Can Cross-Lingual Transferability of Multilingual Transformers Be Activated Without End-Task Data?

Zewen Chi[1], **Heyan Huang**[12]*, **Xian-Ling Mao**[1]
[1]School of Computer Science and Technology, Beijing Institute of Technology
[2]Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications
{czw,hhy63,maoxl}@bit.edu.cn

Abstract

Pretrained multilingual Transformers have achieved great success in cross-lingual transfer learning. Current methods typically activate the cross-lingual transferability of multilingual Transformers by fine-tuning them on end-task data. However, the methods cannot perform cross-lingual transfer when end-task data are unavailable. In this work, we explore whether the cross-lingual transferability can be activated without end-task data. We propose a cross-lingual transfer method, named PLUGIN-X. PLUGIN-X disassembles monolingual and multilingual Transformers into submodules, and reassembles them to be the multilingual end-task model. After representation adaptation, PLUGIN-X finally performs cross-lingual transfer in a plug-and-play style. Experimental results show that PLUGIN-X successfully activates the cross-lingual transferability of multilingual Transformers without accessing end-task data. Moreover, we analyze how the cross-model representation alignment affects the cross-lingual transferability.

1 Introduction

Annotated data is crucial for learning natural language processing (NLP) models, but they are mostly only available in high-resource languages, typically in English, making NLP applications hard to access in other languages. This motivates the studies on cross-lingual transfer, which aims to transfer knowledge from a source language to other languages. Cross-lingual transfer has greatly pushed the state of the art on NLP tasks in a wide range of languages (Conneau et al., 2020; Chi et al., 2021; Xue et al., 2021).

Advances in cross-lingual transfer can be substantially attributed to the cross-lingual transferability discovered in pretrained. multilingual Transformers (Devlin et al., 2019; Conneau and Lample, 2019). Pretrained on large-scale multilingual text data, the multilingual Transformers perform cross-lingual transfer surprisingly well on a wide range of tasks by simply fine-tuning them (Wu and Dredze, 2019; K et al., 2020; Hu et al., 2020). Based on this finding, follow-up studies further improve the transfer performance in two aspects, by (1) designing pretraining tasks and pretraining multilingual models with better cross-lingual transferability (Wei et al., 2021; Chi et al., 2021), or (2) developing fine-tuning methods with reduced cross-lingual representation discrepancy (Zheng et al., 2021; Yang et al., 2022).

Current methods typically activate the transferability of multilingual Transformers by fine-tuning them on end-task data. However, they cannot perform cross-lingual transfer when end-task data are unavailable. It is common that some publicly available models are trained with non-public in-house data. In this situation, one can access an already-trained end-task model but cannot access the in-house end-task data due to privacy policies or other legal issues. As a consequence, current methods cannot perform cross-lingual transfer for such models because of the lack of end-task data.

In this work, we study the research question: **whether the cross-lingual transferability of multilingual Transformers can be activated without end-task data?** We focus on the situation that we can access an already-trained monolingual end-task model but cannot access the in-house end-task data, and we would like to perform cross-lingual transfer for the model. To achieve this, we propose a cross-lingual transfer method named PLUGIN-X. PLUGIN-X disassembles the monolingual end-task model and multilingual models, and reassembles them into the multilingual end-task model. With cross-model representation adaptation, PLUGIN-X finally performs cross-lingual transfer in a plug-and-play style.

To answer the research question, we conduct experiments on the cross-lingual transfer on the natu-

*Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2023, pages 12572–12584

ral language inference and the extractive question answering tasks. In the experiments, the multilingual model only sees unlabeled raw English text, so the performance of the reassembled model indicates whether the cross-lingual transferability is activated. Experimental results show that PLUGIN-X successfully transfers the already-trained monolingual end-task models to other languages. Moreover, we analyze how the cross-model representation alignment affects the cross-lingual transferability of multilingual Transformers, and discuss the benefits of our work.

Our contributions are summarized as follows:

- We investigate whether the cross-lingual transferability of multilingual Transformers can be activated without end-task data.
- We propose PLUGIN-X, which transfers already-trained monolingual end-task models to other languages without end-task data.
- Experimental results demonstrate PLUGIN-X successfully activates the transferability.

2 Related Work

Cross-lingual transfer aims to transfer knowledge from a source language to target languages. Early work on cross-lingual transfer focuses on learning cross-lingual word embeddings (CLWE; Mikolov et al. 2013) with shared task modules upon the embeddings, which has been applied to document classification (Schwenk and Li, 2018), sequence labeling (Xie et al., 2018), dialogue systems (Schuster et al., 2019), etc. Follow-up studies design algorithms to better align the word embedding spaces (Xing et al., 2015; Grave et al., 2019) or relax the bilingual supervision of lexicons and parallel sentences (Lample et al., 2018; Artetxe et al., 2018). Later studies introduce sentence-level alignment objectives and obtain better results (Conneau et al., 2018).

Most recently, fine-tuning pretrained language models (PLM; Devlin et al. 2019; Conneau and Lample 2019; Conneau et al. 2020) have become the mainstream approach to cross-lingual transfer. Benefiting from large-scale pretraining, pretrained multilingual language models are shown to be of cross-lingual transferability without explicit constraints (Wu and Dredze, 2019; K et al., 2020). Based on this finding, much effort has been made to improve transferability via (1) pretraining new multilingual language models (Wei et al., 2021; Chi et al., 2021; Luo et al., 2020; Ouyang et al., 2020), or (2) introducing extra supervision such as translated data to the fine-tuning procedure (Fang et al., 2021; Zheng et al., 2021; Yang et al., 2022). PLM-based methods have pushed the state of the art of the cross-lingual transfer on a wide range of tasks (Goyal et al., 2021; Chi et al., 2022; Xue et al., 2021).

3 Methods

In this section, we first describe the problem definition. Then, we present how PLUGIN-X performs cross-lingual transfer with model reassembling and representation adaptation.

3.1 Problem Definition

For the common setting of cross-lingual transfer, the resulting multilingual end-task model is learned by finetuning pretrained multilingual Transformers:

$$\boldsymbol{\theta}_t^{\mathrm{x}} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_t(\mathcal{D}_t^{\mathrm{en}}, \boldsymbol{\theta}), \tag{1}$$

where $\mathcal{D}_t^{\mathrm{en}}$ and $\mathcal{L}_t$ stand for the end-task training data in the source language and the loss function for learning the task t, respectively. The initial parameters of the end-task model are from a pretrained multilingual Transformer, i.e., $\boldsymbol{\theta}_0 := \boldsymbol{\theta}^{\mathrm{x}}$.

Differently, we present the public-model-in-house-data setting for cross-lingual transfer, or **PMID**. Specifically, given an already-trained monolingual end-task model, we assume that the model is obtained by finetuning a publicly available pretrained monolingual Transformer but the training data for the end task are non-public in-house data. Under the PMID setting, we can access a monolingual end-task model $\boldsymbol{\omega}_t^{\mathrm{en}}$ and its corresponding pretrained model before finetuning $\boldsymbol{\omega}^{\mathrm{en}}$. The goal of cross-lingual transfer can be written as

$$\boldsymbol{\omega}_t^{\mathrm{x}} = \arg\min_{\boldsymbol{\omega}} \mathcal{L}(\boldsymbol{\omega}_t^{\mathrm{en}}, \boldsymbol{\theta}^{\mathrm{x}}, \mathcal{D}_{\mathrm{u}}^{\mathrm{en}}, \boldsymbol{\omega}), \tag{2}$$

where using the easily-accessible unlabeled text data $\mathcal{D}_{\mathrm{u}}^{\mathrm{en}}$ is allowed. In what follows, we describe how PLUGIN-X performs cross-lingual transfer under the PMID setting.

3.2 Model Reassembling

Figure 1 illustrates the procedure of model reassembling by PLUGIN-X. PLUGIN-X disassembles monolingual and multilingual models and reassembles them into a new multilingual end-task model.

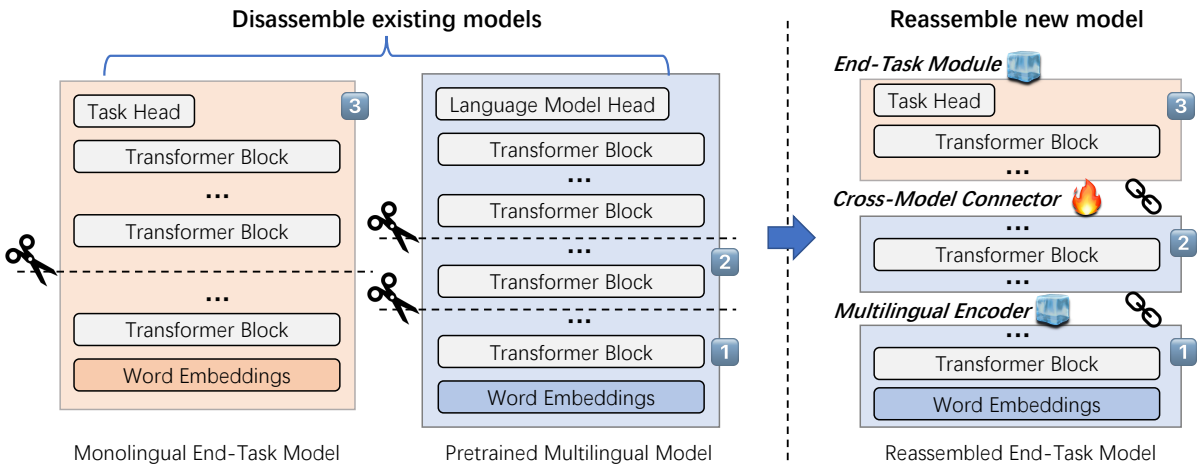


Figure 1: Model reassembling by PLUGIN-X. PLUGIN-X disassembles monolingual and multilingual models and then reassembles them into a new multilingual end-task model. The resulting model consists of three modules, namely multilingual encoder, cross-model connector, and end-task module.

The resulting model consists of three modules, multilingual encoder, cross-model connector, and end-task module. The multilingual encoder and cross-model connector are assembled as a pipeline, which is then plugged into the end-task module.

Multilingual encoder To enable the monolingual end-task model to work with other languages, we use a pretrained multilingual language model as a new encoder. Inspired by the 'universal layer' (Chi et al., 2021) phenomenon, we divide the pretrained model into two sub-modules at a middle layer and keep the lower module as the encoder, because it produces the representations that are better aligned across languages (Jalili Sabet et al., 2020).

Cross-model connector Although the multilingual encoder provides language-invariant representations, the representations can not be directly used by the monolingual end-task model as they are unseen by the end-task model before. Thus, we introduce a cross-model connector, which aims to map the multilingual representations to the representation space of the monolingual end-task model. We simply employ a stack of Transformer (Vaswani et al., 2017) layers as the connector, because: (1) pretrained contextualized representations have more complex spatial structures, so simple linear mapping is not applicable; (2) using the Transformer structure enables us to leverage the knowledge from the remaining pretrained parameters that are discarded by the multilingual encoder.

End-task module We plug the aforementioned two modules into a middle layer of the end-task model. The bottom layers are discarded and the remaining top layers work as the end-task module. Under the PMID setting, the end-task model is a white-box model, which means we can obtain its inner states and manipulate its compute graph.

We reassemble the above three sub-modules as a pipeline. Formally, let $f_{\mathrm{x}}(\cdot;\boldsymbol{\theta}^{\mathrm{x}})$, $f_{\mathrm{c}}(\cdot;\boldsymbol{\omega}_{\mathrm{c}})$ and $f_t(\cdot;\boldsymbol{\omega}_t^{\mathrm{en}})$ denote the forward function of the multilingual encoder, cross-model connector, and end-task module, respectively. The whole parameter set of the reassembled model is $\boldsymbol{\omega}_t^{\mathrm{x}} = \{\boldsymbol{\theta}^{\mathrm{x}}, \boldsymbol{\omega}_{\mathrm{c}}, \boldsymbol{\omega}_t^{\mathrm{en}}\}$. Given an input sentence x, the output $\hat{y}$ of our model is computed as

$$\hat{y} \sim p(y|x;\boldsymbol{\omega}_t^{\mathrm{x}}) = f_t \circ f_{\mathrm{c}} \circ f_{\mathrm{x}}(x;\boldsymbol{\omega}_t^{\mathrm{x}}). \quad (3)$$

3.3 Representation Adaptation

PLUGIN-X activates the cross-lingual transferability by cross-model representation adaptation. It adapts the representation of the multilingual encoder to the representation space of the monolingual end-task module, by tuning the cross-model connector. We employ masked language modeling (MLM; Devlin et al. 2019) as the training objective, which ensures that the training does not require the in-house end-task data but only unlabeled text data. To predict the masked tokens, we use the original pretrained model of $\boldsymbol{\omega}_t^{\mathrm{en}}$ as the end-task module, denoted by $\boldsymbol{\omega}^{\mathrm{en}}$.

However, it is infeasible to directly apply MLM because the reassembled uses two different vocabularies for input and output. Therefore, we propose *heterogeneous masked language modeling* (HMLM) with different input and output vocabularies. As shown in Figure 2, we generate training examples with the following procedure. First, give

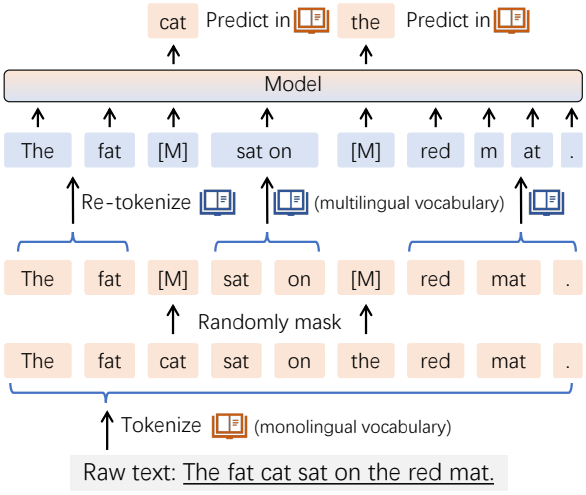


Figure 2: Heterogeneous masked language modeling with different input and output vocabularies for representation adaptation.

an input sentence x, we tokenize x into subword tokens with the vocabulary of the monolingual model $\boldsymbol{\omega}^{\text{en}}$. Then, we randomly select masked tokens as the labels. Next, we re-tokenize the text spans separated by mask tokens using the vocabulary of the multilingual encoder. Finally, the re-tokenized spans and the mask tokens are concatenated into a whole sequence as the input, denoted by $\tilde{x}$. The final loss function is defined as

$$\mathcal{L}_{\text{PlugIn-X}} = -\sum_{i \in \mathcal{M}} \log p(x_i | \tilde{x}, i; \boldsymbol{\omega}_{\text{c}}), \quad (4)$$

where p stands for the predicted distribution over the multilingual vocabulary, and $\mathcal{M}$ is the set of mask positions. Notice that only the connector $\boldsymbol{\omega}_{\text{c}}$ is updated during training, and the other two modules are frozen.

3.4 Plug-and-Play Transfer

Figure 3 illustrates how the resulting reassembled model performs cross-lingual transfer in a plug-and-play manner. After the aforementioned cross-model representation adaptation procedure, we remove the current end-task module $\boldsymbol{\omega}^{\text{en}}$ on the top, which is for the HMLM task. Then, we plug the remaining part of the model into the end-task module $\boldsymbol{\omega}_t^{\text{en}}$, and now the model can directly perform the end-task t in target languages.

4 Experiments

4.1 Setup

Data We perform PLUGIN-X representation adaptation training on the unlabeled English text data from the CCNet (Wenzek et al., 2019) corpus, which provides massive unlabeled text data for a variety of languages crawled from webpages.

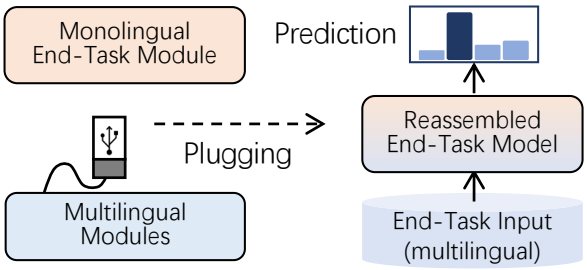


Figure 3: Illustration of how the reassembled model performs cross-lingual transfer in a plug-and-play manner.

Model PLUGIN-X utilizes Transformer (Vaswani et al., 2017) as the backbone architecture of the models. We build two models, named PLUGIN-$\text{X}_{\text{XLM-R}}$ and PLUGIN-$\text{X}_{\text{InfoXLM}}$, where the multilingual encoders and cross-model connectors are from the pretrained multilingual Transformers of base-size XLM-R (Conneau et al., 2020) and InfoXLM (Chi et al., 2021), respectively. The embedding layer and the bottom six layers are assigned to the multilingual encoder, while the other six Transformer layers are assigned to initialize the cross-model connector. The multilingual encoders and cross-model connectors are plugged into the monolingual end-task model at the sixth layer for both representation adaptation and plug-and-play transfer. We use the RoBERTa (Liu et al., 2019) model as the public model for the monolingual model. During representation adaptation, our model is trained on 512-length token sequences with a batch size of 256. We use the Adam (Kingma and Ba, 2015) optimizer for 30K update steps. More training details can be found in Appendix A.

Evaluation We evaluate the reassembled models on two natural language understanding tasks, i.e., natural language inference and extractive question answering. The experiments are conducted under the PMID setting, where the models are not allowed to access end-task data but only an already-trained monolingual task model. On both tasks, we use the finetuned RoBERTa (Liu et al., 2019) models as the monolingual task model to be transferred.

Baselines We implement two cross-lingual transfer baselines that satisfy the PMID setting, and also include the direct finetuning method as a reference.

Model	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	avg
The public-model-in-house-data setting (PMID)															
EMBMAP	33.3	33.3	33.1	33.3	33.6	33.6	33.2	33.4	34.1	33.3	33.3	33.3	33.7	33.6	33.4
EMBLEARN	36.8	36.5	36.2	33.9	34.8	35.5	35.6	34.1	37.4	35.2	35.3	33.4	34.5	34.7	35.3
PLUGIN-$X_{\texttt{XLM-R}}$	66.2	63.4	**65.8**	63.0	65.5	62.4	57.3	58.2	63.7	59.0	60.5	56.5	48.3	52.4	60.2
PLUGIN-$X_{\texttt{InfoXLM}}$	**67.4**	**67.6**	65.6	**64.7**	**66.2**	**65.0**	**60.3**	**61.4**	**66.6**	**63.7**	**64.2**	**59.2**	**55.2**	**53.9**	**62.9**
The cross-lingual transfer setting															
FINETUNE$_{\texttt{XLM-R}}$	79.7	80.7	78.7	77.5	79.6	78.1	74.2	73.8	76.5	74.6	76.7	72.4	66.5	68.3	75.5
FINETUNE$_{\texttt{InfoXLM}}$	80.3	80.9	79.3	77.8	79.3	77.6	75.6	74.2	77.1	74.6	77.0	72.2	67.5	67.3	75.8

Table 1: Evaluation results on XNLI natural language inference under the PMID setting. We report the average results with three random seeds for baselines and PLUGIN-X. Results of FINETUNE are from Chi et al. (2021). Notice that the results are not comparable between the two settings.

(1) EMBMAP learns a linear mapping between the word embedding spaces of the monolingual RoBERTa model and the multilingual InfoXLM (Chi et al., 2021) model. Following Mikolov et al. (2013), the mapping is learned by minimizing L_2 distance. After mapping, we replace the word embeddings of the end-task model with the mapped multilingual embeddings.

(2) EMBLEARN learns multilingual word embeddings for the monolingual end-task model. We replace the vocabulary of RoBERTa with a joint multilingual vocabulary of 14 languages of XNLI target languages. Then, we build a new word embedding layer according to the new multilingual vocabulary. We learn the multilingual word embeddings by training the model on 14-language text from CCNet with 30K training steps and a batch size of 256. Following Liu et al. (2019), the training data is masked language modeling with 512-length text sequences. During training, we freeze all the parameters except the multilingual word embeddings. Finally, we replace the word embeddings of the end-task model with the newly-learned multilingual word embeddings.

(3) FINETUNE directly finetunes the multilingual Transformers for the end tasks, which does not satisfy the PMID setting. We include the results as a reference.

Notice that our goal is to investigate whether PLUGIN-X can activate the cross-lingual transferability of multilingual Transformers, rather than achieving state-of-the-art cross-lingual transfer results. Therefore, we do not compare our models with machine translation systems or state-of-the-art cross-lingual transfer methods.

4.2 Natural Language Inference

Natural language inference aims to recognize the textual entailment between the input sentence pairs. We use the XNLI (Conneau et al., 2018) dataset that provides sentence pairs in fifteen languages for validation and test. Given an input sentence pair, models are required to determine whether the input should be labeled as 'entailment', 'neural', or 'contradiction'. For both baselines and PLUGIN-X, we provide the same monolingual NLI task model, which is a RoBERTa model finetuned on MNLI (Williams et al., 2018).

We present the XNLI accuracy scores in Table 1, which provides the average F1 scores over three runs. Overall, PLUGIN-X outperforms the baseline methods on XNLI cross-lingual natural language inference in terms of average accuracy, achieving average accuracy of 60.2 and 62.9. The results demonstrate that PLUGIN-X successfully activates the cross-lingual transferability of XLM-R and InfoXLM on XNLI without accessing XNLI data. In addition to high-resource languages such as French, our models perform surprisingly well for low-resource languages such as Urdu. Besides, we see that the choice of the multilingual Transformer can affect the cross-lingual transfer results.

4.3 Question Answering

Our method is also evaluated on the extractive question answering task to validate cross-lingual transferability. Given an input passage and a question, the task aims to find a span in the passage that can answer the question. We use the XQuAD (Artetxe et al., 2020) dataset, which provides passages and question-answer pairs in ten languages.

The evaluation results are shown in Table 2, in which we report averaged F1 scores of extracted

Model	es	de	el	ru	tr	ar	vi	th	zh	hi	avg
The public-model-in-house-data setting (PMID)											
EMBMAP	1.1	1.3	1.9	0.4	0.6	0.9	1.4	0.7	1.5	1.6	1.1
EMBLEARN	9.4	4.9	5.6	8.6	8.2	5.7	12.1	4.6	7.6	3.1	7.0
PLUGIN-X$_{\texttt{XLM-R}}$	45.6	40.2	29.1	29.7	22.4	27.6	31.5	21.2	34.1	25.1	30.6
PLUGIN-X$_{\texttt{InfoXLM}}$	**53.3**	**52.4**	**41.2**	**51.4**	**42.4**	**45.1**	**51.6**	**37.3**	**54.7**	**40.9**	**47.0**
The cross-lingual transfer setting											
FINETUNE$_{\texttt{XLM-R}}$	76.4	74.4	73.0	74.3	68.3	66.8	73.7	66.5	51.3	68.2	69.3

Table 2: Evaluation results on XQuAD extractive question answering under the PMID setting. We report the average results with three random seeds for baselines and PLUGIN-X. Results of FINETUNE are from Pfeiffer et al. (2020). Notice that the results are not comparable between the two settings.

Model	XNLI	XQuAD
PLUGIN-X	**53.5**	**35.7**
– Middle-layer plugging	46.7	4.7
– Deeper connector	37.7	16.9
– Multilingual encoder	38.9	9.9

Table 3: Ablation studies on key components of PLUGIN-X.

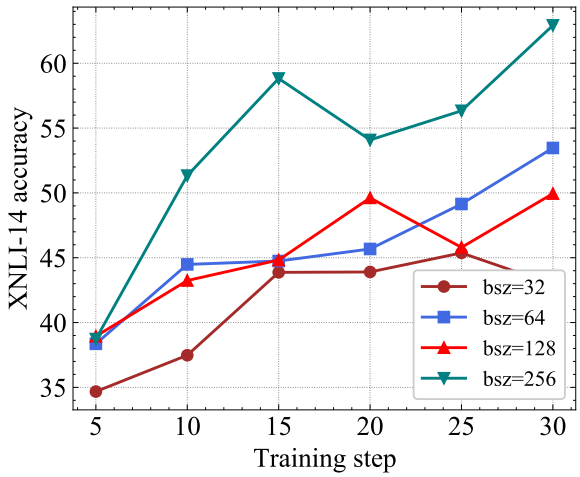


Figure 4: The average XNLI-14 accuracy scores, where we perform PLUGIN-X representation adaptation various batch sizes and training steps.

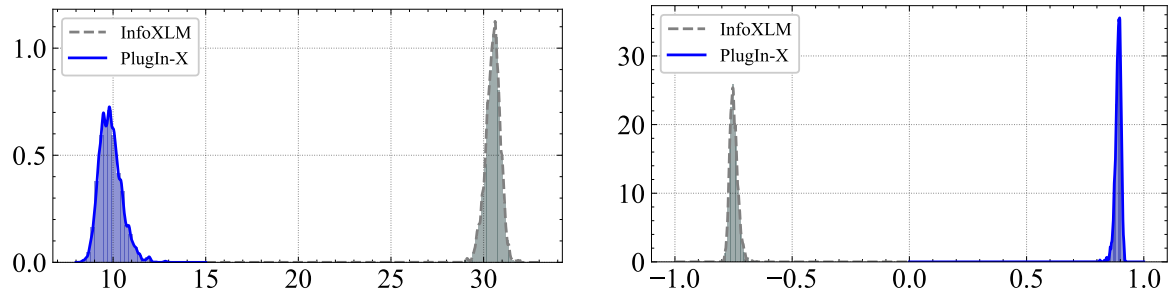
spans from runs with three random seeds. Similar to the results on XNLI, PLUGIN-X obtains the best average F1 score among the baseline methods. The results demonstrate the effectiveness of our model on question answering under the PMID setting, which also indicates PLUGIN-X successfully activates the cross-lingual transferability. Nonetheless, it shows that PLUGIN-X lags behind FINETUNE, showing that PMID is a challenging setting for cross-lingual transfer.

4.4 Ablation Studies

In the ablation studies, we train various models with PLUGIN-X with different architectural or hyper-parameter configurations. Notice that the models are plugged into the same English end-task model for plug-and-play cross-lingual transfer, so the end-task performance can directly indicate the cross-lingual transferability.

Key architectural components We conduct experiments to validate the effects of key architectural components of PLUGIN-X. We train several models with a batch size of 64 for 30K steps. The models are described as follows. (1) The '– Middle-layer plugging' model plugs the connector to the bottom of the monolingual task model and replaces the embedding layer with the output of the connector; (2) the '– Deeper connector' model uses a shallower connector, reducing the number of connector layers from 6 to 2; (3) the '– Multilingual encoder' model discards the frozen multilingual encoder except for the word embeddings, and regards the whole Transformer body as a connector. The models are evaluated on the XNLI and XQuAD under the PMID setting. The evaluation results are presented in Table 3. It can be observed that the model performs less well when removing any of the components. Discarding the frozen multilingual encoder leads to severe performance drops on both tasks, demonstrating the importance of the frozen multilingual encoder. Besides, using a shallower connector produces the worst results on XNLI and '– Middle-layer plugging' performs worst on XQuAD.

Effect of training step and batch size Figure 4 illustrates the XNLI-14 accuracy curves, where we perform PLUGIN-X representation adaptation with various training steps and batch sizes. Consistently, it shows an upward trend as the models are trained



(a) Cross-model L_2 distance distribution

(b) Cross-model cosine similarity distribution

Figure 5: Cross-model representation distance/similarity distribution on XNLI validation sets.

Model	en-zh		en-ur	
	$L_2 \downarrow$	Cosine $\uparrow$	$L_2 \downarrow$	Cosine $\uparrow$
InfoXLM	30.50	-0.75	30.69	-0.75
PLUGIN-X	**9.86**	**0.89**	**10.20**	**0.88**

Table 4: Quantitative analysis on cross-model representation alignment between the monolingual and multilingual models. We measure the L_2 distance and cosine similarity between the monolingual and multilingual models.

with more steps, indicating that the representation adaptation leads to better activation of cross-lingual transferability. Besides, PLUGIN-X also tends to activate better cross-lingual transferability when using larger batch sizes and obtains the best performance with a batch size of 256.

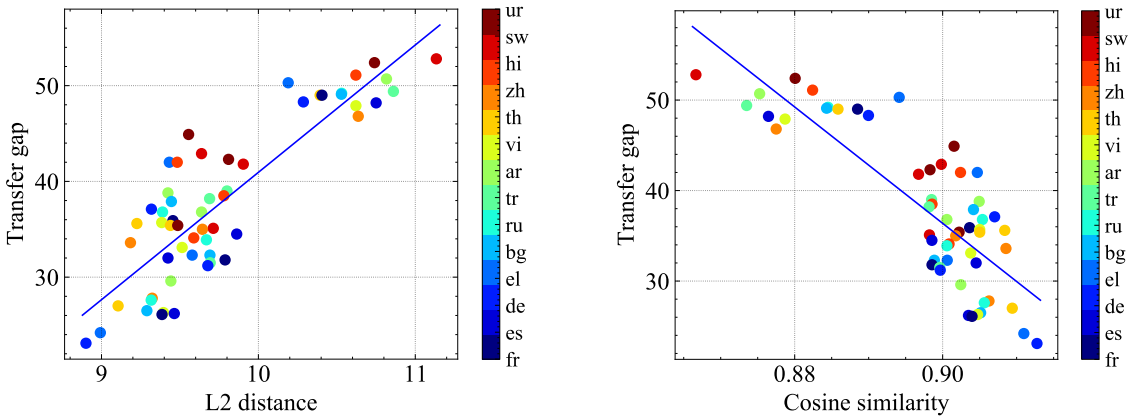
4.5 Analysis

We present analyses on the cross-model representation alignment of the reassembled models, and investigate their cross-lingual transferability.

Cross-model alignment A key factor for our method to achieve cross-lingual transfer under the PMID setting is that PLUGIN-X performs representation adaptation. We conduct experiments to directly provide quantitative analysis on the alignment property of the reassembled models. To this end, we leverage the parallel sentences provided by XNLI as input, and compute their sentence embeddings. Specifically, we first extract the sentence embeddings of the English sentences using the monolingual end-task model, where the embeddings are computed by an average pooling over the hidden vectors from the sixth layer. Then, the sentence embeddings of other languages are obtained from the connector of PLUGIN-X. We also compute the sentence embeddings of other languages using the hidden vectors from the sixth layer of InfoXLM for comparison with our model. Finally, we measure the alignment of the representation spaces by measuring the L_2 distance and cosine similarity between the sentence embeddings output. We compare results between the original InfoXLM model and the reassembled model.

Table 4 and Figure 5 show the quantitative analysis results of representation alignment and the distance/similarity distribution on XNLI validation sets, respectively. Compared with InfoXLM, our reassembled model achieves a notably lower L_2 distance than the monolingual end-task model. Consistently, our model also obtains larger cosine similarity scores with low variance. The results show that, although the InfoXLM provides well-aligned representations across languages, there is a mismatch between its representation space and the space of the monolingual end-task model. On the contrary, PLUGIN-X successfully maps the representation space without accessing the in-house end-task data.

Transferability For a better understanding of how PLUGIN-X activates the cross-lingual transferability, we analyze the relation between transferability and cross-model representation alignment. We use the transfer gap metric (Hu et al., 2020) to measure the cross-lingual transferability. In specific, the transfer gap score is computed by subtracting the XNLI accuracy score in the target language from the score in the source language, which means how much performance is lost after transfer. When computing the transfer gap scores, we use the monolingual end-task model results for the source language, and our reassembled model results for target languages. To measure the representation alignment, we follow the procedure mentioned above, using the metrics of $L2$ distance and



(a) Relation between transfer gap and L_2 distance

(b) Relation between transfer gap and cosine similarity

Figure 6: Relation between cross-lingual transferability and cross-model representation alignment.

cosine similarity. We compute transfer gap, L_2 distance, and cosine similarity scores with the reassembled models from various steps on the validation sets of XNLI in fourteen target languages.

In Figure 6 we plot the results. We see a clear trend that the transfer gap decreases as PLUGIN-X achieves lower cross-model L_2 distance. The trend is also confirmed when we switch the representation alignment metric to the cosine similarity. This highlights the importance of cross-model representation alignment between the monolingual model and the multilingual model for the activation of cross-lingual transferability. More interestingly, the data points have the same trend no matter what language they belong to. Besides, we also observe that the data points of blue colors are high-resource languages, which typically have lower transfer gaps. Our findings indicate that the cross-lingual transfer can be improved by encouraging cross-model alignment.

5 Discussion

Transferability activation To answer our research question, we have conducted experiments on cross-lingual transfer under the public-model-in-house-data (PMID) setting. Our experimental results in Section 4.2 and Section 4.3 show that PLUGIN-X successfully activates the cross-lingual transferability of multilingual Transformers without using the in-house end-task data. Notice that our goal is to answer the research question, rather than develop a state-of-the-art algorithm for the common cross-lingual transfer setting.

Transferability quantification It is difficult to quantify cross-lingual transferability because the results are non-comparable and the compared models typically have different performances in the source language. We propose to transfer an already-trained end-task model to other languages. As the end-task model is stationary, the transfer gap is only dependent on cross-lingual transferability. Therefore, we recommend that the models to be evaluated should transfer the same end-task model to obtain comparable transferability scores.

Model fusion We show that two models with two different capabilities, i.e., end-task ability and multilingual understanding ability, can be fused into a single end-to-end model with a new ability, performing the end task in multiple languages. We hope this finding can inspire research on the fusion of models with different languages, modalities, and capabilities.

6 Conclusion

In this paper, we have investigated whether the cross-lingual transferability of multilingual Transformers can be activated without end-task data. We present a new problem setting of cross-lingual transfer, the public-model-in-house-data (PMID) setting. To achieve cross-lingual transfer under PMID, we propose PLUGIN-X, which reassembles the monolingual end-task model and multilingual models as a multilingual end-task model. Our results show that PLUGIN-X successfully activates the cross-lingual transferability of multilingual Transformers without accessing the in-house end-task data. For future work, we would like to study the research question on more types of models such as large language models (Huang et al., 2023).

Limitations

Our study has limitations in two aspects. First, multilingual Transformers support a wide range of task types, and it is challenging to study our research question on all types of end tasks. We conduct experiments on two common types of end tasks, i.e., text classification and question answering. We leave the study on other types of end tasks in further work. Second, under PMID, we only consider the situation that the end-task models are obtained by finetuning public pretrained models. The cross-lingual transfer of black-box end-task models is also an interesting research topic to study. Besides, PLUGIN-X reassembles the modules from publicly-available models rather than training from scratch, so it can naturally inherit the risks from those models.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No.U21B2009).

References

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798.

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.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online. Association for Computational Linguistics.

Zewen Chi, Shaohan Huang, Li Dong, Shuming Ma, Bo Zheng, Saksham Singhal, Payal Bajaj, Xia Song, Xian-Ling Mao, He-Yan Huang, et al. 2022. Xlm-e: Cross-lingual language model pre-training via electra. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6170–6182.

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 and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7057–7067. Curran Associates, Inc.

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.

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.

Yuwei Fang, Shuohang Wang, Zhe Gan, Siqi Sun, and Jingjing Liu. 2021. Filter: An enhanced fusion method for cross-lingual language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12776–12784.

Naman Goyal, Jingfei Du, Myle Ott, Giri Anantharaman, and Alexis Conneau. 2021. Larger-scale transformers for multilingual masked language modeling. *arXiv preprint arXiv:2105.00572*.

Edouard Grave, Armand Joulin, and Quentin Berthet. 2019. Unsupervised alignment of embeddings with wasserstein procrustes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1880–1890. PMLR.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *arXiv preprint arXiv:2003.11080*.

Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, et al. 2023. Language is not all you need: Aligning perception with language models. *arXiv preprint arXiv:2302.14045*.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. SimAlign: High quality word alignments without parallel training data

using static and contextualized embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1627–1643, Online. Association for Computational Linguistics.

Karthikeyan K, Zihan Wang, Stephen Mayhew, and Dan Roth. 2020. Cross-lingual ability of multilingual bert: An empirical study. In *International Conference on Learning Representations*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, San Diego, CA.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations*.

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

Fuli Luo, Wei Wang, Jiahao Liu, Yijia Liu, Bin Bi, Songfang Huang, Fei Huang, and Luo Si. 2020. VECO: Variable encoder-decoder pre-training for cross-lingual understanding and generation. *arXiv preprint arXiv:2010.16046*.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Xuan Ouyang, Shuohuan Wang, Chao Pang, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie-m: Enhanced multilingual representation by aligning cross-lingual semantics with monolingual corpora. *arXiv preprint arXiv:2012.15674*.

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.

Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2019. Cross-lingual transfer learning for multilingual task oriented dialog. 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 3795–3805.

Holger Schwenk and Xian Li. 2018. A corpus for multilingual document classification in eight languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008. Curran Associates, Inc.

Xiangpeng Wei, Rongxiang Weng, Yue Hu, Luxi Xing, Heng Yu, and Weihua Luo. 2021. On learning universal representations across languages. In *International Conference on Learning Representations*.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzman, Armand Joulin, and Edouard Grave. 2019. CCNet: Extracting high quality monolingual datasets from web crawl data. *arXiv preprint arXiv:1911.00359*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL*, pages 1112–1122, New Orleans, Louisiana.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Jiateng Xie, Zhilin Yang, Graham Neubig, Noah A Smith, and Jaime G Carbonell. 2018. Neural cross-lingual named entity recognition with minimal resources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 369–379.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1006–1011.

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

Huiyun Yang, Huadong Chen, Hao Zhou, and Lei Li. 2022. Enhancing cross-lingual transfer by manifold mixup. In *International Conference on Learning Representations*.

Bo Zheng, Li Dong, Shaohan Huang, Wenhui Wang, Zewen Chi, Saksham Singhal, Wanxiang Che, Ting Liu, Xia Song, and Furu Wei. 2021. Consistency regularization for cross-lingual fine-tuning. 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 3403–3417, Online. Association for Computational Linguistics.

A Additional Experiment Details

We implement PLUGIN-X with the PyTorch[1] library and using pretrained Transformers from the Hugging Face[2] repositories. The data of XNLI and XQuAD are from the XTREME[3] (Hu et al., 2020) repository. The above repositories provide the data, models, and licenses. The representation adaptation is accomplished by learning heterogeneous masked language modeling (HMLM). The whole training process takes about 30 hours on four Nvidia V100 GPU cards. The detailed training hyperparameters are shown in Table 5.

Hyperparameters	Value
Multilingual encoder layers	6
Connector layers	6
End-task module layers	6
Hidden size	768
FFN inner hidden size	3,072
Attention heads	12
Training steps	30K
Batch size	256
Adam ϵ	1e-6
Adam β	(0.9, 0.98)
Learning rate	2e-4
Learning rate schedule	Linear
Warmup steps	3K
Gradient clipping	2.0
Weight decay	0.01
HMLM Input length	512
HMLM Mask ratio	0.15

Table 5: Hyperparameters for training with PLUGIN-X.

[1] pytorch.org
[2] huggingface.co
[3] github.com/google-research/xtreme

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitations (p9)

☑ A2. Did you discuss any potential risks of your work?
Limitations (p9)

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 4

☑ B1. Did you cite the creators of artifacts you used?
Section 4

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Appendix Section A

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Appendix Section A

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Appendix Section A

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Appendix Section A

C ☑ Did you run computational experiments?

Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix Section A

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix Section A

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 4

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix Section A

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.