Adapting BigScience Multilingual Model to Unseen Languages

Anonymous ACL submission

Abstract

We benchmark different strategies of adding new languages (German and Korean) into the BigScience's pretrained multilingual language model with 1.3 billion parameters that currently supports 13 languages. We investigate the factors that affect the language adaptability of the model and the trade-offs between computational costs and expected performance.

1 Introduction

Pretrained multilingual language models (LMs) have enabled cross-lingual transfer (Artetxe et al., 2020; Conneau et al., 2020b; K et al., 2021). While several works suggest that such knowledge transfer goes beyond just vocabulary sharing across languages (Artetxe et al., 2020; K et al., 2021), others have shown that the models' performance is sensitive to the quality of the tokenization it relies on (Pfeiffer et al., 2021b). Moreover, the quality of knowledge transfer can degrade for unseen languages, especially when the scripts are unknown to the model (Muller et al., 2021).

Data availability, but also the *curse of multilinguality* (Conneau et al., 2020a) makes training a single model covering all the languages challenging: there is a trade off between the number of languages in pretraining data, model capacity and the downstream performance for each individual language. Finally, it is hard to anticipate any potential usage of pretrained LM in advance, hence it is important to study *a posteoriori* adaptation to new tasks/languages. Recently proposed methods include **continual pretraining** of the model (restricted to the embedding layer training only in some cases) (Artetxe et al., 2020; Chau et al., 2020; Muller et al., 2021; Zhang et al., 2020; Wang et al., 2020), or training of **language-specific adapters** (Pfeiffer et al., 2020, 2021a,b; Philip et al., 2020; Üstün et al., 2021; Berard, 2021) for the target language. The core motivation behind these methods is to benefit from knowledge transfer encoded in the pretrained LM for the new language processing at a small computational cost (compared to full model retraining) .

In this work, we aim at better understanding the trade-offs between the amount of compute and the final downstream task performance. Specifically, we study the impact of the following three factors: original pretraining steps, adaptation strategies, and adapters' capacity on the Natural Language Inference (NLI) task. As part of the initiative of BigScience, we experiment with its multilingual LM, which currently only supports 13 languages. We investigate how researchers could adapt BigScience's full-open-access multilingual models to their languages of interest. Moreover, BigScience open source its intermediate pretraining checkpoints, therefore we also study how does the adaptability to new language changes with the amount of pretraining.

Our main findings are: (1) the most promising strategy is training of the embedding layers and adapters simultaneously; (2) position encoding is an important component to adapt; (3) pretraining steps benefit zero-shot performance; (4) adaptation to low-resource languages can be done with low parameter budget.

2 Background and Related Work

Most of the works investigating extension of pretrained models to new languages consider models such as (Devlin et al., 2019) and XLM-R (Conneau et al., 2020a) that were pretrained to 100+ languages with Masked Language Modeling (MLM) objective. Artetxe et al. (2020); Pfeiffer et al. (2021b) demonstrate that it is possible to add new language to these models by training new embedding layer. Muller et al. (2021) continues training the pretrained mBERT on the new language data, and finds that transliteration of languages using non-latin script boosts performance on these

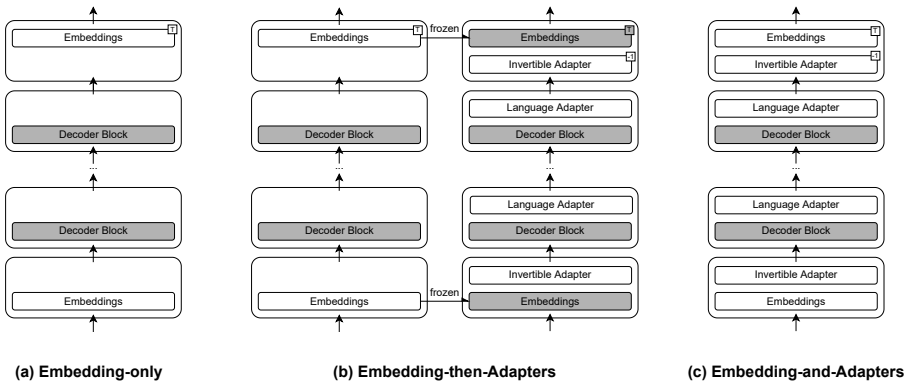

Figure 1: Three different language adaptation strategies: (a) Embedding-only strategy: finetune the embedding layer of the pretrained model, which is tied with the language modeling head. (b) Embedding-then-Adapters strategy: first finetune the embedding layer, then train the invertible adapter and language adapters. (c) Embedding-and-Adapters strategy: finetune the embedding layer, invertible and language adapters at the same time. Grey blocks indicate frozen layers whereas white blocks indicate trainable layers.

languages.

Berard (2021) adds new languages into pretrained multilingual NMT model by training embedding layer and adapter layers. It shows that higher adaptation cost is required for new target languages (as opposed to new source languages).

Closest to our work (Ebrahimi and Kann, 2021) studies different approaches (i.e., continued pretraining, vocabulary expansion and adapter layers) allowing to extend XLM-R model to 1600 languages. They conclude that continued pretraining is the most promising direction. However, the cost of such pretraining will grow with the size of the pretrained model and can be quite prohibitive for many researchers working with low-resource languages.

Our work aims to compare lightweight strategies of adaptation to new language (embedding layer training and adapter layers) across the following previous unstudied dimensions: we consider the (1) adaptation parameter budget and (2) intermediate pretraining checkpoints by analyzing how they impact autoregressive model's adaptability to new languages.

3 Experimental settings

3.1 BigScience Pretrained Multilingual Language Model

BigScience open-sources one of its pretrained multilingual autoregressive LMs[1] (Anonymous, 2022) that relies on transformer architecture with 24 decoder layers, 2048 hidden units per layer, 16 attention heads and 1.3B parameters. This model is trained with Next Work Prediction objective (*aka* GPT-style). It is pretrained on OSCAR deduplicated subcorpora (Ortiz Suárez et al., 2019) for the following thirteen languages for a total of training 400B tokens: Indonesian, Basque, Vietnamese, Chinese, Urdu, Spanish, Catalan, Portuguese, French, English, Hindi, Arabic, and Bengali. See Appendix A for further model pretraining details.

In our experiments, we use the model checkpoint saved after 12,000, 100,500, and 118,500 pretraining steps (final checkpoint).

3.2 New Languages and Data

The need for adaptation of pretrained multilingual LM to new languages typically raises for low-resource languages for three reasons: first, the lack of the available data for such languages; second, the lack of the computational resources with which people working on low-resource languages may experience; lastly, the lack of the evaluation resources, which makes it hard to assess the quality of the adapted model properly. In this study we simulate low-resource settings by restricting the amount of training samples for high-resource languages. This allows us to benefit from good-quality evaluation datasets available for the high resource languages. We expect that our observations could generalize to real low-resource scenarios. We've chosen two languages for the adaptation of BigScience model: (1) *German* (easy language) that belongs in the same Germanic language family as English and shares the Latin script with several pretraining languages; (2) *Korean* (difficult language)

[1] https://huggingface.co/bigscience/tr5b-1B3-multilingual-alpha-checkpoints/tree/main

Language	Prompt Template	Candidate Verbalizer for `[MASK]`		
		Entailment	Contradiction	Neutral
de	`[premise]`, richtig? `[Mask]`, `[hypothesis]`	Ja	Nein	Auch
ko	`[premise]`, 맞지? `[Mask]`, `[hypothesis]`	예	아니요	또한
en	`[premise]`, right? `[Mask]`, `[hypothesis]`	Yes	No	Also

Table 1: Natural language inference (NLI) prompts in German (de), Korean (ko) and English (en). The German and Korean prompts are manually translated from the English counterpart.

	Models	Strategies	Ckpt.	Emb.	Adpt. Red.	(p.) de	en→ de	de→ de	(p.) ko	en→ ko	ko→ ko
(1)	mBERT_{BASE}	-	-	-	-	-	70.0	75.5	-	69.7	72.9
(2)	XLMR_{LARGE}	-	-	-	-	-	82.5	85.4	-	80.4	86.4
(3)	$\text{XGLM}_{1.7B}$	-	-	-	-	45.4	-	-	45.17	-	-
(4)	BigScience	-	-	-	-	34.1	44.8	67.4	-	-	-
(5)	BigScience	Emb	118,500	wte,wpe	-	41.4	50.7	74.3	34.4	45.6	53.4
(6)	BigScience	Emb→Adpt	118,500	wte,wpe	16	40.0	50.5	69.9	33.8	40.4	51.8
(7)	BigScience	Emb+Adpt	118,500	wte	16	42.4	58.4	73.3	38.8	49.7	55.7
(8)	BigScience	Emb+Adpt	118,500	wte	48	42.4	57.6	73.7	36.3	48.3	52.9
(9)	BigScience	Emb+Adpt	118,500	wte	384	42.4	55.3	74.2	37.5	49.4	54.6
(10)	BigScience	Emb+Adpt	100,500	wte	16	44.3	56.9	73.2	37.5	48.6	50.8
(11)	BigScience	Emb+Adpt	12,000	wte	16	33.5	55.2	70.5	32.9	46.4	53.3
(12)	BigScience	Emb+Adpt	118,500	wte,wpe	16	44.7	64.9	73.0	-	-	-

Table 2: Evaluation of language adaptation for German (de) and Korean (ko) on NLI with three baselines (mBERT_{BASE}, XLMR_{BASE}, $\text{XGLM}_{1.3B}$). "Strategies" column indicates language adaptation strategies, which cover Embedding-only (Emb), Embedding-then-Adapters (Emb→Adpt) and Embedding-and-Adapters (Emb+Adpt). "Ckpt." column stands for the BigScience pretrained checkpoint, "Emb." column the types of embedding layers (wte: token embedding, wpe: positional embedding), and "Adpt. Red." column the reduction factor for language adapters. "(p.) de/ko" column reports the prompt-based zero-shot evaluation result, "en→de/ko" cross-lingual result, and "de/ko→de/ko" supervised finetuning result.

that belongs to Koreanic language family and uses Hangul script system, that none of the languages of pretrained LM cover.

We train the pretrained model's byte-level BPE tokenizer on German and Korean deduplicated OSCAR subcorpora (Ortiz Suárez et al., 2019) and use 100,000 samples from OSCAR subcorpora of the respective languages for adaptation.

3.3 Language Adaptation Strategies

We experiment with the following three language adaptation strategies that involve finetuning of the embedding layer and adapters layers (see Figure 1). See Appendix B for further training details.

Embedding-only. The embedding layer consists of token embeddings and positional embeddings. We follow Artetxe et al. (2020) by learning new token embeddings and positional embeddings while freezing the rest of the transformer parameters.

Embedding-then-Adapters. Following the Embedding-only approach, we first train the embedding layer on the target language pretraining data for 25,000 steps. Then, we freeze the entire transformer and add MAD-X invertible and language adapters (Pfeiffer et al., 2020). We subsequently finetune the adapters for an additional 25,000 steps.

Embedding-and-Adapters. Instead of training the embedding layer and the adapters separately, we train both of them at the same time for 50,000 steps. We also vary the adapter capacity, which is defined by the reduction factor (also known as compression rate (Rücklé et al., 2021)) in the adapter's bottleneck layer. A smaller reduction value would lead to a larger amount of adapter parameters and, accordingly, a larger adaptation capacity. We consider the reduction factors of 16 (default), 48, and 384.

3.4 Evaluation

We evaluate our models on the natural language inference (NLI) task, using XNLI dataset (Conneau et al., 2018) for German and KLUE-NLI dataset (Park et al., 2021) for Korean.[2] We adopt the three following settings with increasing levels of task supervision.

[2]We evaluate our models on the dev set of KLUE-NLI.

Prompt-based Zero-shot Following Lin et al. (2021), we use the German and Korean cloze-style prompt templates (see Table 1). The zero-shot prediction is the candidate label verbalizer, which replaces the `[MASK]` token, that maximizes the LM based likelihood of the prompt sentence.

Cross-lingual In this evaluation setting, we follow Pfeiffer et al. (2020) and train task adapters on English task dataset. These adapters are then used on the target language (German or Korean) evaluation set. We adopt adapter size (reduction factor = 16) and run training for 2 epochs with the batch size of 32, learning rate of $5e-5$, and maximum sequence length of 128 input tokens on a single V100 GPU, which takes around 5 hours to complete.

Supervised Finetuning Our finetuning strategies are identical to the cross-lingual evaluation setting above, except that the classification head and the task adapters are finetuned on the target language dataset. Note that this implies that we train different task adapter for each model.

4 Results and Discussion

Table 2 reports the results on different adaptation strategies.

Baselines. XLM-R and mBERT are pretrained multilingual LMs that were trained on 100+ languages and include both German and Korean languages. Therefore the adaptation of these models can be seen as an upper bound performance one could achieve on classification tasks. XGLM (Lin et al., 2021) is a multilingual autoregressive LM that was pretrained on 30 languages, including both German and Korean. It serves as the (upper-bound) baseline for the prompt-based evaluation setting due to its zero-shot learning capability through prompts on NLI (Lin et al., 2021).

Language Adaptation Strategies. First, original BigScience multilingual model **(4)** performs poorly on German.[3] Model **(6)** that trains embedding layers and adapters separately seems to perform worse than models training them simultaneously (models **(7)** and **(12)**). Training adapter layers **(7)** boosts performance compared to the model updating embedding layers only **(5)**, especially under the cross-lingual setting for both German and Korean languages.

We also observe that it is easier to adapt to German, where adapted models **(5)**–**(10)** almost close the gap with the upper bound baselines (**(1)**–**(3)**), but this is not the case for Korean, where this gap is still important. This is in line with Muller et al.'s (2021) findings suggesting that knowledge transfer is more efficient between related languages.

Original Pretraining Steps. The models **(7)**, **(10)** and **(11)** compare the effect of the pretraining steps on the new language adaptability. We observe that, after language adaptation, the early pretraining checkpoint (row **(11)** in Table 2) only performs as good as a non-adapted model **(4)** when evaluated with prompts. However, after further finetuning on the downstream task, its performance catches up with later checkpoints. This preliminary result suggests that researchers who are interested in adapting a multilingual LM to their language of interest may not need a large number of pretraining steps.

Adapters' Capacity. The models with different adapters' capacity (models **(7)**, **(8)** and **(9)**) achieve comparable downstream performance when adapting to German and Korean languages. The small impact of parameter budget on the final performance could be due to the limited amount of training data. This implies that in the real low-resource settings there is no need for a very large parameter budget.

Positional Embeddings. Our results (model **(7)** vs model **(12)** in Table 2) highlight the importance of adapting the positional embedding layer of the LM, which gives 6.5% accuracy boost for cross-lingual NLI.[4] This is in line with the findings by Ravishankar and Søgaard (2021).

5 Conclusion and Future Work

In this work we've compared different adaptation strategies to add a new language into pretrained model and identified important components for adaptation.

Limitations. Our work only studies adaptation of multilingual LMs to two unseen languages on a single task. In the future, we want to include more languages for adaptation and evaluate the models on more NLU and NLG tasks. We also plan to explore the impact of the size of language adaptation training samples on model's downstream performance.

[3] We did not perform evaluation on Korean as the vocabulary used by BigScience model doesn't cover Hangul at all.

[4] The training of model **(12)** diverged on Korean dataset, therefore we can only make conclusions based on German dataset. We need to investigate this issue further

References

Anonymous. 2022. What language model to train if you have two million gpu hours? *In submission*.

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.

Alexandre Berard. 2021. Continual learning in multilingual NMT via language-specific embeddings. In *Proceedings of the Sixth Conference on Machine Translation*, pages 542–565, Online. Association for Computational Linguistics.

Ethan C. Chau, Lucy H. Lin, and Noah A. Smith. 2020. Parsing with multilingual BERT, a small corpus, and a small treebank. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1324–1334, Online. 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. 2020a. 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.

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.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Abteen Ebrahimi and Katharina Kann. 2021. How to adapt your pretrained multilingual model to 1600 languages. 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 4555–4567, Online. Association for Computational Linguistics.

Karthikeyan K, Aalok Sathe, Somak Aditya, and Monojit Choudhury. 2021. Analyzing the effects of reasoning types on cross-lingual transfer performance. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 86–95, Punta Cana, Dominican Republic. 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. 2021. Few-shot learning with multilingual language models.

Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462, Online. Association for Computational Linguistics.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. In *Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019*, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

Sungjoon Park, Jihyung Moon, Sungdong Kim, Won Ik Cho, Jiyoon Han, Jangwon Park, Chisung Song, Junseong Kim, Yongsook Song, Taehwan Oh, Joohong Lee, Juhyun Oh, Sungwon Lyu, Younghoon Jeong, Inkwon Lee, Sangwoo Seo, Dongjun Lee, Hyunwoo Kim, Myeonghwa Lee, Seongbo Jang, Seungwon Do, Sunkyoung Kim, Kyungtae Lim, Jongwon Lee, Kyumin Park, Jamin Shin, Seonghyun Kim, Lucy Park, Alice Oh, Jungwoo Ha, and Kyunghyun Cho. 2021. Klue: Korean language understanding evaluation.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021a. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503, Online. Association for Computational Linguistics.

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.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2021b. UNKs everywhere: Adapting multilingual language models to new scripts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10186–10203, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jerin Philip, Alexandre Berard, Matthias Gallé, and Laurent Besacier. 2020. Monolingual adapters for zero-shot neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4465–4470, Online. Association for Computational Linguistics.

Vinit Ravishankar and Anders Søgaard. 2021. The impact of positional encodings on multilingual compression. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 763–777, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Andreas Rücklé, Gregor Geigle, Max Glockner, Tilman Beck, Jonas Pfeiffer, Nils Reimers, and Iryna Gurevych. 2021. AdapterDrop: On the efficiency of adapters in transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7930–7946, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ahmet Üstün, Alexandre Berard, Laurent Besacier, and Matthias Gallé. 2021. Multilingual unsupervised neural machine translation with denoising adapters. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6650–6662, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020. Extending multilingual BERT to low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2649–2656, Online. Association for Computational Linguistics.

Rong Zhang, Revanth Gangi Reddy, Md Arafat Sultan, Vittorio Castelli, Anthony Ferritto, Radu Florian, Efsun Sarioglu Kayi, Salim Roukos, Avi Sil, and Todd Ward. 2020. Multi-stage pre-training for low-resource domain adaptation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5461–5468, Online. Association for Computational Linguistics.

A BigScience Pretraining

The BigScience multilingual autoregressive language model (LM) uses the GPT architecture and the causal language modeling objective. Its tokenizer is the byte-level BPE with a vocab size of 130,001. For pretraining, it uses the Adam optimizer with a batch size 512, a learning rate of 2e-4 with cosine decay over 16,927,083 samples and warmup over 216,320 samples, weight decay of 0.1, and gradient clipping of 1.0.

The deduplicate subcorpora from OSCAR (Ortiz Suárez et al., 2019) for pretraining the model are sampled using the following probabilities: Indonesian (0.0554), Basque (0.0184), Vietnamese (0.0684), Chinese (0.1339), Urdu (0.0267), Spanish (0.1118), Catalan (0.0395), Portuguese (0.0867), French (0.1110), English (0.2107), Hindi (0.0398), Arabic (0.0638) and Bengali (0.0339).

B Language Adaptation Training Details

For all the language adaptation strategies, we train by using the AdamW optimizer with default parameters, batch size of 8, learning rate of $1e-3$ with linear decay, and maximum sequence length of 1,024. It takes around 30 hours to complete the training on a single V100 GPU machine.

C Relationship between NLI Task Performance and Adapter Capacity

We convert the reduction factor of language adapters to total adapter capacity and plot the NLI performance against it in Figure 2. The figure shows comparable performance across all NLI evaluation settings for both German and Korean languages even with small adapter capacity.

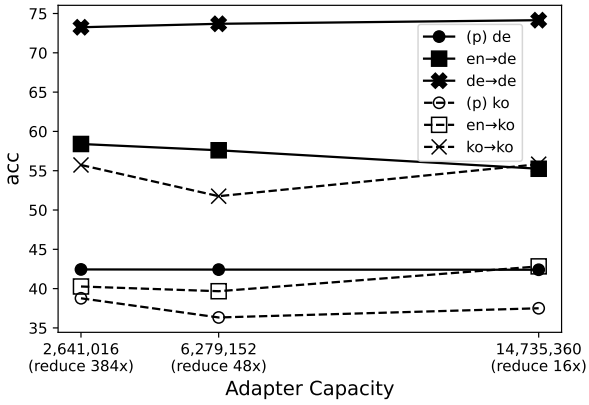

Figure 2: Graph of test accuracy for NLI against total adapter capacity (after applying reduction factor) using the Embedding-and-Adapters adaptation.