Soft Language Prompts for Language Transfer

Ivan Vykopal[1,2], **Simon Ostermann**[3,4] and **Marián Šimko**[2]
[1] Faculty of Information Technology, Brno University of Technology, Brno, Czech Republic
[2] Kempelen Institute of Intelligent Technologies, Bratislava, Slovakia
`{name.surname}@kinit.sk`
[3] German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany
[4] Centre for European Research in Trusted AI (CERTAIN), Saarbrücken, Germany
`simon.ostermann@dfki.de`

Abstract

Cross-lingual knowledge transfer, especially between high- and low-resource languages, remains challenging in natural language processing (NLP). This study offers insights for improving cross-lingual NLP applications through the combination of parameter-efficient fine-tuning methods. We systematically explore strategies for enhancing cross-lingual transfer through the incorporation of language-specific and task-specific adapters and soft prompts. We present a detailed investigation of various combinations of these methods, exploring their efficiency across 16 languages, focusing on 10 mid- and low-resource languages. We further present to our knowledge the first use of soft prompts for language transfer, a technique we call **soft language prompts**. Our findings demonstrate that in contrast to claims of previous work, a combination of language and task adapters does not always work best; instead, combining a soft language prompt with a task adapter outperforms most configurations in many cases.

1 Introduction

Many multilingual large language models (LLMs) have been developed in recent years, demonstrating promising performance on various NLP tasks across multiple languages (Xue et al., 2021; Workshop et al., 2023). These models are pre-trained on extensive corpora of unlabelled data in numerous languages, allowing an adaptation to linguistic characteristics and nuances. In addition, LLMs are often further trained on downstream tasks in a selected subset of languages (Muennighoff et al., 2023). However, only few LLMs focus on low-resource languages (Tang et al., 2020; Xue et al., 2021; Üstün et al., 2024; Gurgurov et al., 2024).

As the number of covered languages in the model increases, the issue of the *curse of multilinguality* arises. This problem occurs when the LLM's capacity is limited, causing languages with less training data to perform poorly (Conneau et al., 2020). Various approaches have been employed to address this limitation, primarily involving additional trainable parameters specific to individual languages (Pfeiffer et al., 2020, 2023).

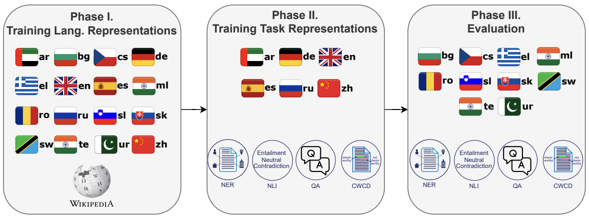

Figure 1: The full pipeline consists of (1) training language representation on 16 languages on the unlabelled data from Wikipedia; (2) training task representations on six high-resource languages on four NLP tasks; and (3) evaluating of cross-lingual transfer capabilities on 10 mid- and low-resource languages.

An alternative to language-specific tuning is *cross-lingual transfer*, where researchers investigate the knowledge transfer between high and low-resource languages. In cross-lingual transfer methods, an LLM is trained on a downstream task in one language, most often high-resource, and evaluated in other languages (Pikuliak et al., 2021). However, training only task-specific representations does not always capture the nuances of languages on which the LLM has not been trained or has been trained only to a small extent. Therefore, incorporating language-specific features can enhance knowledge transfer across languages.

Previous work has primarily investigated language and task representations by training language and task-specific adapters (Pfeiffer et al., 2020; Parović et al., 2022) or by employing language arithmetics (Klimaszewski et al., 2024). Nonetheless, other approaches that involve incorporating additional parameters to the model for language representation have not been thoroughly explored. This opens the opportunity to explore a combina-

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 10294–10313
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

tion of language and task representations using other methods and their impact on cross-lingual knowledge transfer.

To explore the utilization of language and task representations, we evaluate various configurations by combining two parameter-efficient fine-tuning (PEFT) methods that incorporate additional parameters into the LLM, namely *adapters* and *prompt-tuning*. Leveraging these additional language- and task-specific parameters increases the capacity of an mT0-BASE model and improves performance for cross-lingual transfer. We evaluate the performance of each configuration by training on six high-resource languages and evaluating its effectiveness on 10 mid- and low-resource languages on four selected tasks[1].

Our main contributions are:

- We propose **soft language prompts** as an alternative method for cross-lingual transfer.
- We comprehensively evaluate combinations of adapters and soft prompts for cross-lingual transfer and find that language prompts provide a viable alternative to language adapters, especially for low-resource languages.
- In addition, we provide an exhaustive evaluation of both prompts and adapters for task transfer. We find that the best combination of adapters and prompts for task and language transfer depends highly on task and language, resp., and that no solution clearly outperforms the others.

2 Related Work

Adapters and Soft Prompts. PEFT methods are designed to address the problem of the increasing number of trainable parameters in LLMs (He et al., 2022; Dettmers et al., 2023; Zhang et al., 2023; Xu et al., 2023; Xie et al., 2024). These methods reduce the number of trained parameters and incorporate new parameters commonly used to train LLMs on other tasks. Adapters (Houlsby et al., 2019a) and Prompt-Tuning (Lester et al., 2021) represent two PEFT methods for adapting LLMs to different NLP domains. Adapters incorporate new parameters into the transformer architecture by including down- and up-projection layers along with residual connection, while prompt-tuning introduced trainable soft-prompts prepended to input embeddings to condition the LLM's generation.

Limitations of Multilingual LLMs. One major limitation of LLMs is *catastrophic forgetting*, which occurs when training the LLM on a new task, causing it to partially or entirely forget previously learned knowledge for other tasks (McCloskey and Cohen, 1989; Luo et al., 2024; Ren et al., 2024). This forgetting extends beyond task-specific knowledge to language-specific knowledge if the model is fine-tuned on a subset of the original languages (Vu et al., 2022a; Liu and Huang, 2023).

Another challenge with multilingual LLMs is associated with the number of languages on which these LLMs have been pre-trained (Conneau et al., 2020; Pfeiffer et al., 2022). Previous research has shown that as the number of languages covered by LLMs increases, their performance on various NLP tasks degrades (Hu et al., 2020; Ponti et al., 2020). Additionally, low-resource languages are often underrepresented during pre-training, resulting in poor performance in these languages (Wu and Dredze, 2020).

Cross-Lingual Transfer. Given the many low-resource and underrepresented languages, cross-lingual transfer is crucial for training LLMs to address NLP tasks in various languages (Pikuliak et al., 2021). A common approach involves training LLMs in one language and evaluating them in another. Recent methods use additional parameters to create language-specific representations, assisting LLMs in solving NLP tasks in low-resource languages (Üstün et al., 2020; Ansell et al., 2022). These include training task adapters on top of language adapters (Pfeiffer et al., 2020; Ansell et al., 2021; Pfeiffer et al., 2023; Kunz and Holmström, 2024), training language adapters on source and target languages (Parović et al., 2022), and fusing multiple task (Lee et al., 2022) or language adapters (Rathore et al., 2023). Other approaches leverage soft prompts (Huang et al., 2022; Philippy et al., 2024) or grammar prompting (Wang et al., 2024). While many works focus on specific tasks, our study explored different combinations of adapters and soft prompts for cross-lingual transfer on four tasks, minimizing the reliance on machine translation, which is often unreliable for low-resource languages.

[1] Code is available at: https://github.com/kinit-sk/adapter-prompt-evaluation

3 Methodology

We propose a comprehensive study on combinations of language and task representations using adapters and soft prompts. We evaluate for the first time the capabilities of **soft language prompts** in a systematic manner and evaluate the performance of diverse combinations of soft prompts and adapters for cross-lingual transfer. Our pipeline, illustrated in Figure 1, consists of three steps. Firstly, we train language adapters and soft prompts in 16 languages on unlabelled data. Secondly, along with trained language representation, we incorporate task representation into the LLM's architecture and train the task representation on four tasks in six high-resource languages. Finally, we evaluate the cross-lingual transfer on 10 mid- and low-resource languages and four selected tasks.

In the following sections, we first give details on methods that we investigate for representing language (Section 3.1) and task information (Section 3.2). We then explain the combinations of soft prompts and adapters we evaluate (Section 3.3).

3.1 Language Representation

Language Adapters. Previous work has investigated the efficiency of training language-specific transformation using the adapter architecture (Houlsby et al., 2019b). Pfeiffer et al. (2020) proposed a MAD-X framework, which includes training language adapters using the masked language modeling objective on unlabelled data. Inspired by language adapters proposed by the authors, we build upon their architecture and the approach used to train language adapters. In our settings, language adapters are incorporated into each transformer layer of the LLM and trained using unlabelled data from Wikipedia.

Soft Language Prompts. *Soft Prompt Tuning* offers a promising, parameter-efficient method for adapting LLMs. Previous work has predominantly explored task-specific soft prompts to enhance task transferability, typically focusing on a single language (Vu et al., 2022b; Asai et al., 2022). In contrast, we extend this approach by training language-specific soft prompts to guide multilingual LLMs toward a target language. Given that multilingual LLMs can generate responses in various languages, we defined a soft language prompt as a set of token embeddings prepended to the original input embedding. These embeddings are then fed into the LLM to condition its output in the desired language.

Existing studies have highlighted the importance of soft prompt initialization in optimizing the performance of LLMs. Lester et al. (2021) outlined three possible strategies: (1) *random initialization using a Gaussian distribution*; (2) *initialization from the model's vocabulary*; and (3) *initialization with the embeddings of output classes for classification tasks*. While each method has its strengths and limitations, none directly apply to our experiments, which focus on multilingual LLMs. To address this, we introduce a language-specific text instruction for soft prompt initialization (see Appendix C). In this approach, the text instruction is first embedded, and if its length is shorter than the required soft prompt size, the embedding is repeated until the desired length is achieved. This approach already steers the model to the desired language and can fasten the convergence during the training.

Language Modeling Objective. Training language-specific representations requires unlabelled data from the selected languages and careful selection of an appropriate training objective. Given our use of an encoder-decoder architecture, we adopt *span corruption* as the training objective, which has been shown to be efficient in prior work (Raffel et al., 2020; Xue et al., 2021). Unlike the casual language modeling objective, where the LLM predicts the next token in a sequence, *span corruption* randomly masks 15% of the tokens in the input text using sentinel tokens. These tokens serve solely to mark the masked parts, which the LLM is tasked to reconstruct (Raffel et al., 2020). Finally, the LLM is trained to predict the original tokens for the masked portions, enabling it to learn linguistic nuances and patterns that are crucial for training task-specific adapters and soft prompts.

3.2 Task Representation

Task Adapters. Similarly to language adapters, we use task-specific adapters, represented by the same adapter architecture, which are incorporated into each transformer layer of the LLM. However, when combining task representations with language representations, the final architecture differs across configurations and depends on the type of language representation used during the training and inference. Detailed information on the architecture for all combinations is in Section 3.3.

Task adapters are updated only during training on the desired downstream task, while the rest of the model, along with the language representation, is kept frozen. In the case of task-specific representations, LLMs learn knowledge that is characteristic of the specified tasks, and that should be language-independent.

Soft Task Prompts. In addition to task adapters, we use soft task prompts with the same architecture and parameters as soft language prompts. The difference when using a soft task prompt occurs in the configuration consisting of a soft language prompt and a soft task prompt. With this configuration, both soft prompts are combined using a concatenation operation and further fed into the model to condition the final generation. This concatenation represents the stacking of two representations (language and task), similar to stacking two adapters on top of each other used in the MAD-X approach (Pfeiffer et al., 2020).

3.3 Evaluated Combinations of Adapters and Soft Prompts

Since our experiments are focused on evaluating language and task representations and their combinations, we define six possible configurations: (1) only *task adapter*; (2) only *soft task prompt*; (3) MAD-X (Pfeiffer et al., 2020), i.e. the combination of *language and task adapter*; (4) the combination of *language adapter and soft task prompt*; (5) the combination of *soft language prompt and task adapter*; and (6) the combination of *soft language prompt and soft task prompt*. The position of task representations within the LLM highly depends on the type of language representation used in experiments. The architecture, along with the form of the input for all configurations, are illustrated in Figure 2.

Single Task Adapters & Soft Task Prompts. The configurations that employ only task adapters or task soft prompts aim at training task representations without incorporating language-specific representation. Adapters and soft prompts were trained independently on each selected dataset from four tasks, and the resulting task representations were evaluated across all 10 mid- and low-resource languages. Only the adapters and soft prompts are trained during this process, while the rest of the LLM remains frozen.

Language Adapters & Task Adapters. Beyond training task representations alone, we also trained a task adapter on top of a pre-trained language adapter, reproducing the approach outlined in MAD-X (Pfeiffer et al., 2020). Our method leverages the same architecture with distinct training hyperparameters fitted to the tasks at hand. In this setup, the task adapter takes the output of the language adapter as input and further processes it. Only the task adapter is trained during training, while the language adapter and LLM remain frozen.

Adapters and Soft Prompts Combinations. Our study introduces two combinations of language and task representations using adapters and soft prompts. The first configuration involves soft task prompts along with a language adapter. This combination incorporates trained language-specific knowledge using a language adapter and a soft task prompt trained on the desired downstream task.

The second combination includes training a task adapter with the trained soft language prompt. Soft language prompts condition LLMs to activate knowledge specific to the desired language, while task adapters learn task-specific knowledge. In both settings, language representations are kept frozen and only task representations are trained on the desired downstream tasks.

Soft Language Prompts & Soft Task Prompts. The last configuration includes soft language and soft task prompts. Inspired by stacking language and task adapters on top of each other, we concatenated embeddings of language and task prompts to a final soft prompt, with the LLM and soft language prompt being frozen during training.

4 Experiments

In the following section, we describe the process of selection of models (Section 4.1), languages (Section 4.2) and tasks (Section 4.3).

4.1 Model Selection

We selected an encoder-decoder architecture, the mT0-BASE model, to conduct a cross-lingual evaluation. The mT0 model is based on the pre-trained multilingual mT5 model, which has been further fine-tuned on a collection of 46 languages across 16 NLP tasks (Muennighoff et al., 2023). The model selection played a crucial role in further experiments, and we conducted several preliminary experiments with the original mT5-BASE model. However, we observed that in the case of using pre-trained models, which have not been further fine-tuned on downstream tasks, prompt-tuning is

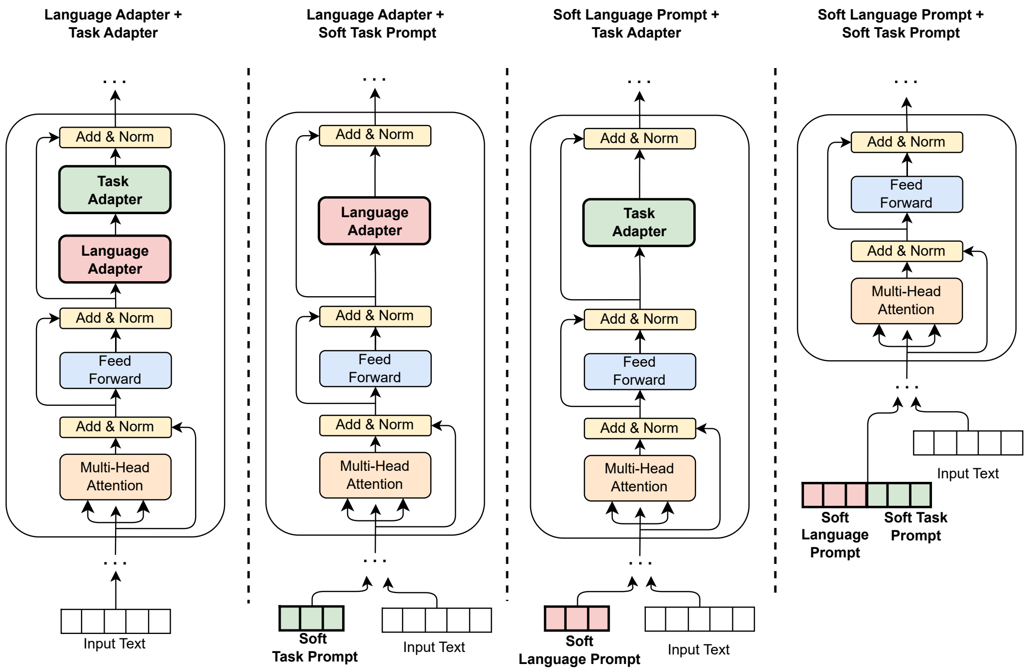
Language Adapter + Task Adapter
Language Adapter + Soft Task Prompt
Soft Language Prompt + Task Adapter
Soft Language Prompt + Soft Task Prompt
Add & Norm
Task Adapter
Language Adapter
Feed Forward
Multi-Head Attention
Input Text
Soft Task Prompt
Soft Language Prompt
...

Figure 2: The architecture for all combinations of language and task representations in our experiments. These configurations include: (1) Language and Task Adapters; (2) Language Adapter and Soft Task Prompt; (3) Soft Language Prompt and Task Adapter; and (4) Soft Language and Soft Task Prompts. Language representations are in red, while task representations are in green color.

not sufficient to guide the LLM to produce meaningful outputs.

4.2 Languages

The original mT5 model was pre-trained on over 100 languages, while mT0 employed only 46 for further fine-tuning. From the list of languages supported by mT5, we selected 16 languages and categorized them into two groups: high- and mid- along with low-resource languages. On the one hand, we consider *Arabic*, *German*, *English*, *Spanish*, *Russian* and *Chinese* to be high-resource languages. On the other hand, we consider *Czech*, *Greek*, *Romanian* and *Slovenian* as mid-resource and *Bulgarian*, *Malayalam*, *Slovak*, *Swahili*, *Telugu* and *Urdu* as low-resource languages. Our distinction between these two groups is based on the number of resources available for each language (in terms of unlabelled and labelled data).

We included languages from various families (e.g., Indo-European, Dravidian, Sino-Tibetan) and script types in the low-resource category, such as Latin, Arabic, Cyrillic and other non-Latin. The purpose of including multiple scripts and language families in our cross-lingual evaluation is to investigate the ability of the mT0 model to transfer knowledge between more similar and more distant languages with respect to both script and language features.

To train language representations on unlabelled data, we selected Wikipedia as a source that contains many articles in various languages, including low-resource ones. All Wikipedia data were taken from the latest preprocessed dump from HuggingFace[2], from November 2023.

4.3 Tasks

To evaluate the capabilities of mT0-Base and our proposed method for cross-lingual transfer, we choose four distinct tasks involving various NLP areas to explore the model performance. These tasks differ in the type of the provided output and include question answering (QA), named-entity recognition (NER), natural language inference (NLI), and check-worthy claim detection (CWCD). They were selected based on the availability of datasets for selected languages and to include various NLP tasks related to reading comprehension, recognizing textual entailment, or fact-checking. Table 1 lists the datasets used in our experiments. For Bulgarian,

[2] https://huggingface.co/datasets/wikimedia/wikipedia

Dataset	Task	Languages	Citation
SQuAD	QA	en	Rajpurkar et al. (2016)
MLQA	QA	ar, de, hi, zh, es, vi	Lewis et al. (2019)
XQuAD	QA	el, ro	Artetxe et al. (2020)
SK-QuAD	QA	sk	Hládek et al. (2023)
Czech SQuAD	QA	cs	Macková and Straka (2020)
TeQuAD	QA	te	Vemula et al. (2022)
KenSWQuAD	QA	sw	Wanjawa et al. (2023)
UQA	QA	ur	Arif et al. (2024)
Slovene SQuAD	QA	sl	Borovič et al. (2022)
IndicQA	QA	ml	Doddapaneni et al. (2023)
WikiANN	NER	ar, bg, cs, de, el, en, es, ml, ro, ru, sl, sk, sw, te, ur, zh	Rahimi et al. (2019)
XNLI	NLI	ar, bg, de, el, en, es, ru, sw, ur, zh	Conneau et al. (2018)
IndicXNLI	NLI	ml, te	Aggarwal et al. (2022)
CS ANLI	NLI	cs, sk*	CS-ANLI
RoNLI	NLI	ro	Poesina et al. (2024)
Sl-NLI	NLI	sl	Klemen et al. (2024)
MultiClaim	CWCD	ar, bg, cs, de*, el*, en, es, ml*, ro*, ru*, sl*, sk, sw*, te*, ur*, zh*	Pikuliak et al. (2023) Hyben et al. (2023)

Table 1: The list of datasets used in our experiments. Languages marked with * represent language versions of datasets that are not original but were obtained by translating texts from Czech (CS ANLI) or English (MultiClaim).

no question answering dataset is available.

Due to the absence of datasets for some languages, we employed Google Translate API to translate data for several languages. This concerns, in particular, the dataset for the Slovak NLI task and the dataset for check-worthy claim detection. In the case of the missing Slovak NLI dataset, we utilized the CS ANLI dataset and translated it from Czech to Slovak. For check-worthy claim detection, we translated the English dataset into multiple languages to obtain results for comparison.[3]

4.4 Experimental Setup

Language Representations. Language adapters and soft prompts were trained using a *span corruption* objective with different learning rates for training language adapters and soft language prompts, which were identified based on experiments on English data. Detailed parameters are listed in Table 2 in Appendix D.

Task Representations. In training task representations, we divided the training set into training and validation splits using 15% of the records for validation, which was done only for datasets that do not include a test set and the original validation split was considered a test set. This is especially the case for the question answering and check-worthy claim detection tasks. Secondly, we preprocessed each dataset by transforming each record from the particular dataset into the text-to-text format employing prompt templates listed in Appendix B.

Task representations in all configurations were trained using the same training parameters across all tasks, with differences only between learning rates and weight decay.[4] In addition, the instruction used for training soft prompts differs across languages and tasks. These variations are based on the language in which the answer is to be generated and the task that the LLM is solving.

The best model was chosen based on the performance on the validation split with respect to the loss. For classification tasks, we set the maximum number of tokens to generate based on the predicted classes. This minimizes the problem of the LLM continuing to generate an answer and enables us to evaluate the LLM's performance correctly. Table 2 in Appendix D shows the exact parameters for training language and task representations.

Evaluation. For evaluation, we selected several standard metrics employed for particular tasks. Specifically, we use the F1-Score or Accuracy for classification tasks and QA in the SQuAD format. Besides the F1-Score for QA, we also calculated Exact Match, assessing how many of the answers exactly match the ground truth.[5] For the evaluation, we employed metrics implemented in the Hugging Face evaluate library[6].

We evaluated the results on cross-lingual transfer from high-resource languages to mid- and low-resource ones, where task representations were trained on datasets in high-resource languages. We aim to assess the combination of language representations of low-resource languages with task representations trained on datasets from high-resource languages, i.e., high-resource language as source language and low-resource as target ones. Extended results are shown in Appendix E.

[3] To evaluate the accuracy of the translations, we manually verified a subset of samples, with a particular focus on translations between Czech and Slovak, leveraging input from native speakers. Our analysis found that the translations generated by Google Translate were correct for this language pair.

[4] We employed only one seed due to computational and time limitations. However, we performed a check of the generalizability of the approach by training the task representation on the German version of the WikiANN dataset for NER using two additional seeds and evaluated cross-lingual transfer from German to six languages. The results are in Appendix F.

[5] Exact Match tends to underestimate models' performance for low-resource languages, where LLMs are not often able to produce the exact answer with the correct grammar.

[6] https://huggingface.co/docs/evaluate

Baselines. To evaluate the proposed methods, we employed several baseline approaches and configurations. Baselines include task adapters, soft task prompts (prompt-tuning approach), and MAD-X – the combination of a language and task adapter (Pfeiffer et al., 2020). These baselines provided a foundation for assessing the effectiveness of cross-lingual transfer in our experiments.

5 Results and Analyis

This section summarizes our observations based on the experiments that were conducted. We provide the overall findings in Section 5.1 and then the results for particular tasks, especially question answering (Section 5.2), named entity recognition (Section 5.3), natural language inference (Section 5.4) and check-worthy claim detection tasks (Section 5.5).

5.1 Overall Results

Our study on cross-lingual transfer performance between high-resource and mid- and low-resource languages is summarized in Table 3 in Appendix E, which reports averaged metrics across four tasks for mid- and low-resource languages. Additionally, Figure 3 demonstrates the comparison of all combinations across high-resource languages, where the presented scores represent the average calculated across all tasks and all low-resource languages.

The results demonstrate that **the selection of source languages plays an important role in the overall results**, with distinct languages demonstrating different performance gains. Using English as a source language yielded the highest performance for most mid- and low-resource languages when employing task representations alone. A possible explanation might be that multilingual models often remain biased toward the source language, even after adaptation, as demonstrated in Alabi et al. (2024). They show that language adaptation in models primarily occurs in the final layers, while the source language still influences earlier predictions. However, for Bulgarian and Slovak, the combination of soft language prompts with task adapters proved to be more effective.

In contrast, when Arabic, German, Spanish, and Russian were used as source languages, configurations that combined language and task representations yielded superior scores. Specifically, **transferring knowledge from Spanish using a combination of soft language prompts and task adapters resulted in the highest overall performance**. This configuration proved to be the most effective, making Spanish the most effective high-resource language for cross-lingual transfer between languages across various scripts. The superior performance of Spanish can be attributed to its performance gain in both NER and CWCD tasks compared to other high-resource languages. For instance, in NER, the gap between the best and second-best averaged performance was approximately 4%. The most significant improvement was observed when transferring knowledge to Romanian, likely due to the linguistic similarity between Spanish and Romanian. Additionally, Spanish also shares similarities with some Slavic languages, further contributing to its effectiveness in cross-lingual transfer.

5.2 Question Answering

Our experiments (see Table 4) revealed that **the configuration of a soft language prompt and task adapter achieved the highest performance in many cases in the QA task** when transferring to mid- and low-resource languages, with only small differences across languages. This configuration was particularly effective for Greek, Romanian and Slovak, while for Telugu and Urdu, the task adapter without language representation outperformed other configurations. This suggests that **the complexity of the target language cannot be sufficiently modelled based on the small number of Wikipedia articles** in those languages. Furthermore, English excelled across the board, particularly with Latin and Greek scripts, showcasing its adaptability in cross-lingual transfer.

In addition to investigating the effects of individual configurations, we also evaluated the improvement of a soft language prompt combined with a task adapter over the original mT0-Base model without any language or task representations (see Figure 4). Figure 4 contains relative F1-Score improvements and demonstrates that **training task representations in English and evaluating in other languages provide the most evident improvement**. We also observed that German, English, and Spanish improved performance for most low-resource languages except Telugu and Malayalam. In contrast, Arabic, Russian and Chinese, which have different scripts, exhibited negative transfer across all cases, with Arabic and Chinese offering no improvement for any languages. We conjecture that the cross-lingual transfer de-

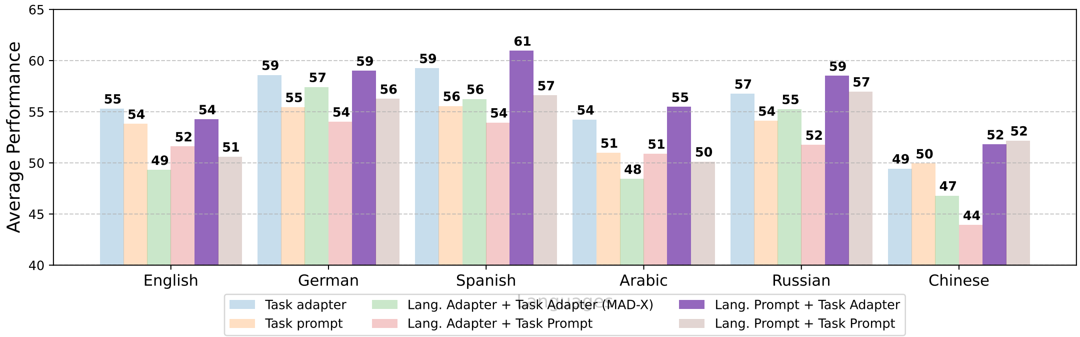

Figure 3: Average performance for the cross-lingual transfer from six high-resource to all low-resource languages, averaged over all low-resource languages. The graph compares different configurations with varying performance for cross-lingual transfer from high-resource to low-resource languages. In most cases, combining soft language prompts with task adapters (purple) proved best.

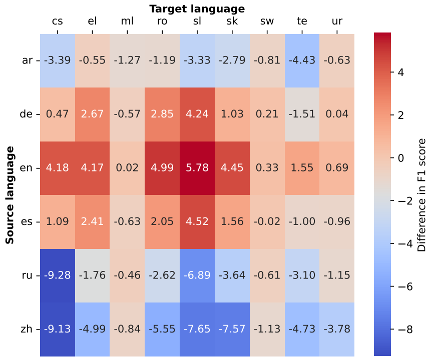

Figure 4: Relative F1 improvement for the QA task in transferring knowledge between languages using soft language prompts and task adapters. The effectiveness of the selected configuration is compared with the results obtained without using any language and task representations (i.e., mT0-BASE inference).

pends on the script used for the language, where we achieved the highest performance for languages in the Latin script.

5.3 Named Entity Recognition

In the case of the NER task, **Arabic, German, Spanish and Russian, among high-resource languages, performed best in cross-lingual transfer** to mid- and low-resource languages, while English and Chinese performed poorly. However, based on Table 5, the best improvements were observed using a soft language prompt with a task adapter, outperforming the other combinations for languages such as Arabic, Spanish and German. This is especially the case for Telugu, where the difference between these configurations is more than 37% in favor of the combination of soft language prompt and task adapter using Russian data.

5.4 Natural Language Inference

The cross-lingual evaluation of the NLI task from Table 6 demonstrated the effectiveness of almost all proposed configurations for knowledge transfer. In particular, we mostly achieve **superior results** using the combination of **language adapters with soft task prompts in Czech, Slovenian and Slovak as target languages**. Meanwhile, the best performance was achieved for Swahili, Telugu, and Urdu without employing language representations. Furthermore, the high effect on the Romanian language observed in the cross-lingual evaluation is probably because Romanian has been involved during the training of mT5, but not as part of fine-tuning the mT0 model.

We observed improvement for most languages across the six proposed configurations for transferring knowledge. However, for Czech, Slovenian and Slovak, several configurations resulted in lower performance compared to inference-only baselines. Notably, **for Slovenian, using Russian for the soft task prompt was the only configuration that outperformed the inference-only approach**. Furthermore, the combination of language and task adapters for Slovenian resulted in the poorest performance, with an average deterioration of 63%.

5.5 Check-Worthy Claim Detection

For check-worthy claim detection, the configuration of soft language prompts and task adapters performs comparably to methods without language representations (see Table 7). When considering both the best and second-best results, this combination proves effective across most language pairs, demonstrating the model's enhanced capabilities for check-worthy claim detection. Notably, **using**

Spanish for knowledge transfer within this setup resulted in the highest performance gains.

6 Discussion

Based on our experiments, we summarize our observations below.

Prompt Tuning Performs Better with Fine-Tuned Models. In our preliminary model selection experiments, we found that prompt tuning does not improve the performance of pre-trained LLMs (e.g., mT5) trained only on unlabelled data for downstream tasks. However, prompt-tuning can enhance the performance of already fine-tuned LLMs on any labelled data, even if the specific tasks were not part of the prior fine-tuning. This was confirmed in our experiments with NER and check-worthy claim detection, where fine-tuned LLMs delivered superior results despite no previous task-specific training on these tasks.

Soft Language Prompts with Task Adapters Perform Best in Many Cases. Our approach of combining soft language prompts with task adapters demonstrated better performance in many cases, compared to the approach of combined language and task adapters, which has been shown to be very effective in previous work. Specifically, the combination of soft language prompts and task adapters is most effective on the classification tasks, achieving superior results most often. For languages with a different script (e.g., Spanish and Telugu), these differences were over 20%.

Language Representations are Unable to Capture Linguistic Characteristics Using Small Number of Unlabelled Data. Language representations have several limitations that led to configurations without language representations performing consistently better on cross-lingual transfer to highly low-resource languages, such as Telugu, Urdu, and Malayalam. We postulate that the reason is the small number of Wikipedia articles on which the language representations were trained, rendering them unable to adequately capture sufficient linguistic characteristics.

7 Conclusion

Our study provides a comprehensive evaluation of various configurations of adapters and soft prompts for cross-lingual transfer in mid- and low-resource languages. With the systematic evaluation of task adapters, soft task prompts, and combinations of language and task representations, we identified configurations that positively affect LLM's performance across different tasks and languages. Our findings demonstrated that the combination of soft language prompts and task adapters emerged as an effective alternative for transferring knowledge between languages. Furthermore, our findings provide valuable insights for the utilization of a combination of PEFT methods for cross-lingual transfer, while highlighting the need to incorporate language-specific knowledge.

Limitations

Model Selection. Our analysis of the effectiveness of the language and task representations focused on highly multilingual LLMs that include a wider variety of low-resource languages. From this perspective, there is not a vast number of open-source multilingual LLMs with such extensive language coverage as the mT5 or BLOOM model, while having fewer than 1B parameters. We also considered the AYA model (Üstün et al., 2024), but due to limited computational resources, it was not feasible to conduct our experiments. Another aspect of the selection was the involvement of only generative models consisting of encoder-decoder or decoder-only architecture.

Other Languages. In selecting appropriate languages, we were limited by the languages covered by the mT5 model. To select high-resource languages, we considered languages that are the most extensive in terms of available resources and are in different scripts, e.g., not only Latin script. On the other hand, when selecting mid- and low-resource languages, we also considered the availability of datasets in multiple languages from different language families as well as the availability of datasets in those languages (both human-annotated and machine-translated).

Other Tasks. The tasks in our experiments were selected based on the availability of datasets for each selected language and covered multiple areas of the NLP domain, i.e., reading comprehension, fact-checking, and recognizing textual entailment. We mostly considered tasks involved in the instruction fine-tuning of the mT0-model, but we also included tasks that were not originally used to train mT0, e.g., NER and check-worthy claim detection.

Acknowledgements

We thank the anonymous reviewers for their feedback; and especially for spending their time through several iterations of the ARR cycles. This research was partially supported by *DisAI - Improving scientific excellence and creativity in combating disinformation with artificial intelligence and language technologies*, a project funded by Horizon Europe under GA No.101079164, by *MIMEDIS*, a project funded by the Slovak Research and Development Agency under GA No. APVV-21-0114, by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254), and by the German Ministry of Education and Research (BMBF) as part of the project TRAILS (01IW24005).

References

CS ANLI. `https://huggingface.co/datasets/ctu-aic/anli_cs`. Accessed: 2024-05-30.

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.

Jesujoba Alabi, Marius Mosbach, Matan Eyal, Dietrich Klakow, and Mor Geva. 2024. The hidden space of transformer language adapters. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6588–6607, Bangkok, Thailand. Association for Computational Linguistics.

Alan Ansell, Edoardo Ponti, Anna Korhonen, and Ivan Vulić. 2022. Composable sparse fine-tuning for cross-lingual transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1778–1796, Dublin, Ireland. Association for Computational Linguistics.

Alan Ansell, Edoardo Maria Ponti, Jonas Pfeiffer, Sebastian Ruder, Goran Glavaš, Ivan Vulić, and Anna Korhonen. 2021. MAD-G: Multilingual adapter generation for efficient cross-lingual transfer. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4762–4781, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Samee Arif, Sualeha Farid, Awais Athar, and Agha Ali Raza. 2024. UQA: Corpus for Urdu question answering. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 17237–17244, Torino, Italia. ELRA and ICCL.

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, Mohammadreza Salehi, Matthew Peters, and Hannaneh Hajishirzi. 2022. ATTEMPT: Parameter-efficient multi-task tuning via attentional mixtures of soft prompts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6655–6672, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mladen Borovič, Kristjan Žagar, Marko Ferme, Sandi Majninger, Milan Ojsteršek, Uroš Šmajdek, Maj Zirkelbach, Matjaž Zupanič, Meta Jazbinšek, Slavko Žitnik, et al. 2022. Slovene translation of the squad2. 0 dataset.

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.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. 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*. Association for Computational Linguistics.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115. Curran Associates, Inc.

Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M. Khapra, Anoop Kunchukuttan, and Pratyush Kumar. 2023. Towards leaving no Indic language behind: Building monolingual corpora, benchmark and models for Indic languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12402–12426, Toronto, Canada. Association for Computational Linguistics.

Daniil Gurgurov, Mareike Hartmann, and Simon Ostermann. 2024. Adapting multilingual LLMs to low-resource languages with knowledge graphs via adapters. In *Proceedings of the 1st Workshop on Knowledge Graphs and Large Language Models (KaLLM 2024)*, pages 63–74, Bangkok, Thailand. Association for Computational Linguistics.

Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. 2022. Towards a unified view of parameter-efficient transfer learning. *Preprint*, arXiv:2110.04366.

Daniel Hládek, Ján Staš, Jozef Juhár, and Tomáš Koctúr. 2023. Slovak dataset for multilingual question answering. *IEEE Access*, 11:32869–32881.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019a. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019b. 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.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multitask benchmark for evaluating cross-lingual generalisation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4411–4421. PMLR.

Lianzhe Huang, Shuming Ma, Dongdong Zhang, Furu Wei, and Houfeng Wang. 2022. Zero-shot cross-lingual transfer of prompt-based tuning with a unified multilingual prompt. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11488–11497, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Martin Hyben, Sebastian Kula, Ivan Srba, Robert Moro, and Jakub Simko. 2023. Is it indeed bigger better? the comprehensive study of claim detection lms applied for disinformation tackling. *Preprint*, arXiv:2311.06121.

Matej Klemen, Aleš Žagar, Jaka Čibej, and Marko Robnik-Šikonja. 2024. SI-NLI: A Slovene natural language inference dataset and its evaluation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14859–14870, Torino, Italia. ELRA and ICCL.

Mateusz Klimaszewski, Piotr Andruszkiewicz, and Alexandra Birch. 2024. No train but gain: Language arithmetic for training-free language adapters enhancement. *Preprint*, arXiv:2404.15737.

Jenny Kunz and Oskar Holmström. 2024. The impact of language adapters in cross-lingual transfer for nlu. *Preprint*, arXiv:2402.00149.

Jaeseong Lee, Seung-won Hwang, and Taesup Kim. 2022. FAD-X: Fusing adapters for cross-lingual transfer to low-resource languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 57–64, Online only. 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.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2019. Mlqa: Evaluating cross-lingual extractive question answering. *arXiv preprint arXiv:1910.07475*, arXiv: 1910.07475.

Lei Liu and Jimmy Xiangji Huang. 2023. Prompt learning to mitigate catastrophic forgetting in cross-lingual transfer for open-domain dialogue generation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 2287–2292, New York, NY, USA. Association for Computing Machinery.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. 2024. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *Preprint*, arXiv:2308.08747.

Kateřina Macková and Milan Straka. 2020. Reading comprehension in czech via machine translation and cross-lingual transfer. In *Text, Speech, and Dialogue*, pages 171–179, Cham. Springer International Publishing.

Michael McCloskey and Neal J. Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. volume 24 of *Psychology of Learning and Motivation*, pages 109–165. Academic Press.

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. *Preprint*, arXiv:2211.01786.

Marinela Parović, Goran Glavaš, Ivan Vulić, and Anna Korhonen. 2022. BAD-X: Bilingual adapters improve zero-shot cross-lingual transfer. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages

1791–1799, Seattle, United States. Association for Computational Linguistics.

Jonas Pfeiffer, Naman Goyal, Xi Lin, Xian Li, James Cross, Sebastian Riedel, and Mikel Artetxe. 2022. Lifting the curse of multilinguality by pre-training modular transformers. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3479–3495, Seattle, United States. 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. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1978–2008, Singapore. 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.

Fred Philippy, Siwen Guo, Shohreh Haddadan, Cedric Lothritz, Jacques Klein, and Tegawendé F. Bissyandé. 2024. Soft prompt tuning for cross-lingual transfer: When less is more. In *Proceedings of the 1st Workshop on Modular and Open Multilingual NLP (MOOMIN 2024)*, pages 7–15, St Julians, Malta. Association for Computational Linguistics.

Matúš Pikuliak, Ivan Srba, Robert Moro, Timo Hromadka, Timotej Smolen, Martin Melisek, Ivan Vykopal, Jakub Simko, Juraj Podrouzek, and Maria Bielikova. 2023. Multilingual previously fact-checked claim retrieval. *Preprint*, arXiv:2305.07991.

Matúš Pikuliak, Marián Šimko, and Mária Bieliková. 2021. Cross-lingual learning for text processing: A survey. *Expert Systems with Applications*, 165:113765.

Eduard Poesina, Cornelia Caragea, and Radu Ionescu. 2024. A novel cartography-based curriculum learning method applied on RoNLI: The first Romanian natural language inference corpus. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 236–253, Bangkok, Thailand. 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. *Preprint*, arXiv:2005.00333.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, 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.

Vipul Rathore, Rajdeep Dhingra, Parag Singla, and Mausam. 2023. ZGUL: Zero-shot generalization to unseen languages using multi-source ensembling of language adapters. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6969–6987, Singapore. Association for Computational Linguistics.

Weijieying Ren, Xinlong Li, Lei Wang, Tianxiang Zhao, and Wei Qin. 2024. Analyzing and reducing catastrophic forgetting in parameter efficient tuning. *Preprint*, arXiv:2402.18865.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *Preprint*, arXiv:2008.00401.

Ahmet Üstün, Arianna Bisazza, Gosse Bouma, and Gertjan van Noord. 2020. UDapter: Language adaptation for truly Universal Dependency parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2302–2315, Online. Association for Computational Linguistics.

Rakesh Vemula, Mani Nuthi, and Manish Srivastava. 2022. TeQuAD:Telugu question answering dataset. In *Proceedings of the 19th International Conference on Natural Language Processing (ICON)*, pages 300–307, New Delhi, India. Association for Computational Linguistics.

Tu Vu, Aditya Barua, Brian Lester, Daniel Cer, Mohit Iyyer, and Noah Constant. 2022a. 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.

Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou', and Daniel Cer. 2022b. SPoT: Better frozen model adaptation through soft prompt transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5039–5059, Dublin, Ireland. Association for Computational Linguistics.

Bailin Wang, Zi Wang, Xuezhi Wang, Yuan Cao, Rif A. Saurous, and Yoon Kim. 2024. Grammar prompting for domain-specific language generation with large language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Barack W. Wanjawa, Lilian D. A. Wanzare, Florence Indede, Owen Mconyango, Lawrence Muchemi, and Edward Ombui. 2023. Kenswquad—a question answering dataset for swahili low-resource language. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 22(4).

BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa

Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. 2023. Bloom: A 176b-parameter open-access multilingual language model. *Preprint*, arXiv:2211.05100.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual bert? *Preprint*, arXiv:2005.09093.

Zhihui Xie, Handong Zhao, Tong Yu, and Shuai Li. 2024. Discovering low-rank subspaces for language-agnostic multilingual representations. *Preprint*, arXiv:2401.05792.

Lingling Xu, Haoran Xie, Si-Zhao Joe Qin, Xiaohui Tao, and Fu Lee Wang. 2023. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *Preprint*, arXiv:2312.12148.

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. *Preprint*, arXiv:2010.11934.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *Preprint*, arXiv:2303.10512.

Ahmet Üstün, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, Freddie Vargus, Phil Blunsom, Shayne Longpre, Niklas Muennighoff, Marzieh Fadaee, Julia Kreutzer, and Sara Hooker. 2024. Aya model: An instruction finetuned open-access multilingual language model. *Preprint*, arXiv:2402.07827.

A Computational Resources

For our experiments, we utilized a computational infrastructure consisting of A10 and A40 NVIDIA GPUs, while our experiments ran in parallel on multiple GPUs. In total, our experiments required around 3,200 GPU hours, ensuring model training and validation for cross-lingual transfer.

B Prompts Used

For the purpose of the encoder-decoder model, the record from each dataset needs to be transformed into a text-to-text format. To choose an appropriate prompt format, we experimented with all the prompts used in the mT0 paper (Muennighoff et al., 2023) and with prompts used in the T5 paper (Raffel et al., 2020). Prompts, which achieved the best performance during inference with the mT0-BASE model, were selected for transforming the records into a text-to-text format. In the following paragraphs, there are the prompts for the individual tasks that have been used to convert to text-to-text format.

B.1 Question Answering

Template: `question: {question} context: {context}`

B.2 Natural Language Inference

Template: `{premise} \n\n Question: Does this imply that "{hypothesis}"? Yes, no, or maybe?`

B.3 Named-Entity Recognition

Template: `tag: {text}`

B.4 Check-Worthy Claim Detection

Template: `checkworthiness claim: {claim}`

C Soft Prompt Initialization

This section includes templates for soft prompts used for the initialization for each language and each task. Templates are divided into language and task templates.

C.1 Language Templates

To train language representation using a language modeling objective, we employed a specific prompt that varied only based on the language present in the instruction, leaving the rest of the instruction the same.

The template we used for initialization is as follows: `"Generate the output in {Language}:"`, where the `Language` is replaced by the desired language.

C.2 Task Templates

The following are initialization prompt templates for each task, where the instruction depends not only on the task but also on the language.

Question Answering. For the question answering task, we utilized `"Answer the question in {Language} language:"`, while replacing `Language` with the desired language.

Natural Language Inference. Natural language inference is the task of assessing whether a hypothesis logically follows from the premise. It is defined as a classification with three possible classes: *entailment*, *contradiction* or *neutral*. However, based on the previous work and instruction tuning of the mT0 model, we replaced above mentioned classes with *Yes*, *No* and *Maybe*, based on the used prompt template.

According to the employed classes, we defined an initialization prompt as follows: `"Select Yes, No or Maybe based on the implication of the premise on the hypothesis in {Language}:"`, while `Language` is replaced by the desired language.

Named-Entity Recognition. The named-entity recognition task aims to identify named entities within the input text. While there are many possible categories, the WikiANN dataset focuses only on detecting three categories: location (LOC), person (PER) and organization (ORG). Based on the defined classes, we created the initialization prompt as follows: `"Identify NER tags (ORG, PER, LOC) in the text in {Language}:"`, where `Language` is substituted with the specific language.

Check-Worthy Claim Detection. The latter task includes check-worthy claim detection, which is a binary classification of assessing whether the given claim is worthy of fact-checking or not. As text labels, we used *Not checkworthy* and *Checkworthy*. This is the initialization prompt for the check-worthy claim detection task: `Determine whether a given claim in {Language} is checkworthy:"`, where `Language` is replaced by the desired language.

D Hyperparameters

Table 2 shows hyperparameters used for training language and task representations using adapters and soft prompts.

E Cross-Lingual Evaluation

Tables 4 to 7 present the results for transferring knowledge from all high-resource languages to all mid- and low-resource languages. The first row in each table represents the scores obtained by inference of the original mT0-Base model without additional training of language or task representations.

F Evaluation with Multiple Training Seeds

In Table 8, we report the evaluation results of all configurations that were trained on the German version of the WikiANN dataset using three different seeds. Along with the mean values, we also report the standard deviation

The obtained results demonstrate that the best results for knowledge transfer from German to other languages are obtained by using task adapters for Bulgarian, Greek, Malayalam, Romanian and Swahili. In contrast, the best combination for Czech, Slovenian, Slovak, Telugu and Urdu was a soft language prompt with a task adapter. This observation supports our previous findings that both configurations achieved superior results on the NER task when transferring knowledge from German.

Hyperparameters	Language Modeling		Task Modeling	
	Language Adapter	Soft Language Prompt	Task Adapter	Soft Task Prompt
Learning rate	5e-5	5e-1	5e-5	5e-1
Weight decay	0	1e-5	0	1e-5
Batch size	32	32	32	32
No. Training steps	100,000	100,000	50,000	50,000
Optimizer	AdamW	Adafactor	AdamW	Adafactor
Evaluation steps	500	500	1000	1000
Max input length	256	256	256	256
Token size of soft prompt	NaN	50	NaN	50

Table 2: Final parameters employed to train language and task representation using adapters and soft prompts.

Task Language	Language Representation	Task Representation	bg	cs	el	ml	ro	sl	sk	sw	te	ur	Average
ar	None	Adapter	68.03	**48.98**	65.25	46.74	63.44	48.74	48.26	50.86	**53.80**	48.15	54.23
		Soft Prompt	64.67	43.16	62.66	47.55	64.43	44.60	40.33	47.22	51.44	43.81	50.99
	Adapter	Adapter	64.82	42.56	64.59	**49.82**	61.73	30.44	39.92	48.99	40.36	41.03	48.43
		Soft Prompt	69.90	39.84	63.36	48.49	55.78	**55.53**	41.28	49.52	41.45	43.75	50.89
	Soft Prompt	Adapter	**71.23**	48.06	**67.22**	48.22	**65.66**	51.68	**48.26**	**51.21**	53.54	**49.63**	**55.47**
		Soft Prompt	66.73	41.59	62.82	47.66	58.81	44.69	40.02	44.10	50.11	44.72	50.13
de	None	Adapter	67.76	51.72	71.12	51.52	**71.55**	54.88	51.27	**56.78**	**57.50**	51.82	58.59
		Soft Prompt	65.31	47.61	69.55	**51.77**	70.02	50.97	46.53	53.16	52.91	46.66	55.45
	Adapter	Adapter	**79.98**	**53.81**	**73.48**	50.39	66.34	49.69	**54.57**	54.29	45.65	45.88	57.41
		Soft Prompt	72.61	47.19	68.12	48.81	60.14	56.41	46.63	51.20	44.41	44.76	54.03
	Soft Prompt	Adapter	70.71	51.91	70.83	50.86	69.86	**57.76**	53.98	55.90	55.90	**52.34**	**59.01**
		Soft Prompt	68.98	49.02	69.83	50.98	69.18	52.18	48.36	54.50	54.50	45.04	56.26
en	None	Adapter	68.77	**49.64**	**70.37**	46.34	67.29	52.54	47.98	49.70	52.13	48.29	**55.31**
		Soft Prompt	64.81	41.26	68.02	**46.60**	70.58	52.32	40.19	**51.43**	**52.43**	50.48	53.81
	Adapter	Adapter	60.90	46.51	65.03	41.80	72.02	37.96	43.30	46.98	38.47	40.09	49.31
		Soft Prompt	64.68	38.29	65.04	42.93	**74.50**	**54.80**	37.41	48.36	37.48	**52.54**	51.60
	Soft Prompt	Adapter	**68.95**	48.37	69.39	43.44	64.20	51.34	**50.88**	51.05	49.16	45.78	54.26
		Soft Prompt	56.33	47.75	65.98	43.22	58.94	52.64	46.48	48.96	47.12	38.62	50.60
es	None	Adapter	68.65	53.61	70.06	49.75	**73.83**	56.23	51.82	**57.47**	**57.12**	**54.01**	59.26
		Soft Prompt	62.16	50.28	66.95	47.26	71.84	50.34	49.05	54.09	53.94	49.71	55.56
	Adapter	Adapter	73.89	50.03	73.33	51.15	70.92	45.79	53.04	54.24	45.45	44.63	56.25
		Soft Prompt	72.63	50.19	64.62	44.62	72.70	47.07	51.71	51.95	39.95	43.94	53.94
	Soft Prompt	Adapter	**75.37**	**55.23**	**73.50**	**51.39**	72.03	**58.88**	**54.84**	57.41	56.90	54.00	**60.96**
		Soft Prompt	69.07	52.16	64.43	48.19	72.98	53.21	50.93	53.87	50.89	50.26	56.60
ru	None	Adapter	82.44	45.29	65.74	46.79	69.68	49.24	45.24	54.40	56.48	52.27	56.76
		Soft Prompt	79.26	41.51	60.83	48.64	68.09	45.29	40.70	50.73	55.13	50.94	54.11
	Adapter	Adapter	80.74	**52.52**	**73.86**	45.76	**73.75**	40.19	**51.99**	52.08	39.55	42.11	55.26
		Soft Prompt	77.96	34.33	66.75	44.48	71.76	49.84	36.41	50.61	40.14	45.31	51.76
	Soft Prompt	Adapter	**83.96**	44.69	70.25	49.87	72.15	51.06	47.34	**55.21**	**56.89**	**53.62**	**58.50**
		Soft Prompt	79.68	44.50	71.02	**50.63**	70.94	**52.60**	40.99	52.37	54.91	52.04	56.97
zh	None	Adapter	61.19	43.74	57.03	40.50	64.60	44.42	42.70	50.23	45.65	44.33	49.44
		Soft Prompt	58.20	45.17	59.24	42.23	66.36	44.72	43.38	46.27	50.10	43.72	49.94
	Adapter	Adapter	62.73	46.45	61.14	37.45	66.19	35.30	42.70	43.73	35.98	36.28	46.80
		Soft Prompt	49.40	43.50	47.04	42.18	45.70	**56.09**	36.34	42.75	37.22	39.00	43.92
	Soft Prompt	Adapter	**65.97**	45.20	60.49	41.62	65.64	48.49	**45.52**	**52.19**	50.20	42.72	51.80
		Soft Prompt	62.44	**48.74**	**61.62**	**43.09**	**67.48**	49.20	44.85	48.53	**51.09**	**44.69**	**52.17**

Table 3: Average scores for each configuration across all tasks for low-resource languages. The languages in rows represent the language in which the task representation was trained, and the languages in columns represent the language representation that was used, if any (except for configurations with None in the language representation). For each language pair, the best results are **boldfaced** and the second best are underlined.

Task Language	Language Representation	Task Representation	cs	el	ml	ro	sl	sk	sw	te	ur	Average
	None	None	31.34 (24.78)	57.00 (47.56)	1.37 (1.07)	57.00 (47.56)	31.50 (22.58)	26.39 (9.78)	3.24 (0.36)	18.64 (12.10)	13.37 (7.02)	
ar	None	Adapter	**29.14 (21.80)**	55.90 (46.30)	0.22 (18.94)	55.90 (46.30)	27.83 (19.74)	23.15 (8.44)	2.39 (0.36)	**15.70 (11.00)**	**13.00 (10.38)**	**24.80 (20.36)**
		Soft Prompt	22.93 (18.02)	56.08 (46.89)	0.10 (0.82)	56.08 (46.89)	23.91 (17.33)	19.89 (7.85)	0.56 (0.18)	11.53 (8.60)	12.05 (8.29)	22.57 (17.21)
	Adapter	Adapter	25.60 (14.45)	45.96 (35.71)	**1.12 (4.15)**	43.17 (31.85)	23.55 (11.82)	22.84 (6.03)	1.77 (0.18)	9.54 (4.90)	9.04 (4.08)	20.29 (12.57)
		Soft Prompt	24.70 (1.67)	52.76 (42.94)	0.95 (0.63)	46.35 (35.38)	27.65 (16.04)	**25.10 (1.28)**	**2.57 (1.09)**	10.50 (5.10)	10.43 (5.84)	22.33 (12.22)
	Soft Prompt	Adapter	27.95 (20.78)	**56.45 (46.97)**	0.10 (21.27)	55.81 (46.47)	**28.17 (19.88)**	23.60 (8.72)	2.43 (0.45)	14.21 (10.10)	12.74 (10.07)	24.61 (20.52)
		Soft Prompt	27.14 (20.55)	55.51 (46.72)	0.03 (0.31)	**56.11 (47.23)**	26.00 (18.62)	20.96 (8.04)	0.77 (0.27)	11.04 (8.30)	12.37 (8.63)	23.33 (17.63)
de	None	Adapter	35.37 (27.29)	58.88 (49.08)	0.99 (2.27)	58.88 (49.08)	36.15 (24.66)	27.51 (10.16)	**3.84 (0.82)**	**18.81 (12.80)**	**14.02 (10.27)**	**28.27 (20.71)**
		Soft Prompt	28.56 (21.31)	57.12 (47.65)	0.57 (2.20)	57.12 (47.65)	30.34 (20.52)	24.54 (8.92)	1.93 (0.73)	12.46 (9.70)	10.54 (8.21)	24.80 (18.54)
	Adapter	Adapter	36.78 (27.82)	57.46 (47.31)	1.05 (1.38)	57.69 (47.90)	38.03 (25.07)	**31.43 (11.35)**	3.01 (0.54)	11.32 (6.00)	9.62 (4.72)	27.37 (19.12)
		Soft Prompt	**38.13 (31.20)**	54.70 (45.29)	**1.88 (0.94)**	55.11 (46.55)	**38.88 (27.69)**	30.70 (11.98)	3.13 (1.00)	16.05 (8.70)	12.63 (8.01)	27.91 (20.15)
	Soft Prompt	Adapter	31.81 (24.29)	**59.67 (48.91)**	0.80 (2.08)	**59.85 (49.08)**	35.74 (23.84)	27.42 (10.16)	3.45 (0.36)	17.13 (11.20)	13.41 (10.25)	27.70 (20.02)
		Soft Prompt	32.68 (24.65)	57.86 (48.15)	0.58 (8.68)	58.38 (49.08)	30.67 (20.78)	27.95 (10.30)	2.25 (1.09)	12.34 (9.50)	9.35 (9.58)	25.78 (20.20)
en	None	Adapter	**36.95 (28.57)**	60.24 (50.34)	1.18 (0.94)	60.24 (50.34)	37.04 (26.07)	30.11 (11.46)	3.21 (0.36)	19.65 (12.70)	13.93 (9.37)	29.17 (21.13)
		Soft Prompt	33.59 (25.55)	60.35 (50.76)	0.82 (0.94)	60.35 (50.76)	34.77 (24.36)	27.76 (10.07)	2.81 (0.45)	19.38 (12.80)	13.81 (9.25)	28.18 (20.55)
	Adapter	Adapter	33.75 (24.33)	57.44 (48.24)	1.25 (1.70)	58.11 (49.33)	31.65 (19.76)	28.98 (10.58)	3.25 (0.63)	9.85 (5.00)	9.06 (4.20)	25.93 (18.20)
		Soft Prompt	33.94 (26.22)	58.49 (49.58)	0.97 (0.25)	57.72 (48.74)	35.89 (24.36)	29.82 (11.54)	3.06 (0.45)	12.67 (6.60)	9.67 (4.94)	26.91 (19.19)
	Soft Prompt	Adapter	35.52 (27.27)	**61.17 (51.68)**	**1.39 (2.71)**	**61.99 (52.10)**	**37.28 (25.96)**	**30.84 (11.73)**	**3.57 (0.27)**	**20.19 (13.50)**	**14.06 (10.54)**	**29.56 (21.75)**
		Soft Prompt	35.35 (27.02)	59.80 (50.34)	0.68 (2.20)	60.02 (50.34)	36.49 (26.02)	29.89 (11.39)	3.46 (0.54)	11.71 (7.30)	9.83 (5.68)	27.47 (20.09)
es	None	Adapter	33.72 (25.43)	**59.61 (50.34)**	**1.29 (1.20)**	**59.61 (50.34)**	34.97 (32.80)	27.24 (9.71)	**3.42 (0.82)**	**19.10 (12.50)**	**13.03 (9.48)**	**28.00 (20.40)**
		Soft Prompt	25.98 (19.06)	54.72 (45.29)	0.12 (0.31)	54.72 (45.29)	27.38 (18.50)	22.43 (7.62)	1.10 (0.36)	11.94 (9.00)	9.13 (6.84)	23.06 (16.92)
	Adapter	Adapter	**33.98 (23.88)**	55.44 (45.13)	1.06 (1.26)	56.58 (46.22)	34.52 (21.06)	28.66 (10.01)	2.34 (0.45)	10.76 (5.00)	9.12 (4.81)	25.83 (17.54)
		Soft Prompt	32.86 (24.45)	53.57 (43.78)	0.98 (0.57)	52.82 (43.45)	34.73 (23.35)	27.82 (10.35)	2.49 (0.54)	10.89 (5.40)	9.27 (4.17)	25.05 (17.34)
	Soft Prompt	Adapter	32.43 (24.92)	59.41 (49.75)	0.74 (1.07)	59.05 (49.24)	**36.02 (24.92)**	**27.95 (10.27)**	3.22 (0.45)	17.64 (11.40)	12.41 (8.42)	27.65 (20.05)
		Soft Prompt	27.21 (19.82)	54.39 (44.45)	0.27 (0.82)	54.93 (45.21)	28.25 (19.39)	21.49 (7.49)	1.11 (0.18)	9.28 (7.20)	8.36 (6.11)	22.81 (16.74)
ru	None	Adapter	27.45 (15.86)	**55.56 (42.52)**	0.73 (5.73)	**55.56 (42.52)**	24.81 (14.05)	23.69 (7.96)	**2.93 (0.45)**	16.80 (10.50)	**12.45 (10.12)**	24.44 (16.63)
		Soft Prompt	17.94 (8.27)	51.61 (37.73)	0.25 (0.13)	51.61 (37.73)	17.58 (8.13)	16.98 (4.10)	0.78 (0.27)	12.40 (8.00)	8.71 (5.49)	19.76 (12.21)
	Adapter	Adapter	31.24 (14.98)	54.17 (40.25)	**1.27 (0.76)**	54.72 (40.50)	32.90 (17.04)	32.13 (9.71)	2.72 (0.18)	10.66 (5.30)	8.98 (3.85)	**25.42 (14.73)**
		Soft Prompt	**32.53 (19.94)**	51.63 (38.91)	0.89 (0.25)	51.78 (37.31)	**33.57 (18.08)**	**34.00 (13.34)**	1.96 (0.09)	10.37 (4.80)	8.78 (3.91)	25.06 (15.18)
	Soft Prompt	Adapter	22.06 (12.33)	55.24 (42.44)	0.91 (1.32)	54.38 (41.60)	24.61 (14.26)	22.75 (7.53)	2.63 (0.27)	15.54 (9.50)	12.22 (7.80)	23.37 (15.23)
		Soft Prompt	32.13 (18.00)	53.90 (40.00)	0.99 (0.25)	53.37 (39.83)	29.68 (14.91)	31.42 (10.26)	2.06 (0.18)	**16.86 (9.20)**	11.63 (6.04)	25.78 (15.41)
zh	None	Adapter	22.06 (16.33)	50.83 (40.08)	0.65 (0.50)	50.83 (40.08)	21.96 (14.96)	18.38 (6.74)	1.42 (0.27)	13.29 (9.20)	9.66 (6.31)	21.01 (14.94)
		Soft Prompt	26.25 (20.33)	**56.57 (47.39)**	0.65 (0.44)	**56.57 (47.39)**	25.64 (18.29)	22.32 (8.53)	0.91 (0.45)	16.00 (11.80)	12.49 (9.17)	24.16 (18.20)
	Adapter	Adapter	26.10 (16.29)	43.57 (30.34)	**1.30 (0.57)**	41.48 (28.57)	26.16 (14.21)	22.50 (6.17)	1.94 (0.36)	11.77 (6.50)	9.04 (5.01)	20.43 (12.00)
		Soft Prompt	24.83 (14.06)	47.06 (35.80)	1.08 (0.06)	39.35 (25.55)	25.58 (12.45)	24.19 (5.44)	**2.64 (0.82)**	10.34 (4.90)	8.90 (4.24)	20.44 (11.48)
	Soft Prompt	Adapter	22.21 (16.90)	52.01 (41.01)	0.53 (0.44)	51.45 (40.25)	23.85 (16.20)	18.82 (6.95)	2.11 (0.45)	13.91 (9.10)	9.59 (6.26)	21.61 (15.28)
		Soft Prompt	**31.34 (24.20)**	56.51 (47.90)	1.23 (0.82)	56.39 (47.98)	**29.25 (21.17)**	**25.06 (9.42)**	1.85 (0.45)	**17.52 (12.20)**	**12.80 (9.48)**	**25.77 (19.29)**

Table 4: Results for the question answering task for cross-lingual transfer from high-resource to mid- and low-resource languages. The results are reported as *F1-Score (Exact Match)*. For each source-target language pair, the best-performing result is highlighted in **bold**, while the second-best scores are underlined. Additionally, language pairs with improved performance compared to inference-only (without incorporating any language or task representation) are marked in green, and those with decreased performance are marked in red.

Task Language	**Language Representation**	**Task Representation**	**bg**	**cs**	**el**	**ml**	**ro**	**sl**	**sk**	**sw**	**te**	**ur**	**Average**
	None	None	0	0	0	0	0	0	0	0	0	0	0
ar	None	Adapter	44.54	60.54	44.29	31.32	45.73	55.63	60.30	49.92	48.53	32.10	47.29
		Soft Prompt	38.11	50.72	39.73	38.32	36.44	48.28	48.69	46.90	45.53	20.61	41.33
	Adapter	Adapter	63.91	**63.98**	**63.89**	**49.19**	53.44	24.02	62.98	49.38	18.05	27.71	47.66
		Soft Prompt	**64.09**	40.09	52.23	48.21	**55.95**	44.67	54.13	51.37	26.65	25.77	46.32
	Soft Prompt	Adapter	53.66	62.72	52.68	41.95	54.82	**60.70**	**64.62**	**55.43**	**51.43**	**43.36**	**54.14**
		Soft Prompt	48.84	40.30	46.02	42.93	33.25	37.20	44.48	30.41	46.01	25.18	39.46
de	None	Adapter	30.63	66.61	53.35	40.49	**62.39**	56.11	67.96	60.19	48.56	34.37	52.07
		Soft Prompt	31.61	61.04	54.21	44.55	51.52	54.04	62.28	57.26	43.20	24.21	48.39
	Adapter	Adapter	**68.49**	64.70	**68.83**	**46.45**	60.64	49.32	68.42	**60.92**	24.52	28.41	54.07
		Soft Prompt	62.16	54.79	60.52	42.90	54.74	44.46	51.17	57.28	27.96	**47.28**	50.33
	Soft Prompt	Adapter	39.84	**67.12**	55.29	40.43	61.59	**63.28**	**71.04**	56.53	**49.70**	45.00	**54.98**
		Soft Prompt	43.30	62.84	61.20	44.44	52.13	57.21	70.80	54.24	48.67	34.37	52.92
en	None	Adapter	35.49	41.53	49.44	24.71	44.91	55.13	44.15	36.52	32.02	25.96	38.99
		Soft Prompt	29.04	42.31	46.31	27.92	53.48	53.06	45.40	48.41	33.41	39.54	41.89
	Adapter	Adapter	21.42	33.99	39.86	17.16	42.90	23.39	23.18	32.18	13.91	21.95	26.99
		Soft Prompt	**50.16**	40.62	**56.07**	**37.99**	**63.94**	**56.46**	35.97	45.38	19.98	**61.68**	**46.83**
	Soft Prompt	Adapter	44.12	40.65	47.99	24.97	52.36	51.04	**51.62**	48.55	31.23	23.32	41.59
		Soft Prompt	22.90	**44.33**	41.92	25.22	49.67	55.77	48.64	**48.77**	**33.85**	9.87	38.09
es	None	Adapter	35.41	**68.66**	49.50	35.38	**66.54**	66.07	**70.46**	64.65	46.24	46.35	54.93
		Soft Prompt	24.70	63.95	48.38	31.45	62.20	60.00	63.25	62.21	47.81	45.11	50.91
	Adapter	Adapter	63.38	58.05	**67.33**	42.61	59.17	44.54	66.00	56.36	17.84	30.86	50.61
		Soft Prompt	**69.95**	52.71	64.27	**42.75**	59.96	52.05	60.82	58.11	31.27	48.71	54.06
	Soft Prompt	Adapter	54.37	67.45	64.81	40.37	64.62	**70.83**	68.78	**66.53**	**50.91**	**53.78**	**60.25**
		Soft Prompt	44.72	65.55	39.99	39.17	63.38	63.52	66.04	61.86	40.59	47.45	53.23
ru	None	Adapter	72.90	46.64	34.14	25.54	49.66	45.84	48.16	50.02	49.73	37.42	46.01
		Soft Prompt	69.54	53.60	25.83	34.23	51.11	51.04	51.10	48.37	49.36	40.56	47.47
	Adapter	Adapter	71.70	54.14	**72.40**	38.30	59.91	30.15	50.78	48.08	18.41	25.92	46.98
		Soft Prompt	73.07	53.33	65.15	37.04	53.23	58.78	52.96	50.20	21.37	27.01	49.21
	Soft Prompt	Adapter	**77.27**	54.48	54.56	42.01	**60.01**	53.82	56.33	**56.61**	**55.41**	**46.92**	55.74
		Soft Prompt	71.73	**59.97**	65.74	**44.44**	58.04	**60.23**	**56.40**	54.53	47.95	46.62	**56.57**
zh	None	Adapter	14.91	43.07	8.42	1.21	39.80	41.54	42.05	40.50	11.46	11.33	25.43
		Soft Prompt	12.35	51.86	17.37	7.50	45.41	**53.70**	49.48	46.97	28.33	17.65	33.06
	Adapter	Adapter	29.40	41.60	**39.63**	7.39	37.02	19.26	28.99	21.72	5.56	6.54	23.71
		Soft Prompt	**37.86**	46.02	19.56	**43.69**	30.45	34.32	18.60	**55.69**	**38.07**	**33.44**	35.77
	Soft Prompt	Adapter	28.41	46.21	24.62	8.49	**50.18**	44.69	47.55	51.13	28.76	15.06	34.51
		Soft Prompt	23.34	**55.43**	32.02	12.41	43.96	53.55	**55.26**	48.51	32.11	18.78	**37.54**

Table 5: Results for the named-entity recognition task using F1-Score. The best scores are **boldfaced**, and the second best are underlined.

Task Language	Language Representation	Task Representation	bg	cs	el	ml	ro	sl	sk	sw	te	ur	Average
	None	None	43.35	35.50	40.88	40.62	4.98	68.74	36.42	38.90	39.58	37.64	38.66
ar	None	Adapter	**74.77**	35.50	74.05	**68.98**	66.06	28.76	37.08	**64.55**	**66.67**	**66.01**	**58.24**
		Soft Prompt	69.42	35.92	69.94	65.47	**77.59**	25.45	35.83	59.94	64.23	60.68	56.45
	Adapter	Adapter	72.16	35.67	**74.17**	65.59	77.15	2.91	37.17	62.18	55.93	47.17	53.01
		Soft Prompt	56.39	**38.08**	62.69	57.98	42.24	**67.54**	37.17	58.16	46.19	57.19	52.36
	Soft Prompt	Adapter	74.21	34.42	72.46	68.70	63.62	32.77	36.17	62.61	66.05	64.05	57.51
		Soft Prompt	65.45	37.50	67.84	62.53	59.03	30.96	**37.25**	61.02	61.44	59.60	54.26
de	None	Adapter	75.03	35.17	**74.51**	69.92	64.61	30.16	35.92	**65.83**	**68.90**	**65.73**	**58.58**
		Soft Prompt	69.84	35.50	70.44	65.67	**79.10**	31.36	37.00	62.75	64.45	62.79	57.89
	Adapter	Adapter	74.41	34.92	72.34	68.74	49.85	16.13	35.50	62.85	66.81	60.18	54.17
		Soft Prompt	62.51	**37.17**	62.81	55.29	35.16	**46.19**	36.67	56.67	46.45	53.13	49.21
	Soft Prompt	Adapter	**75.23**	33.75	71.50	**70.06**	60.25	35.17	34.50	65.81	67.13	64.25	57.77
		Soft Prompt	69.12	36.67	69.06	64.61	70.75	26.35	**37.67**	61.18	64.37	60.00	55.98
en	None	Adapter	75.03	35.00	**74.33**	**70.30**	69.48	28.66	34.58	**67.07**	**69.38**	**64.63**	**58.85**
		Soft Prompt	70.42	33.83	70.32	64.53	77.00	29.96	34.58	63.31	63.43	60.22	56.76
	Adapter	Adapter	64.07	35.33	64.79	56.51	**90.09**	1.00	35.08	61.30	44.11	39.78	49.21
		Soft Prompt	55.19	**38.00**	51.30	46.53	81.35	33.77	**38.75**	56.97	33.33	52.81	48.80
	Soft Prompt	Adapter	**75.11**	33.83	72.50	69.50	54.05	**35.07**	35.58	64.67	67.88	64.07	57.23
		Soft Prompt	65.37	35.58	69.50	64.51	35.64	34.07	37.08	60.94	63.35	59.62	52.57
es	None	Adapter	**74.87**	35.08	**75.29**	**70.40**	72.12	28.06	35.17	**66.17**	**70.00**	**66.09**	**59.33**
		Soft Prompt	68.98	34.83	70.12	65.55	79.83	31.76	35.75	60.66	64.11	61.36	57.30
	Adapter	Adapter	73.95	34.92	74.61	68.26	71.78	10.02	35.83	64.77	59.40	50.42	54.40
		Soft Prompt	54.97	**38.75**	44.47	41.48	**81.98**	8.42	**36.42**	55.11	33.49	46.13	44.12
	Soft Prompt	Adapter	74.27	33.92	72.38	69.68	67.58	**32.67**	36.25	64.35	67.33	62.95	58.14
		Soft Prompt	66.15	35.92	67.54	62.77	78.52	28.76	36.33	59.66	63.19	57.66	55.65
ru	None	Adapter	75.55	34.92	**74.81**	**70.56**	75.24	**29.96**	34.67	**67.54**	**69.54**	**66.29**	**59.91**
		Soft Prompt	69.90	36.25	69.78	64.85	82.62	28.16	35.67	61.60	63.73	62.36	57.49
	Adapter	Adapter	72.63	35.75	74.19	61.14	83.50	4.91	36.25	65.77	51.16	46.19	53.15
		Soft Prompt	63.45	**37.58**	53.43	44.51	**87.01**	11.32	**38.08**	54.67	37.60	53.13	48.08
	Soft Prompt	Adapter	**76.03**	34.92	73.75	68.18	75.68	28.06	35.83	65.77	68.24	63.99	59.05
		Soft Prompt	68.42	37.25	67.72	63.53	75.05	23.65	37.25	57.72	62.55	60.08	55.32
zh	None	Adapter	**74.95**	**35.75**	**73.79**	**69.16**	74.32	28.66	36.33	**68.00**	**69.38**	**66.11**	**59.65**
		Soft Prompt	70.80	35.75	70.44	66.59	78.52	21.84	35.17	62.36	65.75	62.85	57.01
	Adapter	Adapter	61.72	34.08	64.27	50.90	**89.01**	0.20	35.17	60.66	35.71	38.98	47.07
		Soft Prompt	43.47	33.33	56.09	56.29	45.90	**98.20**	35.00	47.07	34.57	47.76	49.77
	Soft Prompt	Adapter	73.43	35.42	72.16	68.06	65.77	31.06	35.50	64.37	66.91	63.67	57.64
		Soft Prompt	68.68	34.67	68.86	64.69	76.95	23.55	**36.50**	60.42	63.39	61.78	55.95

Table 6: For NLI, we report accuracy as a metric. The best results for each language pair are highlighted in **bold** and the second best are underlined. Additionally, language pairs with improved performance compared to inference-only are marked in green, and those with decreased performance are marked in red.

Task Language	Language Representation	Task Representation	bg	cs	el	ml	ro	sl	sk	sw	te	ur	Average
	None	None	0	0	0	0	0	0	0	0	0	0	0
ar	None	Adapter	84.78	**70.74**	86.77	86.45	86.06	82.74	**72.52**	**86.59**	84.32	81.50	**82.25**
		Soft Prompt	86.48	63.08	84.90	86.32	87.61	80.77	56.91	81.47	**84.47**	**81.92**	79.39
	Adapter	Adapter	58.38	45.01	74.35	83.39	73.17	71.29	36.69	82.63	77.91	80.18	68.30
		Soft Prompt	**89.22**	56.49	85.76	**86.82**	78.56	82.24	48.71	85.96	82.44	81.62	77.78
	Soft Prompt	Adapter	85.83	67.14	**87.29**	82.13	**88.41**	**85.08**	68.66	84.36	82.47	78.36	80.97
		Soft Prompt	85.89	61.41	81.91	85.13	86.84	84.60	57.40	84.20	81.95	81.71	79.10
de	None	Adapter	**97.63**	69.72	**97.74**	94.66	**98.11**	**97.11**	73.70	**97.25**	**93.73**	**93.17**	**91.28**
		Soft Prompt	94.49	65.32	96.42	**96.28**	92.33	88.16	62.31	90.69	91.54	89.10	86.66
	Adapter	Adapter	97.03	**78.84**	95.30	85.30	97.19	95.28	82.91	90.39	79.96	85.29	88.75
		Soft Prompt	93.15	58.68	94.46	95.17	95.57	96.11	67.99	87.70	93.06	65.98	84.79
	Soft Prompt	Adapter	97.06	74.94	96.84	92.17	97.76	96.85	**82.96**	95.40	89.64	86.71	91.03
		Soft Prompt	94.51	63.87	91.18	94.29	95.44	94.49	57.84	86.67	93.02	76.01	84.73
en	None	Adapter	95.79	**85.10**	97.46	89.15	94.52	89.31	83.09	**91.98**	87.46	88.65	90.25
		Soft Prompt	94.98	55.29	95.09	**93.12**	91.50	91.47	53.03	91.17	**93.50**	88.35	84.75
	Adapter	Adapter	**97.22**	82.95	**98.04**	92.28	**96.98**	**95.80**	**85.98**	91.20	85.99	**89.57**	**91.60**
		Soft Prompt	88.69	40.60	94.31	86.24	94.97	93.09	46.60	88.02	81.91	86.00	80.04
	Soft Prompt	Adapter	87.61	83.47	95.89	77.89	88.38	81.98	85.49	87.42	77.34	81.67	84.71
		Soft Prompt	80.71	75.72	92.71	82.49	90.44	84.21	70.08	82.65	79.55	75.16	81.37
es	None	Adapter	95.66	76.98	95.85	91.93	**97.06**	95.81	74.40	**95.62**	93.15	**90.57**	90.70
		Soft Prompt	92.80	76.38	94.59	91.93	90.62	82.23	74.78	92.40	91.90	83.25	87.09
	Adapter	Adapter	84.35	73.15	95.94	92.67	96.16	94.07	81.65	93.48	**93.80**	88.13	89.34
		Soft Prompt	92.98	77.88	96.15	93.27	96.03	93.07	81.02	92.09	83.19	71.66	87.73
	Soft Prompt	Adapter	**97.47**	**87.14**	**97.41**	**94.78**	96.89	**96.01**	**86.36**	95.55	91.72	86.85	**93.02**
		Soft Prompt	96.35	79.97	95.79	90.56	95.08	92.59	78.85	92.85	89.11	87.56	89.87
ru	None	Adapter	98.87	72.15	**98.46**	90.34	98.26	96.35	74.45	**97.10**	89.87	**92.93**	**90.88**
		Soft Prompt	98.33	58.26	96.09	95.24	87.02	84.39	59.06	92.15	**95.03**	92.11	85.77
	Adapter	Adapter	97.90	**88.93**	94.70	82.33	96.89	92.79	**88.80**	91.75	77.95	87.34	89.94
		Soft Prompt	97.36	13.87	96.77	**95.50**	95.03	95.70	20.58	95.59	91.23	92.34	79.40
	Soft Prompt	Adapter	98.59	67.29	97.46	88.39	**98.53**	**97.73**	74.44	95.83	88.39	91.36	89.80
		Soft Prompt	**98.89**	48.65	96.71	93.57	97.32	96.85	38.89	95.17	92.26	89.83	84.81
zh	None	Adapter	93.70	74.08	95.07	90.97	93.44	85.53	74.05	90.99	88.47	90.22	87.65
		Soft Prompt	91.44	66.83	92.58	**94.18**	84.92	77.69	66.55	74.85	90.33	81.90	82.13
	Adapter	Adapter	**97.08**	**84.02**	**97.09**	90.20	**97.27**	**95.58**	**84.14**	90.62	90.89	**90.58**	**91.75**
		Soft Prompt	66.88	69.81	65.43	67.65	67.11	66.28	67.57	65.58	65.92	65.92	66.82
	Soft Prompt	Adapter	96.07	76.96	93.18	89.41	95.15	94.37	80.19	**91.13**	91.22	82.55	89.02
		Soft Prompt	95.30	73.52	89.10	94.01	92.61	90.44	62.60	83.33	**91.33**	85.38	85.76

Table 7: Results for the check-worthy claim detection task for cross-lingual transfer. Results are reported using F1-Score, with the best scores in **bold** and the second best underlined.

Language Representation	Task Representation	bg	cs	el	ml	ro	sl	sk	sw	te	ur
None	Adapter	38.46 ± 9.60	66.00 ± 0.77	55.88 ± 5.33	40.99 ± 0.67	60.97 ± 1.74	56.29 ± 0.62	67.55 ± 1.22	58.60 ± 1.97	47.69 ± 2.35	29.15 ± 8.78
	Soft Prompt	28.98 ± 3.24	61.48 ± 0.69	50.64 ± 6.63	43.08 ± 1.81	52.65 ± 1.58	54.03 ± 0.98	62.42 ± 0.75	56.49 ± 2.86	43.94 ± 1.64	26.67 ± 3.05
Adapter	Adapter	**66.85** ± 2.03	64.78 ± 0.15	**69.13** ± 1.48	**47.34** ± 1.86	**64.07** ± 4.97	49.29 ± 2.42	68.15 ± 1.19	**59.45** ± 2.28	25.96 ± 2.86	20.20 ± 11.17
	Soft Prompt	63.93 ± 3.01	50.76 ± 14.17	62.05 ± 3.25	43.61 ± 2.42	56.11 ± 3.25	50.30 ± 7.88	53.63 ± 4.44	55.27 ± 3.95	26.83 ± 4.04	46.63 ± 1.51
Soft Prompt	Adapter	42.19 ± 2.93	**67.25** ± 0.23	60.19 ± 6.02	43.67 ± 4.32	61.89 ± 1.73	**61.75** ± 1.90	**72.95** ± 2.35	56.29 ± 0.65	**52.58** ± 5.43	**47.86** ± 3.77
	Soft Prompt	50.66 ± 12.65	64.00 ± 2.18	63.38 ± 3.85	46.37 ± 2.46	55.90 ± 4.95	57.77 ± 0.71	69.87 ± 2.21	54.26 ± 1.31	48.35 ± 2.19	37.46 ± 3.93

Table 8: Results of cross-lingual transfer from German to six languages for the NER task. We report the mean of three runs along with the standard deviation. The best results are **bolded** and the second best results are underlined.