Enhancing Cross-lingual Natural Language Inference by Prompt-learning from Cross-lingual Templates

Kunxun Qi[1,3], **Hai Wan**[1,3]*, **Jianfeng Du**[2,4]*, **Haolan Chen**[5]
[1] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
[2] Guangzhou Key Laboratory of Multilingual Intelligent Processing,
Guangdong University of Foreign Studies, Guangzhou, China
[3] Key Laboratory of Machine Intelligence and Advanced Computing (Sun Yat-sen University),
Ministry of Education, China [4] Pazhou Lab, Guangzhou, China
[5] Platform and Content Group, Tencent, Shenzhen, China

Abstract

Cross-lingual natural language inference (XNLI) is a fundamental task in cross-lingual natural language understanding. Recently this task is commonly addressed by pre-trained cross-lingual language models. Existing methods usually enhance pre-trained language models with additional data, such as annotated parallel corpora. These additional data, however, are rare in practice, especially for low-resource languages. Inspired by recent promising results achieved by prompt-learning, this paper proposes a novel prompt-learning based framework for enhancing XNLI. It reformulates the XNLI problem to a masked language modeling problem by constructing cloze-style questions through cross-lingual templates. To enforce correspondence between different languages, the framework augments a new question for every question using a sampled template in another language and then introduces a consistency loss to make the answer probability distribution obtained from the new question as similar as possible with the corresponding distribution obtained from the original question. Experimental results on two benchmark datasets demonstrate that XNLI models enhanced by our proposed framework significantly outperform original ones under both the full-shot and few-shot cross-lingual transfer settings.

1 Introduction

Cross-lingual language understanding (XLU) plays a vital role in multilingual systems. It aims at training a model in a source language which is then applied to other languages. Cross-lingual natural language inference (XNLI) is a challenge task for evaluating XLU (Conneau et al., 2018). Natural language inference (NLI) aims to determine the inferential relationship between the text of a premise and the text of a hypothesis while XNLI upgrades NLI to the cross-lingual scenarios.

* Corresponding authors: wanhai@mail.sysu.edu.cn, jfdu@gdufs.edu.cn

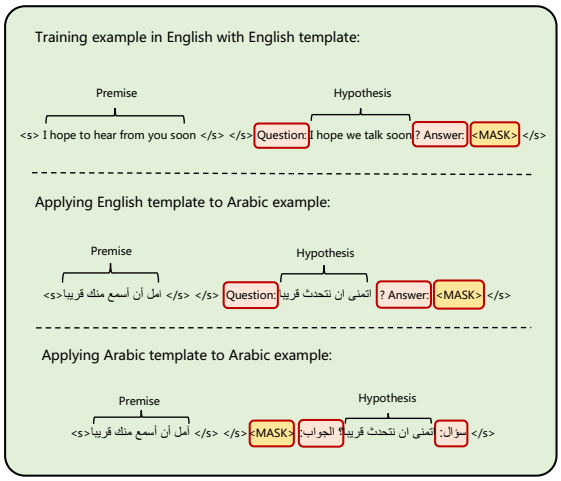


Figure 1: Examples of applying cross-lingual templates.

Nowadays pre-trained cross-lingual language models (Conneau and Lample, 2019; Conneau et al., 2020) have become a dominant paradigm for XLU, significantly improving the performance in various XLU tasks included XNLI. Existing methods (Huang et al., 2019; Chi et al., 2021a,b) usually utilize various auxiliary tasks to improve the cross-lingual transferability of a pre-trained cross-lingual language model, mainly relying on annotated parallel corpora. In practice, these methods can hardly work for low-resource language scenarios where parallel corpora are rare.

Recently, prompt-learning based methods (Schick and Schütze, 2021; Shin et al., 2020) have shown to achieve promising results for few-shot natural language processing (NLP). These methods reformulate the text classification problem into a masked language modeling problem. In particular, the work (Zhao and Schütze, 2021) demonstrates that prompt-learning outperforms fine-tuning in few-shot XNLI. We argue that the effectiveness of prompt-learning in XNLI still needs to be explored by a larger margin. The reasons are two-fold. On one hand, the effectiveness of prompt-learning in XNLI under the full-shot setting is still unknown.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1910 - 1923

On the other hand, the way to make the best of question templates is unexplored yet. The work (Zhao and Schütze, 2021) uses a uniform template in English for all examples in different languages. This way can hardly capture language-specific characteristics in XNLI, especially for those languages that are right-to-left written such as Arabic and Urdu. We naturally expect that language-specific question templates lead to higher performance in XNLI. Figure 1 illustrates how language-specific question templates are used. The second sub-figure shows the uniform question template used in (Zhao and Schütze, 2021) to handle an Arabic example, where the corresponding example in English is shown in the first sub-figure. The last sub-figure shows the Arabic-specific question template used for the same Arabic example, which is right-to-left written and conforms to the Arabic grammar.

In order to introduce language-specific characteristics in question templates while capturing correspondence between different languages, we propose a novel prompt-learning based framework named PCT (shot for *Prompt-learning from Cross-lingual Templates*) for XNLI. As illustrated in Figure 2, PCT first constructs a cloze-style question by filling the template in the source language (namely English), then randomly samples a template in another language (such as Chinese) to construct an augmented question, where the augmented question is written in two languages and thus its template is called a *cross-lingual template*. Both the original question and the augmented question are fed into a pre-trained cross-lingual language model to calculate the answer probability distributions for inferential relationships that are represented by predefined tokens mapped from the mask token. To enforce answer consistency for the two questions, i.e., to make the two probability distributions of inferential relationships as similar as possible, the two probability distributions are regularized by the Kullback-Leibler divergence (KLD) loss. The entire model is trained by minimizing the sum of the cross-entropy loss for classification accuracy and the KLD loss for answer consistency.

We employ PCT to enhance pre-trained cross-lingual language models XLM-R (Conneau et al., 2020) and INFOXLM (Chi et al., 2021a). Experimental results on the XNLI (Conneau et al., 2018) benchmark and the PAWS-X (Yang et al., 2019) benchmark show that PCT improves the original models by a significant margin under both the full-shot and few-shot cross-lingual transfer settings.

Main contributions of this work include:

1. We propose a novel prompt-learning based framework for XNLI. In this framework, a data augmentation strategy is introduced which relies merely on predefined cross-lingual templates; moreover, a consistency loss is introduced to enforce similar output probability distributions for arbitrary two languages so as to capture correspondence between different languages.

2. We conduct extensive experiments on two large-scale benchmarks to demonstrate significant improvements achieved by the proposed framework, under both the full-shot and few-shot cross-lingual transfer settings.

2 Related Work

Up to date XLU including XNLI are widely addressed by pre-trained cross-lingual language models (Devlin et al., 2019; Conneau and Lample, 2019; Conneau et al., 2020). Multilingual BERT (mBERT) (Devlin et al., 2019) extends the basic pre-trained language model BERT by training with multilingual corpora. XLM (Conneau and Lample, 2019) enhances mBERT by introducing the translation language modeling (TLM) objective. XLM-RoBERTa (XLM-R) (Conneau et al., 2020) trains XLM with larger corpora and more epochs.

Cross-lingual language models can further be enhanced by post-training tasks that rely on large-scale parallel corpora. UNICODER (Huang et al., 2019) introduces several post-training tasks to utilize parallel corpora. INFOXLM (Chi et al., 2021a) enhances XLM-R by introducing the cross-lingual contrastive learning task using 42 GB parallel corpora. XLM-ALIGN (Chi et al., 2021b) introduces a denoising word alignment pre-training task using several parallel corpora. These enhancements can hardly be applied to low-resource languages for which parallel corpora are rare. To alleviate the dependence on parallel corpora, some data augmentation strategies have been proposed for XNLI. TMAN (Qi and Du, 2020) enhances XNLI by exploiting adversarial training from translated data. The work (Dong et al., 2021) proposes a data augmentation strategy for XNLI by generating augmented data from a pre-trained sequence-to-sequence model. UXLA (Bari et al., 2021) improves the performance of XNLI by data augmen-

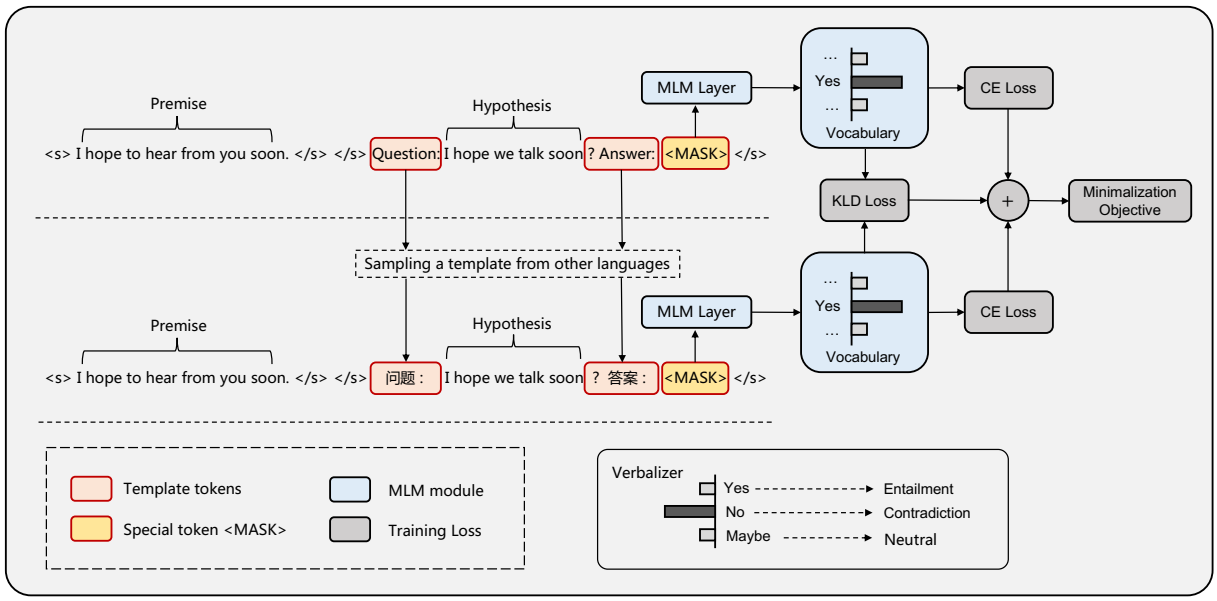


Figure 2: The proposed PCT framework.

tation and unsupervised sample selection. All these strategies require a large amount of external resources for data augmentation. In contrast, our proposal augments data by only predefined cross-lingual templates.

Recently prompt-learning based methods have shown to achieve promising results in various few-shot NLP tasks. The key of these methods is reformulating the text classification problem into a masked language modeling problem by constructing cloze-style questions. The work (Schick and Schütze, 2021) applies prompt-learning to text classification (including NLI) with manually defined templates. The work (Shin et al., 2020) proposes to search for optimal discrete templates by a gradient based approach. Several approaches (Li and Liang, 2021; Liu et al., 2021; Han et al., 2021) have been proposed to search continuous prompts. The work (Zhao and Schütze, 2021) compares prompt-learning with fine-tuning in few-shot XNLI. Different from (Zhao and Schütze, 2021), this work significantly advances prompt-learning in XNLI further by introducing a new data augmentation strategy and a new consistency loss for regularization. The effectiveness of prompt-learning is also demonstrated further under both the full-shot and few-shot cross-lingual transfer settings.

3 The PCT Framework

The proposed PCT framework is illustrated in Figure 2. For every training triple (premise, hypothesis, label) in English, PCT first constructs a cloze-style question by filling the template in English, then samples a predefined template from another language such as Chinese to construct an augmented question. Both the original question and the augmented question are fed into a pre-trained cross-lingual model to calculate the answer distributions of the mask token, through the masked language modeling (MLM) layer in the pre-trained cross-lingual model. The entire model is trained by minimizing the cross-entropy loss for classification accuracy and the Kullback-Leibler divergence (KLD) loss for answer consistency.

3.1 Formalization of PCT

The training phase of PCT is formalized in Algorithm 1. For every training triple (P_i, H_i, Y_i) in English, where $P_i = \{w_j^P\}_{j=1}^m$ denotes the word sequence of the premise, $H_i = \{w_j^H\}_{j=1}^n$ the word sequence of the hypothesis, $Y_i \in \mathcal{Y}$ the index of the NLI label, PCT first constructs a cloze-style question X_i by filling the English template, and then randomly samples a template from other languages to construct an augmented question $\overline{X}_i$. A template in an arbitrary language is a textual string with three unfilled slots: a input slot [P] to fill the input premise, a input slot [H] to fill the input hypothesis and an answer slot [Z] that allows language models to fill label words. [Z] is usually filled by the mask token [MASK] when using pre-trained language models. For instance, the English template is expressed as "<s>[P]</s></s>Question: [H]? Answer: [MASK]</s>", where <s> and </s>

Algorithm 1 The training phase of PCT

Require: the number of epochs E and the training set $\mathbb{D} = \{(P_i, H_i, Y_i)\}_{i=1}^{M}$ in English.

1: Reform $\mathbb{D}$ to a set of tuples $\mathbb{S} = \{X_i, Y_i\}_{i=1}^{M}$ by filling the English template.
2: Extend $\mathbb{S}$ to $\mathbb{T} = \{(X_i, \overline{X}_i, Y_i)\}_{i=1}^{M}$ by filling a randomly sampled template from other languages for each (P_i, H_i).
3: Divide $\mathbb{T}$ into a set of mini-batches $\mathbb{B}$.
4: **for** epoch from 1 to E **do**
5: Shuffle $\mathbb{B}$.
6: **for** each mini-batch $\{(X_i, \overline{X}_i, Y_i)\}_{1 \le i \le N}$ in $\mathbb{B}$ **do**
7: Compute total loss $\mathcal{L}$ by Eq. (5).
8: Update parameters θ by gradient descent.
9: **end for**
10: **end for**

are special tokens in XLM-R to separate sentences. The verbalizer $\mathcal{M} : \mathcal{Y} \to \mathcal{V}$ is a function to map NLI labels to indices of answer words in the given vocabulary. Let l denote the size of the given vocabulary and d the dimension of the contextualized representation of a token, output by a pre-trained cross-lingual language model with an MLM layer, such as XLM-R (Conneau et al., 2020). The answer probability distribution is calculated by:

$$y_i = \text{softmax}(W_{\text{lm}} h_i^{[\text{MASK}]}) \tag{1}$$

where $W_{\text{lm}} \in \mathbb{R}^{l \times d}$ denotes the parameters of the pre-trained MLM layer and $h_i^{[\text{MASK}]} \in \mathbb{R}^d$ denotes the contextualized representation of the [MASK] token of the i^{th} training triple. Compared with the standard fine-tuning method, no extra parameters are required to be initialized, therefore the model can be optimized by fewer samples.

Given a mini-batch $(X_i, \overline{X}_i, Y_i)_{1 \le i \le N}$ of N triples, the two cross-entropy losses for the original question and the augmented question are respectively calculated by:

$$\mathcal{L}_X = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{l} I(j = \mathcal{M}(Y_i)) \log y_{i,j}^{X_i} \tag{2}$$

$$\mathcal{L}_{\overline{X}} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{l} I(j = \mathcal{M}(Y_i)) \log y_{i,j}^{\overline{X_i}} \tag{3}$$

where $y_{i,j}^{X_i}$ (resp. $y_{i,j}^{\overline{X_i}}$) denotes the j^{th} element of $y_i \in \mathbb{R}^l$ for the input X_i (resp. for the input $\overline{X}_i$). $I(C)$ is an indicator function that returns 1 if C is true or 0 otherwise.

We observe that, given the same input premise and hypothesis, the answer probability distribution of the question constructed by a cross-lingual template may evidently deviate from that of the question constructed from the English template. Such a deviation may lead to an increase of errors when applying cross-lingual templates to examples in other languages. Our ablation study in Section 4 confirms this phenomenon. To eliminate the negative effect of this deviation, we propose a consistency loss function to regularize the answer probability distributions. More precisely, we employ the symmetric Kullback-Leibler divergence (KLD) loss to enforce the answer probability distributions $y_i^{X_i}$ and $y_i^{\overline{X_i}}$ to be as similar as possible, which is formally defined bellow.

$$\begin{aligned} \mathcal{L}_{\text{KLD}} &= \frac{1}{N} \sum_{i=1}^{N} (\text{KL}(y_i^{X_i} || y_i^{\overline{X_i}}) + \text{KL}(y_i^{\overline{X_i}} || y_i^{X_i})) \\ &= \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{l} (y_{i,j}^{X_i} \log \frac{y_{i,j}^{X_i}}{y_{i,j}^{\overline{X_i}}} + y_{i,j}^{\overline{X_i}} \log \frac{y_{i,j}^{\overline{X_i}}}{y_{i,j}^{X_i}}) \end{aligned} \tag{4}$$

The entire model is trained by minimizing the total loss $\mathcal{L}$ formally defined as:

$$\mathcal{L} = \mathcal{L}_X + \mathcal{L}_{\overline{X}} + \mathcal{L}_{\text{KLD}} \tag{5}$$

where we simply apply the same weight for the three loss terms.

3.2 Inference with Cross-lingual Templates

Since the English template may not conform to the grammar of other languages such as Arabic and Urdu, PCT uses the cross-lingual template in the target language for predicting test examples in the target language. For instance, every Chinese test example is reformed to a Chinese cloze-style question by filling the Chinese template "<s>[P]</s></s>问题: [H]? 答案: [MASK]</s>", which is obtained from the English template by translating prompt words in English to prompt words in Chinese, where the slots [P] and [H] are filled by the premise and the hypothesis in Chinese, respectively. More generally, all cross-lingual templates are obtained from the English template by translating prompt words in English to prompt words in other languages using Google translator [1], where for Arabic

[1] https://translate.google.com/

and Urdu, the prompt part is written from right to left rather than from left to right as in English and other languages. By considering that the English label words have been fine-tuned to work for different languages during the training phase, we use the same English verbalizer $\mathcal{M}$ for all languages in the inference phase.

4 Experiments

To evaluate the effectiveness of the proposed PCT framework, we applied PCT to enhance several pre-trained cross-lingual language models including XLM-R_{base}, XLM-R_{large} and INFOXLM$_{large}$. We call the enhanced models PCT-X, where X denotes the original pre-trained cross-lingual model.

4.1 Datasets

We conducted experiments on two large-scale benchmarks, namely XNLI and PAWS-X.

XNLI: The XNLI (Conneau et al., 2018) benchmark[2] extends the MultiNLI (Williams et al., 2018) benchmark (in English) to 15 languages through translation and comes with manually annotated development set and test set. For each language, the training set comprises 393K annotated sentence pairs, whereas the development set and the test set comprises 2.5 K and 5K annotated sentence pairs, respectively.

PAWS-X: The PAWS-X (Yang et al., 2019) is a cross-lingual paraphrase identification benchmark[3], which extends the Wikipedia portion of the PAWS (Zhang et al., 2019) dataset to 7 languages through translation. For each language, the training set comprises 49.5K annotated sentence pairs, whereas both the development set and the test set comprise 2K annotated sentence pairs each.

4.2 Implementation Details

We implemented our enhanced models by Tensorflow 2.4.0 and trained all the models with 8 TPUs on the Google Colab platform[4].

PCT-XLM-R_{base} was initialized by the pre-trained XLM-R_{base} model with 12 transformer layers, which outputs 768-dimensional token embeddings. The transformer encoder was built with 12 heads. We applied dropout (Srivastava et al., 2014) to each layer by setting the dropout rate to 0.1. The model was trained by Adam (Kingma and Ba, 2015) with the warmup mechanism (Devlin et al., 2019) and two training epochs, where the initial learning rate was set to 5e-5, the warmup proportion to 10%, and the mini-batch size to 64.

PCT-XLM$_{large}$ and PCT-INFOXLM$_{large}$ were respectively initialized by the pre-trained XLM-R_{large} and INFOXLM$_{large}$ models with 24 transformer layers, both of which output 1024-dimensional token embeddings. The transformer encoder was built with 16 heads. The models were trained by RMSProp (Dauphin et al., 2015) with one training epoch, where the initial learning rate was set to 5e-6, the mini-batch size to 32, and the dropout rate to 0.1. We used RMSProp instead of Adam for these large models since the training memory is limited by the Google Colab platform. For all the above models, the input sentence pairs were truncated to maximum 128 tokens. Code and data about our implementations are available at https://github.com/qikunxun/PCT.

4.3 Compared Models

We compared our models with the following pre-trained cross-lingual language models: (1) multilingual BERT (mBERT; Devlin et al. (2019)) is a BERT model pre-trained on Wikipedia with 102 languages; (2) XLM (Conneau and Lample, 2019) is pre-trained for two tasks (MLM and TLM) on Wikipedia with 100 languages; (3) XLM-R (Conneau et al., 2020) extends XLM with larger corpora (i.e. the CC-100 corpora with 100 languages) and more training epochs; (4) UNICODER (Huang et al., 2019) continues training XLM by introducing several post-training tasks using parallel corpora; (5) INFOXLM (Chi et al., 2021a) enhances XLM-R by introducing the cross-lingual contrastive learning task using 42 GB parallel corpora; (6) XLM-ALIGN (Chi et al., 2021b) enhances XLM-R by introducing the denoising word alignment pre-training task using several parallel corpora; (7) The work (Dong et al., 2021) proposes an adversarial data augmentation strategy for XNLI based-on XLM-R; (8) UXLA (Bari et al., 2021) extends XLM-R with data augmentation and unsupervised sample selection. (9) The work (Zhao and Schütze, 2021) proposes three prompt-learning methods for few-shot XNLI, including DP (direct prompting), SP (soft prompting) and MP (mixed prompting).

[2] http://www.nyu.edu/projects/bowman/xnli/

[3] https://github.com/google-research-datasets/paws

[4] https://colab.research.google.com/

Models	⊕	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	△
Train multilingual model on training data in English (Cross-lingual Transfer)																	
mBERT	N	73.7	70.4	70.7	68.7	69.1	70.4	67.8	66.3	66.8	66.5	64.4	68.3	64.2	61.8	59.3	67.2
XLM	Y	85.0	78.7	78.9	77.8	76.6	77.4	75.3	72.5	73.1	76.1	73.2	76.5	69.6	68.4	67.3	75.1
XLM (w/o TLM)	N	83.2	76.7	77.7	74.0	72.7	74.1	72.7	68.7	68.6	72.9	68.9	72.5	65.6	58.2	62.4	70.7
UNICODER	Y	85.4	79.2	79.8	78.2	77.3	78.5	76.7	73.8	73.9	75.9	71.8	74.7	70.1	67.4	66.3	75.3
XLM-R_{base}	N	84.6	78.2	79.2	77.0	75.9	77.5	75.5	72.9	72.1	74.8	71.6	73.7	69.8	64.7	65.1	74.2
INFOXLM	Y	86.4	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	76.5
XLM-ALIGN	Y	86.7	80.6	81.0	78.8	77.4	78.8	77.4	75.2	73.9	76.9	73.8	77.0	71.9	67.1	66.6	76.2
Dong et al. (2021)	N	80.8	75.8	77.3	74.5	74.9	76.3	74.9	71.4	70.0	74.5	71.6	73.6	68.5	64.8	65.7	73.0
DP-XLM-$R^{\dagger}_{base}$	N	83.9	78.1	78.5	76.1	75.7	77.1	75.3	73.2	71.6	74.7	70.9	73.4	70.2	63.6	65.5	73.9
SP-XLM-$R^{\dagger}_{base}$	N	84.7	78.3	78.8	75.6	75.3	76.3	75.7	73.3	70.3	74.0	70.6	74.1	70.2	62.8	64.9	73.7
MP-XLM-$R^{\dagger}_{base}$	N	84.2	78.4	78.8	76.9	75.3	76.5	75.7	72.7	71.2	75.2	70.8	72.8	70.7	61.5	66.0	73.8
PCT-XLM-R_{base}(this work)	N	84.9	79.4	79.7	77.7	76.6	78.9	76.9	74.0	72.9	76.0	72.0	74.9	71.7	65.9	67.3	75.3
XLM-R_{large}	N	88.9	83.6	84.8	83.1	82.4	83.7	80.7	79.2	79.0	80.4	77.8	79.8	76.8	72.7	73.3	80.4
UXLA	N	-	-	**85.7**	84.2	-	-	-	-	80.5	-	-	-	78.7	74.7	73.4	-
INFOXLM$_{large}$	Y	**89.7**	**84.5**	85.5	84.1	83.4	84.2	81.3	80.9	80.4	80.8	78.9	80.9	77.9	74.8	73.7	81.4
PCT-XLM-R_{large}(this work)	N	88.3	84.2	85.1	83.7	83.1	84.4	81.9	81.2	80.9	80.7	78.8	80.3	78.4	73.6	**75.6**	81.3
PCT-INFOXLM$_{large}$(this work)	Y	88.6	**84.5**	85.4	**84.6**	**83.7**	**84.7**	**82.3**	**81.4**	**81.1**	**81.7**	**79.5**	**81.4**	**79.5**	**75.6**	**75.6**	**82.0**

Table 1: **Comparison results on XNLI under the full-shot cross-lingual transfer setting.** Every value is the test accuracy in percent. ⊕ indicates whether the model uses additional datasets for training, where Y denotes additional datasets being used and N being not. △ is the average accuracy for 15 languages. DP-XLM-$R^{\dagger}_{base}$, SP-XLM-$R^{\dagger}_{base}$ and MP-XLM-$R^{\dagger}_{base}$ respectively denote the reproduced result of discrete prompting, soft prompting and mixed prompting approaches proposed in (Zhao and Schütze, 2021) based on XLM-R_{base}.

Models	en	fr	es	de	ja	ko	zh	△
mBERT	94.0	87.0	87.4	85.7	73.0	69.6	77.0	82.0
XLM	94.0	87.4	88.3	85.9	69.3	64.8	76.5	80.0
XLM-$R^{\dagger}_{base}$	94.1	88.7	87.9	87.5	76.6	75.0	80.4	84.3
PCT-XLM-R_{base}	94.5	89.8	89.1	88.0	77.6	77.3	81.8	85.4
XLM-R_{large}	94.7	90.4	90.1	89.7	78.7	79.0	82.3	86.4
PCT-XLM-R_{large}	**95.6**	**92.2**	**91.2**	**90.5**	**82.2**	**81.9**	**84.2**	**88.3**

Table 2: **Comparison results on PAWS-X under the full-shot setting.** Every value is the test accuracy in percent. △ is the average accuracy for 7 languages. XLM-$R^{\dagger}_{base}$ denotes the reproduced result of XLM-R_{base}.

4.4 Main Results

We conducted experiments on both XNLI and PAWS-X under the *cross-lingual transfer* setting, where models are trained on data in the source language (usually English) and tested on data in the target language. This setting is commonly used to evaluate XNLI models. It can be further divided into two sub-settings: the full-shot setting using the whole training set, and the few-shot setting using a fixed number of training samples. For both XNLI and PAWS-X we evaluated models under the full-shot setting, whereas for XNLI we additionally evaluated models under the few-shot setting.

Table 1 reports the results for comparing PCT-enhanced models with other models on XNLI under the full-shot setting. The results of compared models are taken from (Chi et al., 2021a) and (Liang et al., 2020). PCT-XLM-R_{base} achieves 75.3% accuracy on the XNLI test set averaged by 15 target languages, significantly outperforming its basic model XLM-R_{base} by an absolute gain of 1.1% accuracy on average. The difference between PCT-XLM-R_{base} and XLM-R_{base} in average accuracy is statistically significant with p-value 1.7e-6 by a two-tailed t-test. Meanwhile, PCT-XLM-R_{base} outperforms the three prompt-learning approaches (i.e. DP-XLM-$R^{\dagger}_{base}$, SP-XLM-$R^{\dagger}_{base}$ and MP-XLM-$R^{\dagger}_{base}$) in (Zhao and Schütze, 2021) under the full-shot setting. PCT-XLM-R_{large} achieves 81.3% accuracy on the XNLI test set averaged by 15 target languages, pushing XLM-R_{large} by an absolute gain of 0.9% accuracy on average. The difference between PCT-XLM-R_{large} and XLM-R_{large} in average accuracy is statistically significant with p-value 2.5e-4 by a two-tailed t-test. Furthermore, it can be seen that the average accuracy of PCT-XLM-R_{large} is close to that of the current state-of-the-art model INFOXLM$_{large}$ (i.e. 81.4%), which is trained with additional data. To further verify the effectiveness of PCT, we also applied PCT to INFOXLM$_{large}$, denoted by PCT-INFOXLM$_{large}$. It can be seen that PCT-INFOXLM$_{large}$ achieves 82.0% accuracy on average, pushing INFOXLM$_{large}$ by an absolute gain of 0.6% on average. The difference between PCT-INFOXLM$_{large}$ and INFOXLM$_{large}$ in average accuracy is statistically significant with p-value 7.5e-3 by a two-tailed t-test. These results imply that PCT is able to further improve the cross-lingual transferability of state-of-the-art models.

Table 2 reports the comparison results on PAWS-X under the full-shot setting. The results of compared models are taken from (Hu et al., 2020). Since the work (Hu et al., 2020) has not reported the result of XLM-R_{base}, we produced the result of

Shots	Models	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	△
K=16	FT	34.7	33.8	33.8	34.3	33.5	33.8	34.1	34.1	33.6	34.0	33.1	33.5	33.1	33.7	33.2	33.7
	DP	38.2	36.6	36.9	37.5	37.4	37.1	36.5	35.7	35.1	35.8	37.2	37.9	35.9	33.8	34.9	36.4
	SP	39.5	40.9	39.4	**40.2**	40.4	40.6	40.6	36.3	38.9	38.5	39.5	37.4	36.9	37.1	35.9	38.8
	MP	33.2	34.4	34.5	34.0	32.6	33.0	33.9	34.7	32.5	33.3	33.5	35.7	34.3	33.3	32.7	33.7
	PCT (this work)	**46.5**	**44.3**	**41.5**	36.9	**45.7**	**40.8**	**42.4**	**43.7**	**43.6**	**44.7**	**43.9**	**44.8**	**44.8**	**40.1**	**42.5**	**43.1**
K=32	FT	36.6	36.5	36.0	36.0	36.1	36.3	35.7	35.9	35.8	36.1	35.7	35.7	36.2	35.3	34.8	35.9
	DP	43.7	43.9	42.8	43.5	42.5	43.5	42.5	42.0	41.8	41.9	40.5	39.9	39.3	37.5	39.8	41.7
	SP	44.7	42.3	42.3	42.1	42.3	43.4	43.8	38.8	40.3	42.1	40.0	39.6	38.9	37.5	38.8	41.1
	MP	45.5	44.7	41.2	42.6	42.3	42.2	42.2	41.2	41.0	41.7	40.2	40.9	40.2	36.5	40.5	41.5
	PCT (this work)	**49.6**	**48.8**	**45.5**	**44.4**	**47.4**	**45.4**	**45.5**	**44.3**	**45.7**	**46.7**	**41.6**	**45.6**	**46.7**	**40.3**	**42.9**	**45.4**
K=64	FT	41.7	39.5	40.3	40.1	39.9	39.6	38.3	39.5	40.2	40.9	39.2	39.6	39.5	39.6	39.2	39.8
	DP	48.9	48.0	45.0	48.1	46.9	47.6	44.9	45.7	45.6	47.3	45.7	45.2	41.6	41.0	43.3	45.7
	SP	49.0	46.1	45.8	46.0	43.7	43.8	44.5	41.9	43.5	45.3	44.7	44.2	40.9	40.5	40.1	44.0
	MP	**51.8**	48.3	46.6	48.2	46.8	46.0	44.8	44.8	43.9	48.3	45.0	43.0	40.1	37.8	**44.0**	45.3
	PCT (this work)	51.5	**51.3**	**50.9**	**49.3**	**50.6**	**50.2**	**49.1**	**47.4**	**48.1**	**49.7**	**47.3**	**48.2**	**47.6**	**44.6**	**44.0**	**48.6**
K=128	FT	46.9	46.0	45.8	45.6	44.4	45.5	44.9	43.7	43.5	44.8	43.3	44.8	43.0	41.4	41.8	44.4
	DP	53.7	49.3	48.5	51.0	47.4	50.5	46.9	49.6	46.2	48.9	44.8	49.6	44.8	42.0	44.2	48.0
	SP	49.5	46.4	45.8	45.0	46.3	46.2	45.0	41.9	44.8	45.0	45.6	45.7	43.3	41.2	41.2	44.9
	MP	52.6	50.3	49.7	49.0	49.1	48.0	46.4	48.5	46.5	48.2	48.1	50.5	47.0	42.9	44.0	48.0
	PCT (this work)	**55.0**	**53.3**	**53.8**	**52.8**	**53.4**	**51.9**	**51.7**	**50.9**	**50.4**	**51.7**	**50.0**	**51.2**	**51.5**	**47.0**	**47.9**	**51.5**
K=256	FT	57.8	55.4	55.9	54.4	54.0	54.6	52.9	52.3	52.1	54.2	51.2	52.1	50.7	50.0	48.6	53.1
	DP	60.1	54.4	50.6	55.4	55.1	55.6	51.4	50.8	53.2	55.1	53.4	52.7	46.1	45.3	48.4	52.5
	SP	**60.6**	55.8	54.8	53.0	53.1	56.0	52.5	52.1	52.3	54.5	54.5	54.6	49.4	47.3	48.5	53.3
	MP	60.1	55.3	51.6	50.7	54.6	54.0	53.5	51.3	52.8	52.3	53.4	53.8	49.6	45.3	47.2	52.4
	PCT (this work)	60.3	**58.3**	**58.3**	**56.3**	**57.9**	**56.7**	**55.2**	**54.6**	**54.7**	**57.4**	**55.6**	**55.8**	**54.6**	**51.6**	**52.6**	**56.0**

Table 3: **Comparison results on XNLI under the few-shot setting.** Every value is the test accuracy in percent, taking from the mean performance of 5 runs. FT, DP, SP, MP denote the fine-tuning, discrete prompting, soft prompting and mixed prompting approaches proposed in (Zhao and Schütze, 2021). △ is the average accuracy.

Models	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	△
Train multilingual model on all training data including translated data in other 14 languages (TRANSLATE-TRAIN-ALL)																
XLM-R_{large}	89.1	85.1	86.6	85.7	85.3	85.9	83.5	83.2	83.1	83.7	81.5	83.7	81.6	78.0	78.1	83.6
PCT-XLM-R_{large}	88.7	85.0	86.0	85.2	84.8	86.3	83.2	82.2	82.6	83.8	81.5	82.9	80.7	78.2	75.1	83.1

Table 4: **Comparison results on XNLI under the TRANSLATE-TRAIN-ALL setting.** Every value is the test accuracy in percent. △ is the average accuracy.

XLM-R_{base} (denoted by XLM-$R^{\dagger}_{base}$). PCT-XLM-R_{base} achieves 85.4% accuracy on the test set averaged by 7 languages, pushing XLM-R_{base}† by an absolute gain of 1.1% accuracy on average. The difference between PCT-XLM-R_{base} and XLM-R_{base}† in average accuracy is statistically significant with p-value 3.2e-3 by a two-tailed t-test. PCT-XLM-R_{large} achieves 88.3% average accuracy on the PAWS-X test set, pushing XLM-R_{large} by an absolute gain of 1.9% accuracy on average. The difference between PCT-XLM-R_{large} and XLM-R_{large} in average accuracy is statistically significant with p-value 3.2e-3 by a two-tailed t-test.

Table 3 reports the results for comparing PCT-XLM-R_{base} with all approaches proposed in (Zhao and Schütze, 2021). Note that all compared models are based on XLM-R_{base} and we evaluated PCT-XLM-R_{base} using the same split of data from (Zhao and Schütze, 2021). The training and validation data are randomly sampled by (Zhao and Schütze, 2021) with $K \in \{16, 32, 64, 128, 256\}$ shots per class from the English training data in XNLI. Results show that PCT-XLM-R_{base} statistically outperforms all baselines in all experiments. In particular, PCT-XLM-R_{base} outperforms the fine-tuning baseline by an absolute gain of 9.4% accuracy on average in the 16-shot experiments. It can also be seen that the difference between PCT-XLM-R_{base} and fine-tuning baseline becomes larger as K decreases, implying that the PCT framework becomes more effective when training data are fewer.

4.5 Evaluation on Translated Training Data

We also evaluated PCT on XNLI under the TRANSLATE-TRAIN-ALL setting, where all translated data are used in training, to see how well PCT is adapted to this setting. We construct an original question from the template of each of the 15 languages and an augmented question from a sampled template of other languages. Table 4 reports the comparison results. PCT-XLM-R_{large} under this setting achieves significantly better performance than under the cross-lingual transfer setting, but fails to outperform its original model XLM-R_{large}. This inferiority may be caused by the relatively low quality of examples in source languages. Note that an example in a source language other than English is translated from an English example

Variant Models	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	△	p-value
Original PCT-XLM-R$_{\text{base}}$	**84.9**	79.4	**79.7**	77.7	**76.6**	**78.9**	**76.9**	**74.3**	**72.9**	**76.0**	72.0	**74.9**	**71.7**	65.9	**67.3**	**75.3**	-
(1) W/o the consistency loss	84.6	**79.6**	79.5	76.7	76.3	78.1	76.0	73.9	72.1	75.0	**72.3**	73.9	71.1	63.9	66.8	74.7	1.2e-3
(2) W/o the PCT framework	83.9	78.1	78.5	76.1	75.7	77.1	75.3	73.2	71.6	74.7	70.9	73.4	70.2	63.6	65.5	73.9	1.5e-9
(3) Using cross-lingual templates in (2)	83.9	77.4	78.3	75.6	75.1	76.5	74.8	72.3	70.8	74.3	70.6	72.0	69.7	63.7	65.0	73.3	3.0e-10
(4) W/o the cross-lingual templates	84.8	79.5	79.6	**77.9**	76.4	78.2	76.7	74.2	72.5	**76.0**	71.9	74.6	71.6	64.8	66.9	75.0	3.0e-2
(5) Using substitute word templates	84.6	79.0	79.6	77.1	76.5	77.9	75.9	73.8	72.0	75.5	71.5	73.9	70.6	**66.3**	65.8	74.7	4.2e-4

Table 5: **Ablation study results for PCT-XLM-R$_{\text{base}}$.** Every value is the test accuracy in percent. △ is the average accuracy for 15 languages. The p-value is calculated by two-tailed t-tests.

Variant Models	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	△	p-value
PCT-XLM-R$_{\text{base}}$ (uniform)	84.9	79.4	79.7	77.7	76.6	78.9	76.9	74.3	**72.9**	76.0	**72.0**	**74.9**	**71.7**	65.9	**67.3**	**75.3**	-
PCT-XLM-R$_{\text{base}}$ (directly proportional)	**85.1**	79.4	79.8	**77.8**	76.0	78.4	**78.4**	73.9	72.7	75.8	71.3	74.2	71.6	64.1	66.9	75.0	0.28
PCT-XLM-R$_{\text{base}}$ (inversely proportional)	84.5	**80.1**	**80.4**	**77.8**	**76.9**	**79.0**	76.1	**74.4**	72.8	**76.1**	71.5	74.8	**71.7**	**66.0**	67.1	**75.3**	0.91
PCT-XLM-R$_{\text{large}}$ (uniform)	88.1	84.2	**85.1**	83.7	83.1	84.4	81.9	**81.2**	**80.9**	80.7	**78.8**	80.3	78.4	**73.6**	**75.6**	81.3	-
PCT-XLM-R$_{\text{large}}$ (directly proportional)	**88.4**	84.0	84.5	**84.0**	83.1	84.2	81.7	80.8	80.3	80.7	78.1	80.3	78.5	73.4	74.9	81.1	0.03
PCT-XLM-R$_{\text{large}}$ (inversely proportional)	**88.4**	**84.4**	84.8	83.8	**83.2**	**84.5**	**82.0**	80.8	80.6	**81.1**	78.6	**80.8**	**78.9**	**73.6**	74.9	**81.4**	0.81

Table 6: **Comparison results for template selection.** Every value is the test accuracy in percent. △ is the average accuracy for 15 languages. The p-value is calculated by two-tailed t-tests. "uniform" denotes the strategy with uniform selection probabilities. "directly proportional" and "inversely proportional" denote two strategies where the selection probabilities are directly proportional to and inversely proportional to the XX-En BLEU scores.

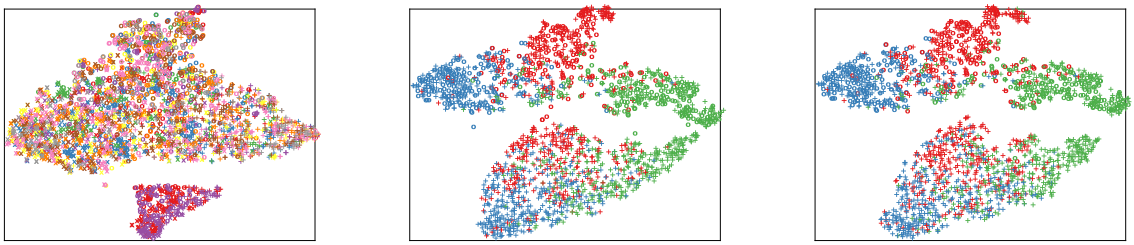
and may have translation errors. As a future work, we will go on studying whether using training data in multiple languages helps to improve XNLI by collecting more real-world data in other languages.

4.6 Ablation Study

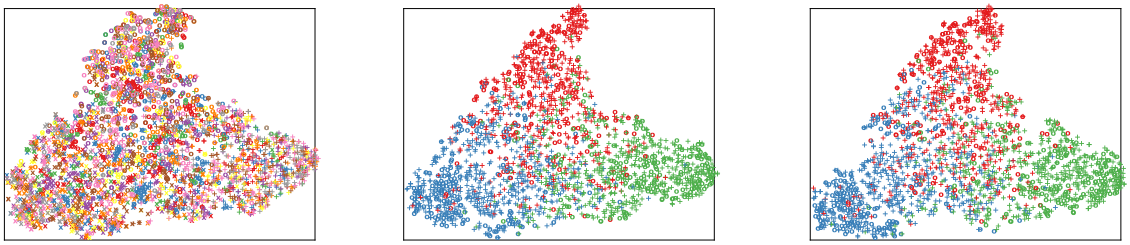
Table 5 reports the ablation study results for PCT-XLM-R$_{\text{base}}$. For the variant (1), we omit the consistency loss in course of training. Results show that the usage of consistency loss achieves better performance on average. For (2), we omit the whole PCT framework in course of training. Results show that the usage of PCT pushes XLM-R$_{\text{base}}$ with standard prompt-learning by an absolute gain of 1.4%. For (3), we apply the cross-lingual templates to the variant (2). Results show that the performance drops about 0.6% on average when applying only the cross-lingual templates. For (4), we use only the English template in the inference phase. Results show that PCT-XLM-R$_{\text{base}}$ achieves better performance on average when the cross-lingual templates are used in inference. For (5), we use the substitute word templates for Arabic and Urdu as for other languages, i.e., the templates for Arabic and Urdu are also left-to-right written. Results show that PCT-XLM-R$_{\text{base}}$ is able to capture certain language-specific characteristics in the target language to achieve better performance.

4.7 Visualization Analysis

To clarify why the proposed PCT framework improves accuracy in predicting NLI labels, we visually compared the representations of the [MASK] token generated by standard prompt-learning based XLM-R$_{\text{base}}$ (denoted by PL-XLM-R$_{\text{base}}$) with that generated by PCT-XLM-R$_{\text{base}}$, by using t-SNE (Laurens and Hinton, 2008) to reduce the dimension. The results are shown in Figure 3. For the sub-figures (a) and (d), the points marked with "x", "+" and "o" correspond to examples with the label "entailment", "contradiction" and "neutral", respectively. The points with different colors correspond to examples in different languages. The figures were obtained by randomly selecting 200 examples for each language from the XNLI test set. It can be seen in (a) that a group of red points (for Urdu) and purple points (for Arabic) are dissociated while all points from different languages are mutually overlapped in (d). Considering that the the points from Arabic and Urdu are quite different, we further analyzed them. For the sub-figures (b), (c), (e) and (f), the points marked with "o" and "+" respectively correspond to examples in English and in either Arabic or Urdu. The points with blue, red and green color correspond to examples with the label "entailment", "neutral" and "contradiction", respectively. Sub-figures (b) and (e) (resp. sub-figures (c) and (f)) were obtained by randomly selecting 1000 examples in English and 1000 in Arabic (resp. in Urdu) from the XNLI test set. Compared with PL-XLM-R$_{\text{base}}$, PCT-XLM-R$_{\text{base}}$ yields clearer distinction between different labels and more confusion between English and the target language (Arabic or Urdu). These results imply that the PCT framework tends to align contextualized representations in different languages into the



(a) PL-XLM-R(all) (b) PL-XLM-R(en+ar) (c) PL-XLM-R(en+ur)



(d) PCT-XLM-R(all) (e) PCT-XLM-R(en+ar) (f) PCT-XLM-R(en+ur)

Figure 3: Visualization of the [MASK] representations.

same space, which helps to improve the prediction accuracy in the XNLI task.

4.8 Different Strategies for Template Selection

We also conducted experiments to show how different strategies for template selection impact the performance. The results are reported in Table 6. We compared the default uniform strategy with two different selection strategies, where one sets the probabilities for selecting XX directly proportional to and the other inversely proportional to the XX-En BLEU scores, which are directly taken from Table 3 in (Conneau et al., 2018) and can be considered as similarity degrees between the target languages XX and English. Results show that the performances of both PCT-XLM-R_{base} and PCT-XLM-R_{large} slightly drop when using the "directly proportional" strategy. It can also be seen that, PCT-XLM-R_{base} with the "inversely proportional" strategy achieves the same average accuracy as with the uniform strategy, while PCT-XLM-R_{large} with the "inversely proportional" strategy is lightly better than with the uniform strategy. This implies that the "inversely proportional" strategy is able to improve the performance by selecting more templates in target languages that are less similar to English. However, the improvements are not significant as p-value > 0.05 by two-tailed t-tests. By considering that XX-En BLEU scores are not available in most practical scenarios, we recommend to use the uniform strategy for template selection.

5 Conclusions

In this paper we have proposed a prompt-learning based framework named PCT for cross-lingual natural language inference. PCT enhances pre-trained cross-lingual language models by augmenting data from cross-lingual templates and by introducing the consistency loss to regularize the answer probability distributions. Experimental results on large-scale benchmarks XNLI and PAWS-X show that PCT pushes existing models by a significant absolute gain in accuracy under both the full-shot and few-shot cross-lingual transfer settings. Our ablation study and visualization analysis further confirm the contributions of different enhancements introduced by PCT. Future work will study PCT further under the TRANSLATE-TRAIN-ALL setting with real-world data in different languages.

Acknowledgements

This paper was supported by the National Natural Science Foundation of China (No. 61976232 and 61876204), Guangdong Basic and Applied Basic Research Foundation (No.2022A1515011355 and 2020A1515010642), Guizhou Science Support Project (No. 2022-259), Humanities and Social Science Research Project of Ministry of Education (18YJCZH006).

References

M. Saiful Bari, Tasnim Mohiuddin, and Shafiq R. Joty. 2021. UXLA: A robust unsupervised data augmentation framework for zero-resource cross-lingual NLP. In *ACL*, pages 1978–1992.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *NeurIPS*, pages 1877–1901.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021a. Infoxlm: An information-theoretic framework for cross-lingual language model pre-training. In *NAACL-HLT*, pages 3576–3588.

Zewen Chi, Li Dong, Bo Zheng, Shaohan Huang, Xian-Ling Mao, Heyan Huang, and Furu Wei. 2021b. Improving pretrained cross-lingual language models via self-labeled word alignment. In *ACL*, pages 3418–3430.

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 *ACL*, pages 8440–8451.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *NeurIPS*, pages 7057–7067.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: evaluating cross-lingual sentence representations. In *EMNLP*, pages 2475–2485.

Yann N. Dauphin, Harm de Vries, and Yoshua Bengio. 2015. Equilibrated adaptive learning rates for non-convex optimization. In *NIPS*, pages 1504–1512.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186.

Xin Dong, Yaxin Zhu, Zuohui Fu, Dongkuan Xu, and Gerard de Melo. 2021. Data augmentation with adversarial training for cross-lingual NLI. In *ACL*, pages 5158–5167.

Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2021. PTR: prompt tuning with rules for text classification. *CoRR*, abs/2105.11259.

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 generalisation. In *ICML*, pages 4411–4421.

Haoyang Huang, Yaobo Liang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, and Ming Zhou. 2019. Unicoder: A universal language encoder by pre-training with multiple cross-lingual tasks. In *EMNLP*, pages 2485–2494.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Van der Maaten Laurens and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *ACL*, pages 4582–4597.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. XGLUE: A new benchmark datasetfor cross-lingual pre-training, understanding and generation. In *EMNLP*, pages 6008–6018.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021. GPT understands, too. *CoRR*, abs/2103.10385.

Kunxun Qi and Jianfeng Du. 2020. Translation-based matching adversarial network for cross-lingual natural language inference. In *AAAI*, pages 8632–8639.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *EACL*, pages 255–269.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *EMNLP*, pages 4222–4235.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research (JMLR)*, 15(1):1929–1958.

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

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *EMNLP*, pages 3685–3690.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: paraphrase adversaries from word scrambling. In *NAACL-HLT*, pages 1298–1308.

Mengjie Zhao and Hinrich Schütze. 2021. Discrete and soft prompting for multilingual models. In *EMNLP*, pages 8547–8555.

A Cross-lingual Templates

Here we introduce the cross-lingual templates that we used in our experiments. We used the same English template defined by (Zhao and Schütze, 2021) for XNLI and used the English template defined by (Brown et al., 2020) for PAWS-X. The cross-lingual templates are generated by translating the English template to target languages using Google translator. The cross-lingual templates for XNLI are given in Figure 4. The cross-lingual templates for PAWS-X are given in Figure 5. The slots [P] and [H] are filled by the premise and the hypothesis, respectively.

B Results with Standard Deviations

Here we report the complete experimental results taken from five runs with standard deviations. The means and the standard deviations are reported in the row "avg." and "s.d.", respectively.

For XNLI under the full-shot cross-lingual transfer setting, the experimental results are reported in Table 7, including the results for all five runs achieved by PCT-XLM-R_{base}, PCT-XLM-R_{large} and INFOXLM-R_{base}.

For PAWS-X under the full-shot cross-lingual transfer setting, the experimental results are reported in Table 8, including the results for all five runs achieved by PCT-XLM-R_{base}, PCT-XLM-R_{large} and INFOXLM-R_{base}.

For XNLI under the few-shot cross-lingual transfer setting, the experimental results with $K \in \{16, 32, 64, 128, 256\}$ shots per class are reported in Table 9, including the results for all five runs achieved by PCT-XLM-R_{base}.

Template	Language
<s>[P]</s></s>Question: [H]? Answer: <mask></s>	English (en)
<s>[P]</s></s>Question: [H]? Réponse: <mask></s>	French (fr)
<s>[P]</s></s>Pregunta: [H]? Respuesta: <mask></s>	Spanish (es)
<s>[P]</s></s>Frage: [H]? Antwort: <mask></s>	German (de)
<s>[P]</s></s>Ερώτηση: [H]? Απάντηση: <mask></s>	Greek (el)
<s>[P]</s></s>Въпрос: [H]? Отговор: <mask></s>	Bulgarian (bg)
<s>[P]</s></s>Вопрос: [H]? Ответ: <mask></s>	Russian (ru)
<s>[P]</s></s><mask> :الجواب ؟:[H] :سؤال </s>	Arabic (ar)
<s>[P]</s></s>Soru: [H]? Cevap: <mask></s>	Turkish (tr)
<s>[P]</s></s>Câu hỏi: [H]? Trả lời: <mask></s>	Vietnamese (vi)
<s>[P]</s></s>คำถาม: [H]? คำตอบ: <mask></s>	Thai (th)
<s>[P]</s></s>问题: [H]? 答案: <mask></s>	Chinese (zh)
<s>[P]</s></s>प्रश्न: [H]? उत्तर: <mask></s>	Hindi (hi)
<s>[P]</s></s>Swali: [H]? Jibu: <mask></s>	Swahili (sw)
<s>[P]</s></s><mask> :جواب ؟:[H] :سوال </s>	Urdu (ur)

Figure 4: Cross-lingual templates for XNLI.

Template	Language
<s>[P]</s></s>Question: [H] paraphrase or not? Answer: <mask></s>	English (en)
<s>[P]</s></s>Question: [H] paraphrase ou pas? Réponse: <mask></s>	French (fr)
<s>[P]</s></s>Pregunta: [H] ¿parafrasear o no? Respuesta: <mask></s>	Spanish (es)
<s>[P]</s></s>Frage: [H] paraphrasieren oder nicht? Antwort: <mask></s>	German (de)
<s>[P]</s></s>質問: [H] 言い換えるかどうか? 回答: <mask></s>	Japanese (ja)
<s>[P]</s></s>질문: [H] 의역 여부는? 답: <mask></s>	Korean (ko)
<s>[P]</s></s>问题: [H]是否转述? 答案: <mask></s>	Chinese (zh)

Figure 5: Cross-lingual templates for PAWS-X.

Models	Runs	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	△
Train multilingual model on training data in English (Cross-lingual Transfer)																	
PCT-XLM-R$_{\text{base}}$	1	85.2	79.7	79.9	77.9	76.8	79.0	76.8	73.5	72.5	75.9	71.8	75.3	71.9	65.2	66.9	75.2
	2	84.9	79.1	80.0	78.1	76.5	79.2	76.9	73.8	73.0	76.2	71.8	74.7	71.6	65.9	67.2	75.3
	3	84.8	78.9	79.3	77.0	76.6	78.6	76.9	74.3	73.4	76.0	72.5	74.9	71.7	65.7	67.8	75.2
	4	84.8	79.6	79.6	77.7	76.7	78.7	76.7	74.3	72.7	75.8	72.0	75.1	71.8	66.9	67.0	75.3
	5	85.0	79.5	79.8	77.6	76.6	78.9	77.0	74.2	72.7	75.9	72.0	74.5	71.7	65.7	67.8	75.3
	avg.	84.9	79.4	79.7	77.7	76.6	78.9	76.9	74.0	72.9	76.0	72.0	74.9	71.7	65.9	67.3	75.3
	s.d.	±0.06	±1.35	±0.77	±1.55	±0.24	±0.46	±0.40	±0.29	±1.34	±0.48	±1.07	±0.48	±0.24	±2.34	±1.55	±0.13
PCT-XLM-R$_{\text{large}}$	1	88.3	84.1	85.0	83.5	82.8	84.1	81.6	80.6	80.5	80.6	78.6	80.7	79.0	73.0	75.0	81.2
	2	88.4	84.2	84.7	84.0	83.3	84.5	82.2	80.8	80.4	80.9	78.3	80.3	79.1	73.8	75.1	81.3
	3	88.1	84.2	85.1	83.7	83.1	84.4	81.9	81.2	80.9	80.7	78.8	80.3	78.4	73.6	75.6	81.3
	4	88.1	84.3	85.1	84.0	83.2	84.5	81.9	80.6	80.7	80.8	78.4	80.4	78.8	73.6	75.1	81.3
	5	88.4	84.0	84.6	83.9	83.0	84.2	81.9	80.7	80.5	81.2	78.2	80.3	78.2	73.8	75.1	81.2
	avg.	88.3	84.2	84.9	83.8	83.1	84.3	81.9	80.8	80.6	80.8	78.5	80.4	78.7	73.6	75.2	81.3
	s.d.	±0.15	±0.11	±0.23	±0.22	±0.19	±0.18	±0.21	±0.25	±0.20	±0.23	±0.24	±0.17	±0.39	±0.33	±0.24	±0.08
INFOXLM-R$_{\text{large}}$	1	88.3	84.6	85.6	84.6	83.8	85.1	82.5	81.6	81.4	81.9	79.9	81.5	79.9	75.4	75.8	82.1
	2	88.7	84.5	85.1	84.7	83.6	84.1	82.1	81.3	80.9	81.7	79.6	81.4	79.2	75.9	75.6	81.9
	3	88.5	84.5	85.4	84.3	83.3	85.0	82.3	81.4	81.0	81.5	79.3	81.1	79.6	75.7	75.4	81.9
	4	88.8	84.8	85.1	84.5	83.7	84.6	82.2	81.3	81.2	81.3	79.5	81.3	79.1	75.9	75.6	81.9
	5	88.6	84.3	85.7	85.0	83.4	84.5	82.2	81.6	80.9	82.0	79.4	81.6	79.5	75.2	75.6	82.0
	avg.	88.6	84.5	85.4	84.6	83.6	84.7	82.3	81.4	81.1	81.7	79.5	81.4	79.5	75.6	75.6	82.0
	s.d.	±0.19	±0.18	±0.28	±0.26	±0.21	±0.40	±0.15	±0.15	±0.22	±0.29	±0.23	±0.19	±0.32	±0.31	±0.14	±0.10

Table 7: **Comparison results on XNLI under the full-shot cross-lingual transfer setting.** Every value is the test accuracy in percent. △ is the average accuracy for 15 languages.

Models	Runs	en	fr	es	de	ja	ko	zh	△
PCT-XLM-R$_{\text{base}}$	1	94.1	89.7	88.9	87.9	78.0	77.5	82.3	85.5
	2	94.1	89.6	89.2	87.9	77.7	77.8	81.9	85.5
	3	95.0	90.0	89.1	87.5	77.5	76.8	81.5	85.3
	4	94.7	90.0	88.9	88.6	76.9	76.9	80.9	85.3
	5	94.6	89.6	89.6	88.0	77.7	77.3	82.2	85.6
	avg.	94.5	89.8	89.1	88.0	77.6	77.3	81.8	85.4
	s.d.	±0.39	±0.20	±0.29	±0.40	±0.41	±0.42	±0.57	±0.12
PCT-XLM-R$_{\text{large}}$	1	95.8	92.2	90.7	90.1	82.2	82.3	83.7	88.1
	2	96.1	92.2	91.0	90.6	82.9	81.7	84.6	88.4
	3	95.3	92.5	91.2	90.5	82.6	82.2	84.5	88.4
	3	95.8	92.1	91.4	90.4	81.3	81.2	83.8	88.0
	5	95.1	91.8	91.6	91.0	81.9	82.3	84.5	88.3
	avg.	95.6	92.2	91.2	90.5	82.2	81.9	84.2	88.3
	s.d.	±0.41	±0.25	±0.35	±0.33	±0.62	±0.48	±0.43	±0.19

Table 8: **Comparison results on PAWS-X under the full-shot cross-lingual transfer setting.** Every value is the test accuracy in percent. △ is the average accuracy for 7 languages.

Shots	Models	Runs	en	fr	es	de	el	bg	ru	tr	ar	vi	th	zh	hi	sw	ur	△
		1	47.2	40.6	40.4	36.7	45.0	41.1	41.5	44.7	43.5	46.1	44.7	45.1	44.7	40.2	42.2	42.9
		2	45.4	45.4	40.3	35.4	45.6	39.6	42.6	41.9	42.7	41.1	41.8	43.5	44.2	39.5	41.9	42.1
		3	46.9	46.7	44.9	40.0	47.1	42.9	43.6	43.8	44.3	46.3	44.6	45.2	46.4	41.1	43.4	44.5
K=16	PCT	4	46.9	46.7	44.9	40.0	47.1	42.9	43.6	43.8	44.3	46.3	44.6	45.2	46.4	41.1	43.4	44.5
		5	46.5	44.1	42.2	37.3	46.0	42.1	42.9	44.4	44.6	46.1	45.1	46.0	45.5	40.1	42.9	43.7
		avg.	46.5	44.3	41.5	36.9	45.7	40.8	42.4	43.7	43.6	44.7	43.9	44.8	44.8	40.1	42.5	43.1
		s.d	±0.68	±2.28	±2.11	±1.96	±0.97	±1.90	±1.00	±1.09	±0.88	±2.23	±1.34	±0.97	±1.19	±0.68	±0.65	±1.03
		1	50.3	49.9	47.2	45.7	48.1	47.2	46.4	46.0	46.2	48.1	44.8	46.7	47.6	41.1	43.8	46.6
		2	50.5	49.6	47.3	46.4	48.0	47.2	46.4	44.2	46.3	48.3	41.3	45.9	47.7	40.2	42.5	46.1
		3	48.1	46.8	42.0	40.7	45.7	42.3	43.6	42.6	44.4	43.5	39.4	44.2	45.4	39.5	42.0	43.3
K=32	PCT	4	49.8	49.3	46.5	46.4	48.5	46.3	46.3	44.5	46.2	46.3	41.0	45.7	47.1	40.0	42.6	45.8
		5	49.3	48.2	44.6	42.6	46.9	43.8	45.0	44.4	45.6	47.2	41.6	45.6	45.6	40.7	43.4	45.0
		avg.	49.6	48.8	45.5	44.4	47.4	45.4	45.5	44.3	45.7	46.7	41.6	45.6	46.7	40.3	42.9	45.4
		s.d	±0.96	±1.27	±2.25	±2.58	±1.14	±2.21	±1.24	±1.21	±0.80	±1.95	±1.97	±0.90	±1.10	±0.62	±0.73	±1.28
		1	50.5	49.9	49.6	48.6	49.4	49.9	48.0	46.2	47.2	48.9	47.3	47.7	47.2	44.4	43.3	47.9
		2	50.3	50.9	49.8	49.0	49.4	49.8	48.7	47.8	47.8	48.8	46.7	48.0	47.2	43.9	44.2	48.2
		3	51.3	51.2	50.5	49.4	51.4	50.1	49.6	47.8	48.1	49.3	46.8	47.9	47.3	44.4	43.7	48.6
K=64	PCT	4	52.7	52.2	52.4	50.1	51.4	50.8	50.0	48.2	48.9	50.8	48.1	48.5	47.9	45.2	44.9	49.5
		5	52.5	52.5	52.1	49.3	51.6	50.1	49.2	46.8	48.4	50.5	47.5	48.7	48.2	44.9	43.9	49.1
		avg.	51.5	51.3	50.9	49.3	50.6	50.1	49.1	47.4	48.1	49.7	47.3	48.2	47.6	44.6	44.0	48.6
		s.d	±1.11	±1.05	±1.30	±0.55	±1.13	±0.39	±0.78	±0.83	±0.64	±0.93	±0.57	±0.42	±0.46	±0.50	±0.60	±0.65
		1	55.6	53.5	53.6	53.0	53.8	51.9	51.4	51.0	50.6	51.8	49.6	50.6	51.9	46.8	46.6	51.4
		2	53.3	51.5	53.0	52.1	51.9	50.8	50.7	48.5	49.0	48.8	48.1	49.1	50.3	45.2	47.7	50.0
		3	55.0	54.1	53.6	52.3	53.9	51.7	50.8	51.8	51.4	52.4	51.9	52.2	51.6	48.2	47.6	51.9
K=128	PCT	4	54.7	53.2	53.9	53.0	53.4	52.4	51.8	51.4	50.7	52.7	50.7	52.4	51.8	47.1	47.9	52.3
		5	56.2	54.2	54.7	53.8	54.0	52.9	53.9	51.8	50.4	52.6	49.6	51.9	51.8	47.5	49.5	52.3
		avg.	55.0	53.3	53.8	52.8	53.4	51.9	51.7	50.9	50.4	51.7	50.0	51.2	51.5	47.0	47.9	51.5
		s.d	±1.10	±1.09	±0.62	±0.67	±0.87	±0.79	±1.30	±1.38	±0.88	±1.63	±1.42	±1.39	±0.67	±1.11	±1.05	±0.89
		1	60.9	57.9	57.7	55.7	57.7	56.1	54.3	54.7	54.5	57.6	55.5	56.6	54.6	51.4	51.5	55.8
		2	60.8	58.1	58.6	56.8	57.9	57.3	55.8	54.5	54.7	57.7	54.9	56.0	54.1	50.5	52.8	56.0
		3	59.6	58.7	58.8	55.7	57.7	57.2	54.7	53.9	54.8	57.9	55.6	56.0	54.8	51.2	52.2	55.9
K=256	PCT	4	59.9	58.3	58.3	56.3	57.5	55.9	55.6	54.6	55.0	56.2	55.9	54.7	54.5	52.1	52.7	55.8
		5	60.3	58.3	58.1	57.2	58.9	56.9	55.7	55.5	54.5	57.5	55.9	55.9	55.2	52.8	53.8	56.4
		avg.	60.3	58.3	58.3	56.3	57.9	56.7	55.2	54.6	54.7	57.4	55.6	55.8	54.6	51.6	52.6	56.0
		s.d	±0.56	±0.30	±0.43	±0.67	±0.55	±0.64	±0.68	±0.57	±0.21	±0.68	±0.41	±0.69	±0.40	±0.88	±0.85	±0.26

Table 9: **Comparison results on XNLI under the few-shot cross-lingual transfer setting.** Every value is the test accuracy in percent. △ is the average accuracy.