR**	**Score**
Sum	mT5	1e-3	4.02	1e-4	7.7
	mBART	1e-5	4.06	1e-6	5.34
	NLLB-200	1e-4	2.86	1e-5	4.62
QA	mT5	1e-4	46.2	1e-4	58.6
	mBART	1e-5	41.1	1e-5	46.6
	NLLB-200	1e-4	17.4	3e-5	18.2

Table 4: Comparison of best LR for non-English languages and best LR for English with model outputs being translated into target languages. Performance averaged over non-English languages, after 20k of full finetuning. Reported metric: Rouge-2 for summarization, F-measure for QA. mBART — pretrained version, no intermediate tuning is used in this experiment.

[7]NLLB-3.3B handles well inputs containing code switching which are frequent in predictions we are translating, and simply copies inputs which are already in the target language.

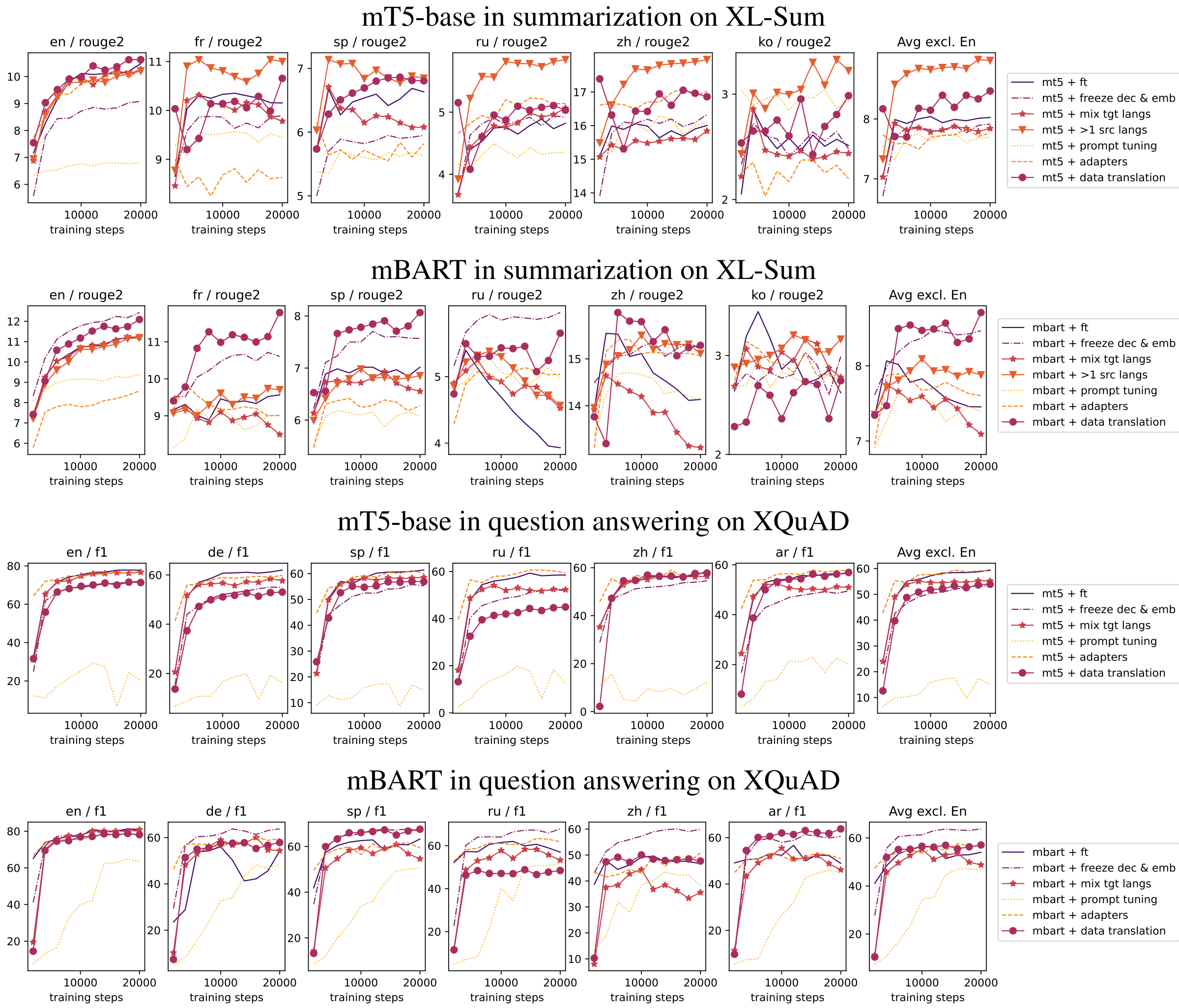

Figure 8: Per-language results on the comparison of adaptation methods. Each plot averaged over 2 runs. Correct language rate is close to 100% in all cases, due to the hyperparameter tuning, except prompt tuning of mT5 in the XQuAD task.

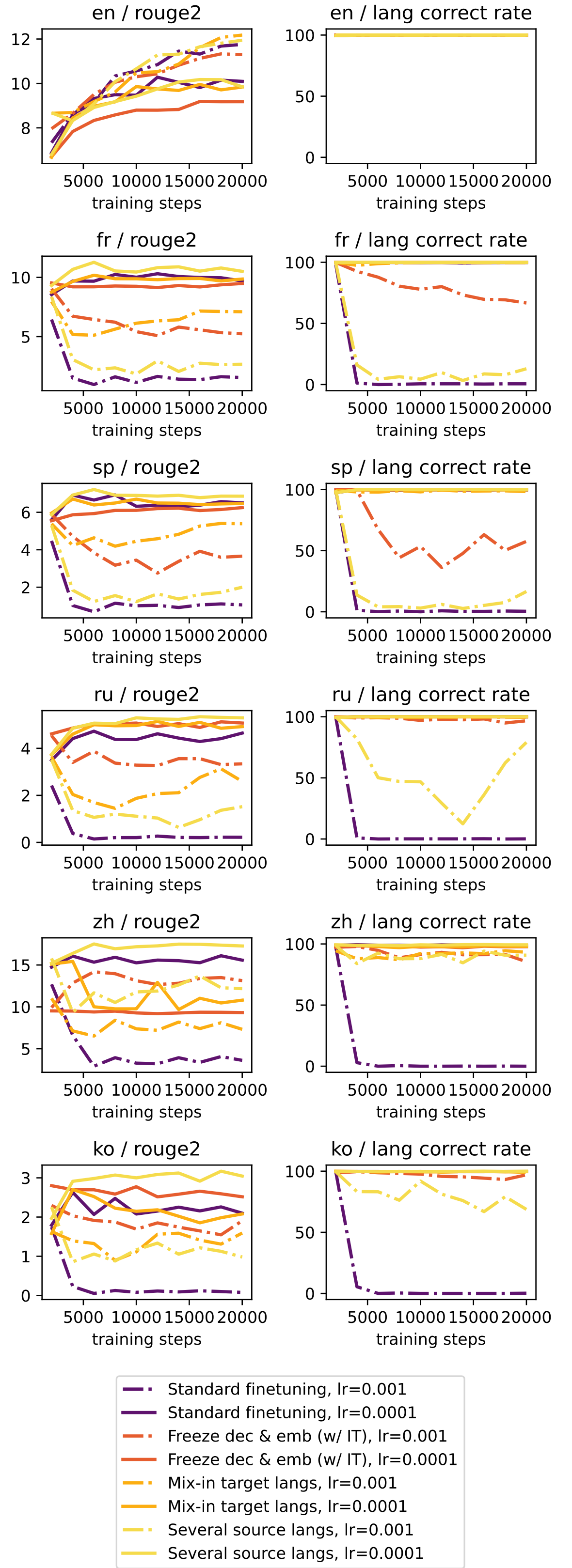

Figure 9: Per-language results on the effect of learning rate, for mT5 on XL-Sum.

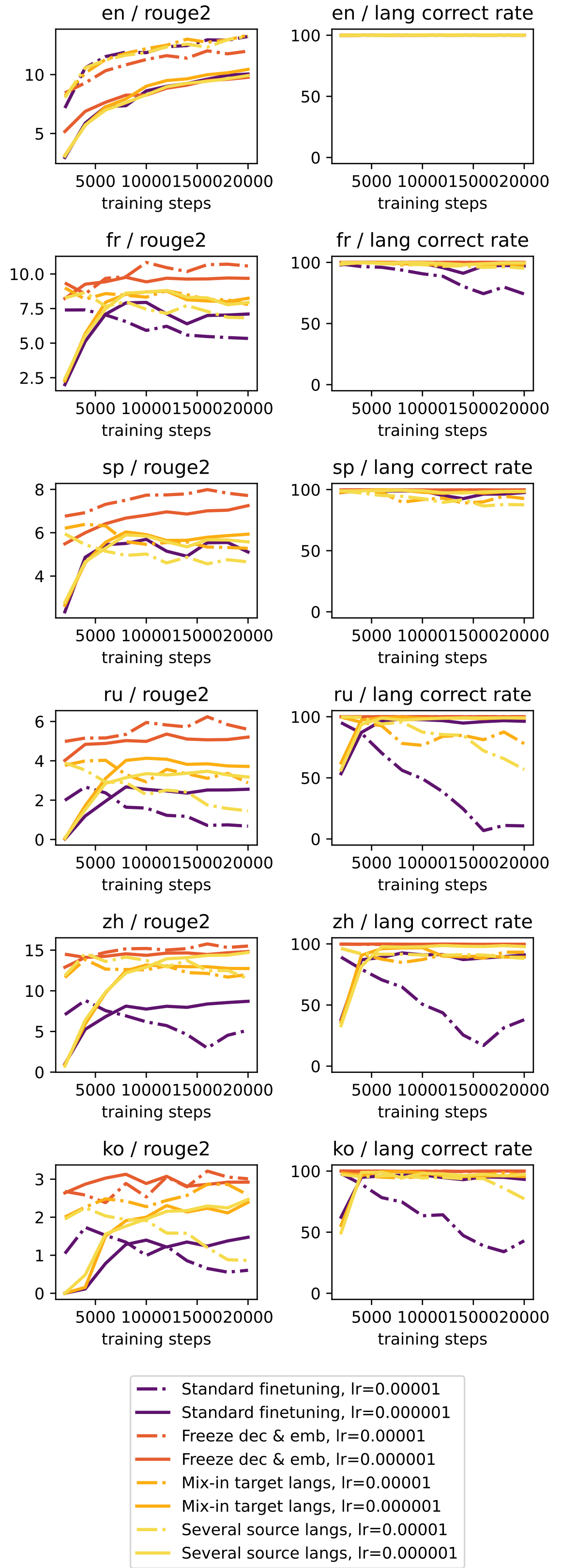

Figure 10: Per-language results on the effect of learning rate, for mBART on XL-Sum.

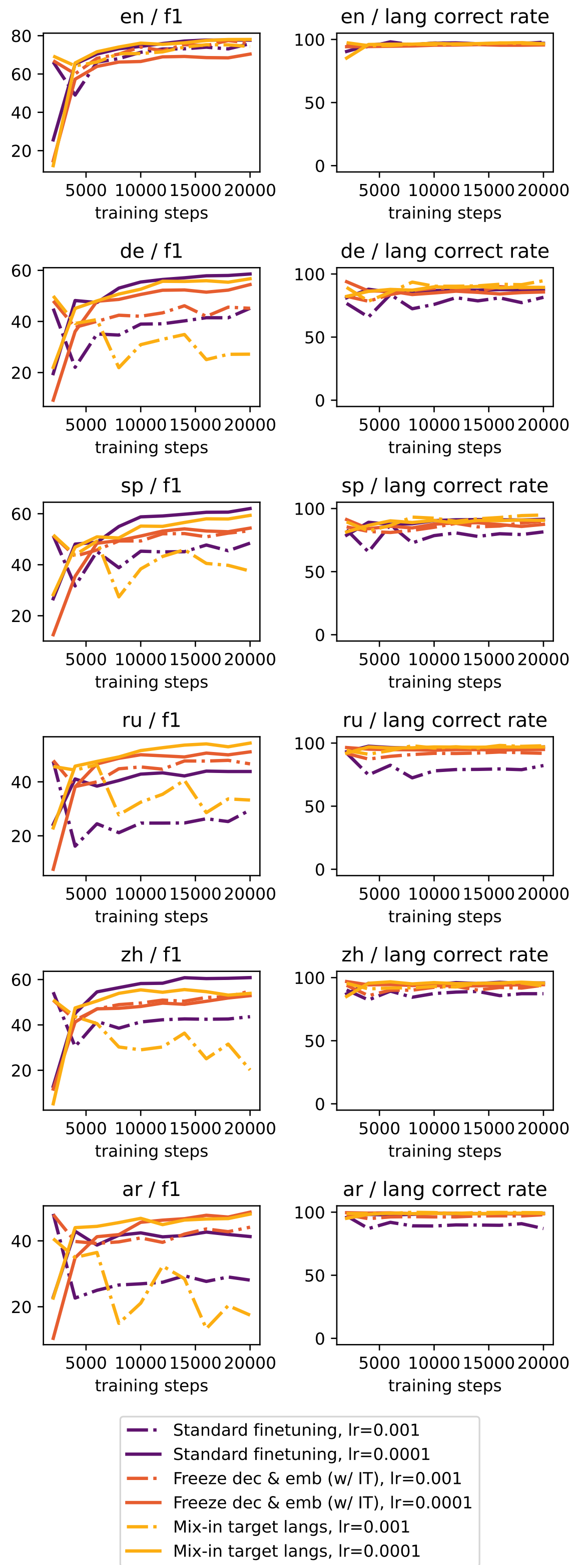

Figure 11: Per-language results on the effect of learning rate, for mT5 on XQuAD.

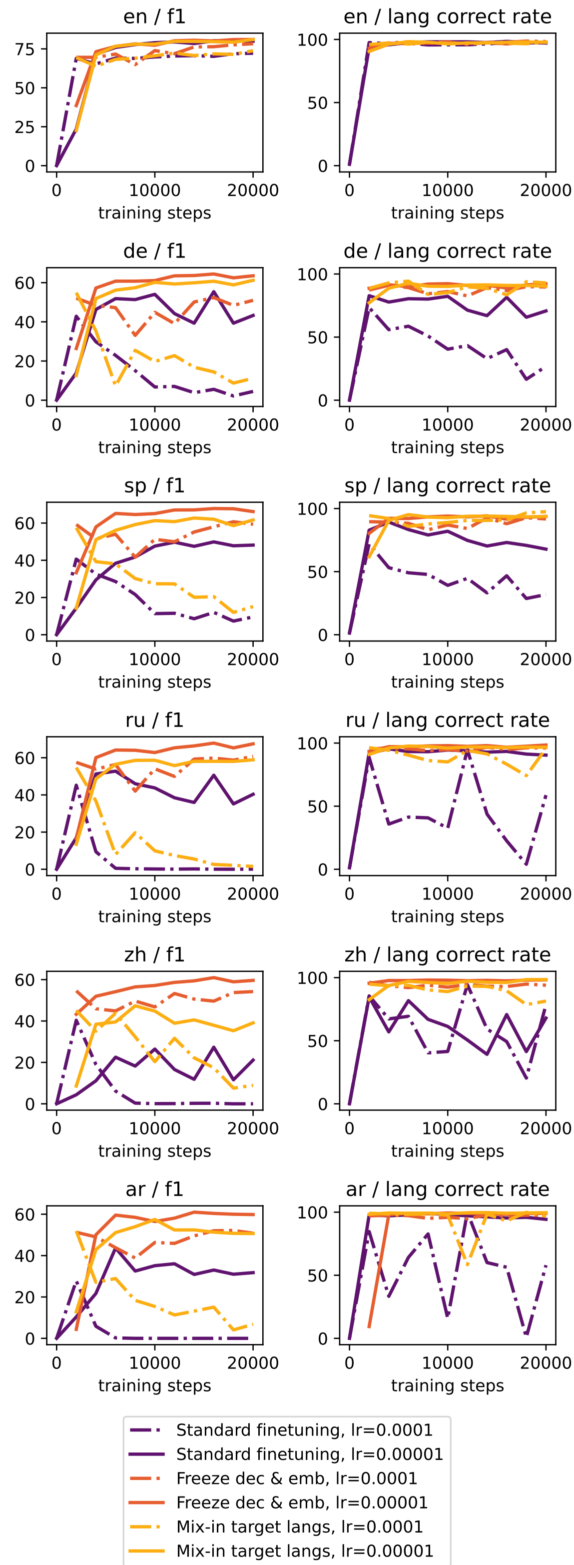

Figure 12: Per-language results on the effect of learning rate, for mBART on XQuAD.