Enhancing Cross-lingual Natural Language Inference by Soft Prompting with Multilingual Verbalizer

Shuang Li[1], **Xuming Hu**[1], **Aiwei Liu**[1], **Yawen Yang**[1], **Fukun Ma**[1],
Philip S. Yu[2], **Lijie Wen**[1*]
[1]Tsinghua University, [2]University of Illinois Chicago
[1]`{lisa18,hxm19,liuaw20,yyw19,mfk22}@mails.tsinghua.edu.cn`
[2]`psyu@uic.edu`, [1]`wenlj@tsinghua.edu.cn`

Abstract

Cross-lingual natural language inference is a fundamental problem in cross-lingual language understanding. Many recent works have used prompt learning to address the lack of annotated parallel corpora in XNLI. However, these methods adopt discrete prompting by simply translating the templates to the target language and need external expert knowledge to design the templates. Besides, discrete prompts of human-designed template words are not trainable vectors and can not be migrated to target languages in the inference stage flexibly. In this paper, we propose a novel **Soft** prompt learning framework with the **M**ultilingual **V**erbalizer (SoftMV) for XNLI. SoftMV first constructs cloze-style question with soft prompts for the input sample. Then we leverage bilingual dictionaries to generate an augmented multilingual question for the original question. SoftMV adopts a multilingual verbalizer to align the representations of original and augmented multilingual questions into the same semantic space with consistency regularization. Experimental results on XNLI demonstrate that SoftMV can achieve state-of-the-art performance and significantly outperform the previous methods under the few-shot and full-shot cross-lingual transfer settings[1].

1 Introduction

Multilingual NLP systems have gained more attention due to the increasing demand for multilingual services. Cross-lingual language understanding (XLU) plays a crucial role in multilingual systems, in which cross-lingual natural language inference (XNLI) is a fundamental and challenging task (Conneau et al., 2018; MacCartney and Manning, 2008; Li et al., 2023, 2022). NLI is a fundamental problem in NLU that could help with tasks like semantic parsing (Liu et al., 2022a; Lin et al., 2022), and relation extraction (Liu et al., 2022b; Hu et al., 2020, 2021). In XNLI settings, the model is trained on the source language with annotated data to reason the relationship between a pair of sentences (namely premise and hypothesis) and evaluated on the target language without parallel corpora.

Type	Prompt Templates
DP	Premise. `Question`: Hypothesis? `Answer`: <MASK>.
SP	Premise. Hypothesis? <v_1>...<v_n> <MASK>.
MP	Premise. `Question`: Hypothesis? <v_1>...<v_n> `Answer`: <MASK>.

Table 1: The example of prompt templates for Discrete Prompts (DP), Soft Prompts (SP), and Mixed Prompts (MP). Premise and Hypothesis are a pair of sentences from the NLI dataset. `Question` and `Answer` are template words of discrete prompts. <v_i> is the trainable vector of soft prompts.

Pre-trained multilingual language models, such as mBERT (Devlin et al., 2019), XLM (Conneau and Lample, 2019), and XLM-R (Conneau et al., 2020), have demonstrated promising performance in cross-lingual transfer learning. These language models learn a shared multilingual embedding space to represent words in parallel sentences. However, these models are trained on a large number of parallel corpora, which are not available in many low-resource languages. The major challenge of XNLI is the lack of annotated data for low-resource languages.

To address this problem, some works explored using prompt learning (Brown et al., 2020; Schick and Schütze, 2021a; Shin et al., 2020) when adapting pre-trained language models to downstream tasks in cross-lingual scenarios. Prompt learning reformulates the text classification problem into a masked language modeling (MLM) problem by constructing cloze-style questions with a special token `<MASK>`. The model is trained to predict the masked word in the cloze-style questions. As shown in Table 1, prompt learning can

[1] The source code will be available at `https://github.com/THU-BPM/SoftMV`.

[*] Corresponding Author.

Findings of the Association for Computational Linguistics: ACL 2023, pages 1361–1374
July 9-14, 2023 ©2023 Association for Computational Linguistics

be divided into three types: Discrete Prompts (DP), Soft Prompts (SP), and Mixed Prompts (MP). Zhao and Schütze (2021) investigated the effectiveness of prompt learning in multilingual tasks by simply applying soft, discrete, and mixed prompting with a uniform template in English. Qi et al. (2022) proposed a discrete prompt learning framework that constructs an augmented sample by randomly sampling a template in another language. By comparing the augmented samples and the original samples in the English template, the model can effectively perceive the correspondence between different languages. However, discrete prompts of human-designed template words require extensive external expert knowledge and are not flexible enough to adapt to different languages. Therefore, the model can't perform well when transferred from high-resource to low-resource languages.

In this paper, we propose a novel **Soft** prompt learning framework with the **M**ultilingual **V**erbalizer (SoftMV) for XNLI. First, we construct cloze-style questions for the input samples with soft prompts which consist of trainable vectors. Second, we apply the code-switched substitution strategy (Qin et al., 2021) to generate multilingual questions which can be regarded as cross-lingual views for the English questions. Compared with discrete prompts, soft prompts perform prompting directly in the embedding space of the model and can be easily adapted to any language without human-designed templates. Both the original and augmented questions are fed into a pre-trained cross-lingual base model. The classification probability distribution is calculated by predicting the masked token with the multilingual verbalizer to reduce the gap between different languages. Finally, the two probability distributions are regularized by the Kullback-Leibler divergence (KLD) loss (Kullback and Leibler, 1951) to align the representations of original and augmented multilingual questions into the same space. The entire model is trained with a combined objective of the cross-entropy term for classification accuracy and the KLD term for representation consistency. The well-trained soft prompt vectors will be frozen in the inference stage. Experimental results on the XNLI benchmark show that SoftMV outperforms the baseline models by a significant margin under both the few-shot and full-shot settings.

Our contributions can be summarized as follows:

- We propose a novel **Soft** prompt learning framework with a **M**ultilingual **V**erbalizer (SoftMV) for XNLI. SoftMV leverages bilingual dictionaries to generate augmented multilingual code-switched questions for original questions constructed with soft prompts.
- We adopt the multilingual verbalizer to align the representations of original and augmented questions into the same semantic space with consistency regularization.
- We conduct extensive experiments on XNLI and demonstrate that SoftMV can significantly outperform the baseline methods under the few-shot and full-shot cross-lingual transfer settings.

2 Related Work

Early methods for cross-lingual natural language inference are mainly neural networks, such as Conneau et al. (2018) and Artetxe and Schwenk (2019). which encode sentences from different languages into the same embedding space via parallel corpora (Hermann and Blunsom, 2014). In recent years, large pre-trained cross-lingual language models have demonstrated promising performance. Devlin et al. (2019) extend the basic language model BERT to multilingual scenarios by pre-trained with multilingual corpora. Conneau and Lample (2019) propose a cross-lingual language model (XLM) which enhances BERT with the translation language modeling (TLM) objective. XLM-R (Conneau et al., 2020) is an improvement of XLM by training with more languages and more epochs. Although these methods do not rely on parallel corpora, they still have limitations because fine-tuning needs annotation efforts which are prohibitively expensive for low-resource languages.

To tackle this problem, some data augmentation methods have been proposed for XNLI. Ahmad et al. (2021) propose to augment mBERT with universal language syntax using an auxiliary objective for cross-lingual transfer. Dong et al. (2021) adopt Reorder Augmentation and Semantic Augmentation to synthesize controllable and much less noisy data for XNLI. Bari et al. (2021) improve cross-lingual generalization by unsupervised sample selection and data augmentation from the unlabeled training examples in the target language. Zheng et al. (2021) propose a cross-lingual fine-tuning method to better utilize four types of data augmentations based on consistency regularization.

However, these methods do not perform well under the few-shot settings.

Recently, prompt learning (Brown et al., 2020; Shin et al., 2020; Lester et al., 2021; Vu et al., 2022; Li and Liang, 2021; Qin and Eisner, 2021; Liu et al., 2022c) has shown promising results in many NLP tasks under the few-shot setting. The key idea of prompt learning for XNLI is reformulating the text classification problem into a masked language modeling problem by constructing cloze-style questions. Su et al. (2022) propose a novel prompt-based transfer learning approach, which first learns a prompt on one or more source tasks and then uses it to initialize the prompt for a target task. Wu and Shi (2022) adopt separate soft prompts to learn embeddings enriched with domain knowledge. Schick and Schütze (2021a) explore discrete prompt learning to NLI with manually defined templates. Zhao and Schütze (2021) demonstrate that prompt learning outperforms fine-tuning for few-shot XNLI by simply applying soft, discrete, and mixed prompting with a uniform template in English. Qi et al. (2022) proposed a discrete prompt learning framework that constructs an augmented sample by randomly sampling a template in another language. However, discrete prompts of human-designed template words require extensive external expert knowledge and are not flexible enough to adapt to different languages. In our work, we adopt trainable soft prompts to capture correspondence between different languages by comparing the augmented multilingual and original questions.

3 Framework

The proposed SoftMV framework is illustrated in Figure 1. The training process of SoftMV is formalized in Algorithm 1. For every training triple (premise, hypothesis, label) in English, SoftMV first constructs a cloze-style question with soft prompts initialized from the vocabulary. Then, we apply the code-switched substitution strategy to generate multilingual questions which can be regarded as cross-lingual views for the English questions. Both the original and augmented questions are fed into a pre-trained cross-lingual model to calculate the answer distributions of the mask token with a multilingual verbalizer. SoftMV is trained by minimizing the cross-entropy loss for classification accuracy and the Kullback-Leibler divergence (KLD) loss for representation consistency. Finally, the well-trained soft prompt vectors are frozen in the inference stage.

3.1 Soft Prompting

Each instance in batch $\mathcal{I}$ in XNLI dataset is denoted as $(P_i, H_i, Y_i)_{i\in\mathcal{I}}$, where $P_i = \{w_j^P\}_{j=1}^m$ denotes the word sequence of premise, $H_i = \{w_j^H\}_{j=1}^n$ denotes the word sequence of hypothesis, and $Y_i \in \mathcal{Y}$ denotes the class label. SoftMV first constructs a cloze-style question with soft prompts as illustrated in Table 1. The question template is expressed as “<s>Premise.</s> <s>Hypothesis? <v_1>...<v_n> <MASK></s>", where <s> and </s> are special tokens to separate sentences, <MASK> is the mask token, and v_i is associated with a trainable vector (in the PLM's first embedding layer). Soft prompts are tuned in the continuous space and initialized with the average value of embeddings of the PLM's multilingual vocabulary. In cross-lingual transfer scenarios, it's a challenge for a model to align contextualized representations in different languages into the same semantic space when trained solely on the English dataset. Therefore, we adopt the code-switched strategy to create multilingual augmentations for the original questions. Followed by Qin et al. (2021), we use bilingual dictionaries (Lample et al., 2018) to replace the words of the original sentences. Specifically, for the English sentence, we randomly choose $n = \alpha * l$ words to be replaced with a translation word from a bilingual dictionary, where α is the code-switched rate and l is the length of the sentence. For example, given the sentence “Two men on bicycles competing in a

Algorithm 1 The training process of SoftMV.

```
Input: the number of epochs $E$ and the training
    set $\mathbb{D} = \{(P_i, H_i, Y_i)\}_{i=1}^M$.
Reform $\mathbb{D}$ to a set of cloze-style questions
    $\mathbb{Q} = \{(Q_i, Y_i)\}_{i=1}^M$ with soft prompts for each
    $(P_i, H_i)$ as illustrated in Figure 1.
Extend the set $\mathbb{Q} = \{(Q_i, Q_i^a, Y_i)\}_{i=1}^M$ by generating
    augmented multilingual questions with
    the code-switched strategy.
Divide $\mathbb{Q}$ into a set of batches $\mathbb{B}$.
for epoch $e = 1$ to $E$ do
    Shuffle $\mathbb{B}$.
    for each batch $\{(Q_i, Q_i^a, Y_i)\}_{1\le i\le N}$ in $\mathbb{B}$
    do
        Compute total loss $\mathcal{L}$ by Eq. 7.
        Update the parameters $\theta$.
    end for
end for
```

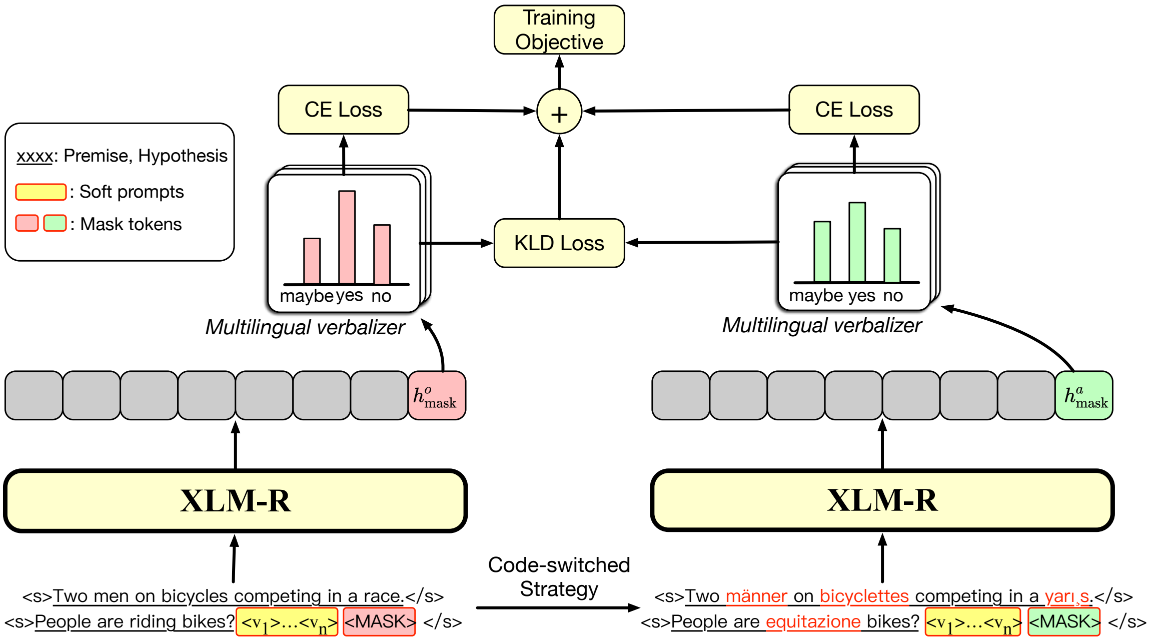
Training Objective
CE Loss
CE Loss
KLD Loss
xxxx: Premise, Hypothesis
: Soft prompts
: Mask tokens
maybe yes no
maybe yes no
Multilingual verbalizer
Multilingual verbalizer
h^o_{mask}
h^a_{mask}
XLM-R
XLM-R
Code-switched Strategy
<s>Two men on bicycles competing in a race.</s>
<s>People are riding bikes? <v1>...<vn> <MASK> </s>
<s>Two männer on bicyclettes competing in a yarış.</s>
<s>People are equitazione bikes? <v1>...<vn> <MASK> </s>

Figure 1: The framework of SoftMV. The left part is the original questions. The right part is the augmented multilingual questions. The model is trained with a combined objective of the cross-entropy losses and the KLD loss.

race." in English, we can generate a multilingual code-switched sample "Two Männer(DE) on Bicyclettes(FR) competing in a yarış(TR)." which can be regarded as the cross-lingual view of the same meaning across different languages. The original and augmented cloze-style questions are fed into a pre-trained cross-lingual model to obtain the contextualized representation of the mask token, denoted as h^o_{mask} and h^a_{mask}. Let l denote the size of the vocabulary and d the dimension of the representation of the mask token, the answer probability distribution of the original question is calculated by:

$$y^o = softmax(\mathbf{W}h^o_{\text{mask}}), \qquad (1)$$

where $\mathbf{W} \in \mathbb{R}^{l \times d}$ is the trainable parameters of the pre-trained MLM layer. The answer probability distribution y^a of the augmented question is calculated in the same way.

3.2 Multilingual Verbalizer

After calculating the answer probability distribution of the mask token, we use the verbalizer to calculate the classification probability distribution. The verbalizer $\mathcal{M} \rightarrow \mathcal{V}$ is a function that maps NLI labels to indices of answer words in the given vocabulary. The model is trained to predict masked words that correspond to classification labels, as determined by the verbalizer. Concretely, the verbalizer of English is defined as {"Entailment" → "yes"; "Contradiction" → "no"; "Neutral" → "maybe"} according to Schick and Schütze (2021b).

Without parallel corpora in cross-lingual scenarios, there is a gap in the classification space between the original and multilingual representations. Using the English verbalizer for all languages might hinder the model's ability to capture semantic representations for multilingual inputs. Thus we use a multilingual verbalizer to learn a consistent classification probability distribution across different languages. The multilingual verbalizer comprises a set of verbalizers for different languages. The multilingual verbalizer is denoted as $\{\mathcal{M}_l, l \in \mathcal{L}\}$, where $\mathcal{L}$ is the set of languages and l is a specific language. The non-English verbalizers are translated from English using bilingual dictionaries. Specifically, the verbalizer of Turkish is defined as {"Entailment" → "Evet."; "Contradiction" → "hiçbir"; "Neutral" → "belki"}.

3.3 Training Objective

In the training stage, given a batch $\mathcal{I}$ of N triples denoted as $(X^o_i, X^a_i, Y_i)_{1 \leq i \leq N}$, the cross-entropy losses for the original question X^o_i and the augmented question X^a_i are respectively calculated by:

$$\ell_i^o = -\frac{1}{|\mathcal{L}|}\sum_{l\in\mathcal{L}}\sum_{j=1}^{N} I(j=\mathcal{M}_l(Y_i))\log y_{i,j}^o, \quad (2)$$

$$\ell_i^a = -\frac{1}{|\mathcal{L}|}\sum_{l\in\mathcal{L}}\sum_{j=1}^{N} I(j=\mathcal{M}_l(Y_i))\log y_{i,j}^a, \quad (3)$$

where $y_{i,j}^o$ (resp. $y_{i,j}^a$) denotes the j-th element of the answer probability distribution y^o for the original question X_i^o (resp. for the input X_i^a) and $I(C)$ is the indicator function that returns 1 if C is true or 0 otherwise. The cross-entropy losses of the original and augmented questions on the batch $\mathcal{I}$ are calculated by:

$$\mathcal{L}_O = -\frac{1}{N}\sum_{i=1}^{N}\ell_i^o, \quad (4)$$

$$\mathcal{L}_A = -\frac{1}{N}\sum_{i=1}^{N}\ell_i^a. \quad (5)$$

However, for the same premise and hypothesis, the answer probability distribution of the augmented multilingual question created by the code-switched strategy may lead to a deviation from that of the original question due to the misalignment of representations in the multilingual semantic space. Such a deviation may cause the model to learn the wrong probability distribution when the model is evaluated on target languages. To alleviate this problem, we propose a consistency regularization to constrain the answer probability distribution. In particular, we adopt the Kullback-Leibler divergence (KLD) to encourage the answer probability distribution of the augmented question to be close to that of the original question. The consistency loss is defined as:

$$\mathcal{L}_{KLD} = \frac{1}{N}\sum_{i=1}^{N}(\mathrm{KL}(y_i^o||y_i^a) + \mathrm{KL}(y_i^a||y_i^o)), \quad (6)$$

The cross-entropy loss encourages the model to learn correct predictions for the augmented inputs, while the KLD loss enforces consistency between the original and augmented representations in the same multilingual semantic space. Using these loss terms together ensures that the model not only performs well on the original inputs but also generalizes to the augmented inputs, resulting in a more robust model that effectively handles cross-lingual tasks. The overall objective in SoftMV is a tuned linear combination of the cross-entropy losses and KLD loss, defined as:

$$\mathcal{L} = \lambda_O\mathcal{L}_O + \lambda_A\mathcal{L}_A + \lambda_{KLD}\mathcal{L}_{KLD}, \quad (7)$$

where λ_* are tuning parameters for each loss term.

4 Experiment Setup

4.1 Benchmark Dataset

We conducted experiments on the large-scale multilingual benchmark dataset of XNLI (Conneau et al., 2018), which extends the MultiNLI (Williams et al., 2018) benchmark (in English) to 15 languages[2] through translation and comes with manually annotated development sets and test sets. For each language, the training set comprises 393K annotated sentence pairs, whereas the development set and the test set comprise 2.5K and 5K annotated sentence pairs, respectively.

We evaluate SoftMV and other baseline models under the few-shot and full-shot cross-lingual settings, where the models are only trained on English and evaluated on other languages. For the few-shot setting, the training and validation data are sampled by Zhao and Schütze (2021) with $k \in \{1, 2, 4, 8, 16, 32, 64, 128, 256\}$ shots per class from the English training data in XNLI. We report classification accuracy as the evaluation metric.

4.2 Implementation Details

We implement SoftMV using the pre-trained XLM-RoBERTa model (Conneau et al., 2020) based on PyTorch (Paszke et al., 2019) and the Huggingface framework (Wolf et al., 2020). XLM-R is a widely used multilingual model and the baseline (PCT) we compare with only report the results using XLM-R.

We train our model for 70 epochs with a batch size of 24 using the AdamW optimizer. The hyperparameter α is set to 0.3 for combining objectives. The maximum sequence length is set to 256. All the experiments are conducted 5 times with different random seeds ({1, 2, 3, 4, 5}) and we report the average scores. The trained soft prompt vectors will be frozen in the inference stage. Appendix A shows the hyperparameters and computing devices used under different settings in detail.

[2] The languages are English (EN), French (FR), Spanish (ES), German (DE), Greek (EL), Bulgarian (BG), Russian (RU), Turkish (TR), Arabic (AR), Vietnamese (VI), Thai (TH), Chinese (ZH), Hindi (HI), Swahili (SW), and Urdu (UR)

4.3 Baseline Models

We compared SoftMV with the following cross-lingual language models: (1) 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 objectives (MLM and TLM) on Wikipedia with 100 languages; (3) XLM-R (Conneau et al., 2020) extends XLM with larger corpora and more epochs; (4) The work (Dong et al., 2021) proposes an adversarial data augmentation scheme based on XLM-R; (5) UXLA (Bari et al., 2021) enhances XLM-R with data augmentation and unsupervised sample selection; (6) The work (Zhao and Schütze, 2021) explores three prompt-learning methods for few-shot XNLI, including DP, SP, and MP; (7) PCT (Qi et al., 2022) is a discrete prompt learning framework with cross-lingual templates.

5 Experiment Results

5.1 Main Results

We conducted experiments on the XNLI dataset under the cross-lingual transfer setting, where models are trained on the English dataset and then directly evaluated on the test set of all languages. The settings can be further divided into two sub-settings: the few-shot setting using a fixed number of training samples per class, and the full-shot setting using the whole training set.

Few-shot results Table 2 reports the results for comparing SoftMV with other models on XNLI under the few-shot setting. The results of compared models are taken from Zhao and Schütze (2021); Qi et al. (2022). PCT† in the 1/2/4/8-shot experiments are reproduced by us, for not being reported before. Note that all models are based on XLM-R$_{\texttt{base}}$ and trained on the same split of data from Zhao and Schütze (2021). Results show that SoftMV significantly outperforms all baselines for all languages under all settings by 3.5% on average. As expected, all models benefit from more shots. When the k shots per class decrease, the gap between the performance of SoftMV and the state-of-the-art model (PCT) becomes larger, implying our model has a stronger ability to align contextualized representations in different languages into the same space when training data are fewer. In particular, SoftMV outperforms PCT by 4.4%, 2.8%, 4.3%, and 8.9% in the 1/2/4/8-shot experiments respectively. When the k shots per class are larger than 8, the average performance of SoftMV also outperforms PCT by an absolute gain of 2.5% on average. Furthermore, for different languages, all methods perform best on EN (English) and worst on AR (Arabic), VI (Vietnamese), UR (Urdu), and SW (Swahili). It is difficult to obtain usable corpora for these low-resource languages for XLM-R. Thus, the model has a poor learning ability for these languages. SoftMV also outperforms PCT on these low-resource languages, which demonstrates that our model is more effective in cross-lingual scenarios, especially for low-resource languages.

Full-shot results Table 3 shows the results on XNLI under the full-shot setting. The results of compared models are taken from Qi et al. (2022). SoftMV-XLM-R$_{\texttt{base}}$ achieves 78.8% accuracy averaged by 15 target languages, significantly outperforming the basic model XLM-R$_{\texttt{base}}$ by 4.6% on average. Compared with PCT, SoftMV improves by 3.5% on average based on XLM-R$_{\texttt{base}}$. Furthermore, we can observe that the accuracy of SoftMV exceeds PCT by 0.3% on EN, but 4.6% on AR, 11.8% on SW, and 10.5% on UR. This indicates that SoftMV has better transferability across low-resource languages with well-trained soft prompt vectors. To further investigate the effectiveness, we also evaluated SoftMV with baselines based on XLM-R$_{\texttt{large}}$ model. It can be seen that SoftMV achieves 82.1% accuracy on average, significantly outperforming PCT and XLM-R$_{\texttt{large}}$ by 0.8% and 1.7%. Compared with the results on XLM-R$_{\texttt{base}}$, the improvements of SoftMV on XLM-R$_{\texttt{large}}$ are smaller, which indicates that SoftMV is more effective on XLM-R$_{\texttt{base}}$ which has fewer parameters and worse cross-lingual ability. The performance gains are due to the stronger ability of SoftMV to align contextualized representations in different languages into the same semantic space with consistency regularization.

5.2 Ablation Study

To better understand the contribution of each key component of SoftMV, we conduct an ablation study under the 8-shot setting with XLM-R$_{\texttt{base}}$. The results are shown in Table 4. After removing the code-switched method, the performance decreases by 1.9% on average which shows the augmented multilingual samples can help the model to understand other languages. When we remove the consistency loss, the average accuracy decreases by 2.5%. The consistency loss can help the model align the representations across different languages

Shots	Models	EN	FR	ES	DE	EL	BG	RU	TR	AR	VI	TH	ZH	HI	SW	UR	AVG.
1	DP	33.2	34.1	33.8	33.0	33.2	33.2	33.8	34.0	32.1	32.8	33.0	33.6	33.4	33.5	32.0	33.2
	SP	36.7	38.6	38.3	36.9	37.5	36.5	37.6	34.8	34.8	35.1	35.7	37.6	36.4	34.5	35.5	36.4
	MP	33.3	33.7	34.0	33.0	32.1	32.3	33.0	34.6	32.3	32.8	32.2	33.4	34.1	32.9	32.7	33.1
	PCT†	37.1	36.2	37.4	37.2	35.8	36.8	36.1	36.4	34.5	35.3	36.6	37.7	35.8	34.1	36.3	36.2
	Ours	**43.0**	**40.1**	**41.1**	**39.8**	**40.2**	**42.5**	**44.0**	**37.4**	**41.1**	**41.5**	**40.4**	**42.2**	**40.1**	**38.3**	**37.7**	**40.6**
2	DP	35.4	34.8	35.4	34.4	34.7	35.1	34.9	35.2	32.9	33.3	35.4	36.5	34.1	33.0	32.8	34.5
	SP	38.0	38.6	38.2	38.2	38.4	38.1	39.2	34.8	35.9	36.7	37.2	37.7	36.3	34.4	35.5	37.1
	MP	34.6	34.3	33.8	34.1	33.3	34.3	34.0	34.5	32.8	33.8	34.6	35.4	33.8	33.9	32.6	34.0
	PCT†	39.3	38.4	39.0	38.7	38.9	39.2	38.8	38.2	37.6	38.1	38.4	40.1	38.2	33.7	38.0	38.3
	Ours	**41.3**	**42.6**	**40.9**	**44.2**	**42.1**	**41.7**	**44.1**	**40.2**	**40.2**	**39.3**	**40.0**	**40.8**	**41.3**	**37.5**	**40.4**	**41.1**
4	DP	39.5	38.3	38.9	38.9	37.7	37.6	37.5	37.2	35.4	36.0	37.8	38.7	36.4	34.7	35.9	37.4
	SP	41.8	41.1	39.8	40.1	40.8	40.5	41.7	35.9	38.0	37.9	39.2	39.5	37.6	35.8	37.7	39.2
	MP	36.3	35.4	35.5	35.2	34.0	33.8	34.2	35.6	33.1	34.1	36.0	37.1	34.6	33.5	33.5	34.8
	PCT†	41.1	39.1	40.9	41.0	39.4	39.5	40.2	39.0	37.4	38.0	38.4	40.3	37.5	35.2	37.9	39.0
	Ours	**46.8**	**45.1**	**45.5**	**46.4**	**44.6**	**44.4**	**44.8**	**42.6**	**40.5**	**39.6**	**41.2**	**43.9**	**43.3**	**38.2**	**42.7**	**43.3**
8	DP	36.4	35.2	35.0	34.8	34.8	34.8	34.6	34.1	32.7	33.7	35.1	35.6	33.0	32.9	33.1	34.4
	SP	39.0	38.8	38.2	38.2	38.7	38.8	39.7	35.1	36.3	37.4	37.9	37.2	35.9	34.5	35.6	37.4
	MP	34.8	34.8	34.7	34.8	33.2	33.2	33.8	35.1	32.7	33.6	34.5	36.3	34.8	33.1	32.7	34.1
	PCT†	38.3	35.8	38.7	37.2	36.6	36.1	37.1	35.9	34.8	35.4	36.3	38.1	36.1	34.5	34.9	36.4
	Ours	**47.5**	**46.7**	**47.0**	**46.4**	**47.5**	**46.5**	**46.3**	**43.7**	**46.5**	**45.8**	**45.1**	**42.5**	**43.2**	**42.1**	**42.8**	**45.3**
16	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	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
	Ours	**48.8**	**48.0**	**47.1**	**47.7**	**47.2**	**47.4**	**47.8**	**44.3**	**45.6**	**46.6**	**44.9**	**46.1**	**44.9**	**43.4**	**43.3**	**46.2**
32	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	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
	Ours	**50.7**	**48.5**	**49.1**	**48.7**	**48.7**	**49.8**	**48.8**	**47.0**	**47.9**	**48.8**	**45.8**	**45.1**	**45.2**	**43.6**	**44.9**	**47.5**
64	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	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	48.7
	Ours	**54.0**	**53.6**	**52.3**	**51.1**	**50.7**	**52.6**	**51.4**	**50.1**	**48.9**	**51.4**	**51.2**	**53.1**	**51.1**	**46.3**	**48.9**	**51.1**
128	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	47.8
	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	48.1
	PCT	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
	Ours	**56.6**	**55.1**	**55.7**	**54.7**	**55.4**	**55.7**	**53.7**	**53.5**	**52.1**	**54.5**	**53.4**	**54.3**	**53.1**	**49.3**	**51.0**	**53.9**
256	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	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
	Ours	**63.3**	**59.5**	**61.0**	**59.5**	**58.6**	**60.5**	**57.8**	**56.4**	**58.2**	**59.2**	**59.1**	**60.6**	**56.1**	**56.0**	**53.5**	**58.6**

Table 2: Comparison results on XNLI under the few-shot cross-lingual transfer setting in accuracy(%). Each number is the mean performance of 5 runs. "AVG." is the average accuracy for 15 languages. PCT† denote our reproduced results of the model in Qi et al. (2022). The best performance is in **bold**.

into the same semantic space. Removing the multilingual verbalizer leads to 1.7% accuracy drop on average. This demonstrates that the multilingual verbalizer can reduce the gap between different languages when calculating the classification probability distribution. We also replace soft prompts with discrete prompts as illustrated in Table 1, which leads to an accuracy drop of 1.3% on average. The accuracy decreases by 1.0% when using mixed prompts instead of soft prompts. The reason is that template words in mixed prompts have a bad effect on SoftMV if not specifically designed with expert knowledge. Furthermore, we use randomly initialized prompts to replace the prompts initialized from the multilingual vocabulary, which leads to 0.5% accuracy drop on average.

5.3 Analysis of Code-switched Method

To further investigate the code-switched method, we conduct experiments using a single language to create augmented multilingual samples. Figure 2

Models	EN	FR	ES	DE	EL	BG	RU	TR	AR	VI	TH	ZH	HI	SW	UR	AVG.
mBERT	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	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
XLM-R$_{\texttt{base}}$	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
Dong et al. (2021)	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$_{\texttt{base}}$	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$_{\texttt{base}}$	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$_{\texttt{base}}$	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$_{\texttt{base}}$	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
SoftMV-XLM-R$_{\texttt{base}}$	**85.2**	**80.8**	**79.9**	**78.7**	**84.1**	**81.3**	**79.5**	**76.0**	**77.5**	**78.8**	**77.0**	**76.0**	**72.0**	**77.7**	**77.8**	**78.8**
XLM-R$_{\texttt{large}}$	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	-	-	85.7	84.2	-	-	-	-	80.5	-	-	-	78.7	74.7	73.4	-
PCT-XLM-R$_{\texttt{large}}$	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
SoftMV-XLM-R$_{\texttt{large}}$	**88.9**	**85.1**	**85.8**	**84.2**	**83.7**	**85.2**	**82.3**	**82.1**	**81.5**	**81.4**	**79.7**	**81.2**	**79.1**	**74.2**	**76.4**	**82.1**

Table 3: Comparison results on XNLI under the full-shot cross-lingual transfer setting in accuracy(%). Each number is the mean performance of 5 runs. "AVG." is the average accuracy for 15 languages. The best performance is in **bold**.

Models	EN	FR	ES	DE	EL	BG	RU	TR	AR	VI	TH	ZH	HI	SW	UR	AVG.
Original	**47.5**	**46.7**	**47.0**	**46.4**	**47.5**	**46.5**	**46.3**	**43.7**	**46.5**	**45.8**	**45.1**	**42.5**	**43.2**	**42.1**	**42.8**	**45.3**
w/o code-switched	46.8	45.4	44.9	45.2	45.7	45.4	45.0	41.4	44.8	44.2	42.7	38.5	40.4	38.9	41.1	43.4
w/o consistency loss	45.3	44.3	44.9	43.6	44.8	43.6	43.5	40.7	44.3	43.7	43.0	39.8	40.2	39.9	40.7	42.8
w/o multilingual verbalizer	44.8	44.7	44.5	43.7	45.0	44.8	44.8	43.2	43.0	43.6	43.1	42.0	42.9	41.6	42.4	43.6
using discrete prompts	46.0	45.4	46.0	45.1	45.4	45.4	45.5	42.2	44.6	44.7	44.2	40.8	42.2	41.4	41.6	44.0
using mixed prompts	46.2	45.8	46.1	45.6	45.7	45.1	45.8	42.3	44.7	44.9	44.6	41.0	42.5	42.0	41.7	44.3
using randomly initialized prompts	47.6	46.6	46.4	45.8	46.7	45.8	44.8	43.0	46.1	45.7	44.7	42.6	42.9	40.3	42.6	44.8

Table 4: Ablation study results for SoftMV under the 8-shot setting in accuracy(%). "AVG." is the average accuracy for 15 languages.

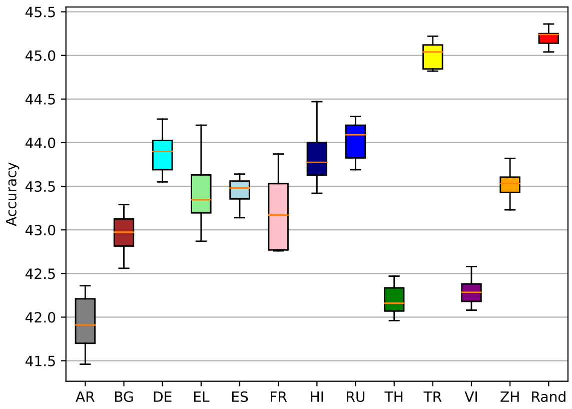

Figure 2: Evaluation results of different strategies of the code-switched method under the 8-shot setting for 15 languages on average.

shows the results of SoftMV with 10 different seeds under the 8-shot setting for 15 languages on average. We can observe that SoftMV performs worst with an accuracy of 42.1% when using AR (Arabic) to replace the words in sentences. When using TR (Turkish) to replace the words in sentences, the performance of SoftMV outperforms the results using another language. The reason is that TR is different from EN, while not too rare like low-resource languages such as UR (Urdu) and AR. Thus the model can better align contextualized representations in different languages into the same semantic space. When randomly selecting languages for the words of each sentence, SoftMV performs best with a lower standard deviation. Therefore, we apply a random strategy for the code-switched method in our experiments.

5.4 Analysis of Soft Prompts

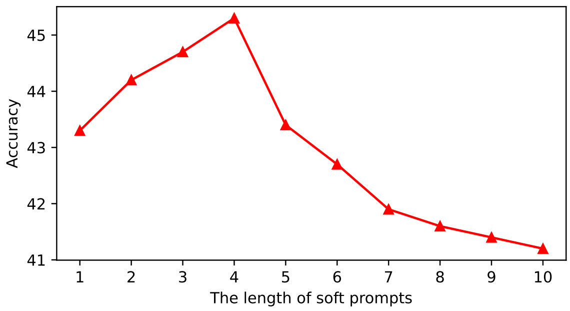

Figure 3: Evaluation results of different lengths of soft prompts under the 8-shot setting for 15 languages on average.

We also conducted experiments to show how the length of soft prompts impacts performance. The results are illustrated in Figure 3 under the 8-shot setting. We can observe that the performance of SoftMV is very sensitive to the value of length. As the length of soft prompts increases, the performance of SoftMV first increases and then decreases. As the length of soft prompts increases, the model has the more expressive power to reduce the gaps across different languages. Therefore, the performance of the model is gradually improved. SoftMV achieves the best performance when the length of soft prompts is 4. When the length is larger than 4, the accuracy decreases sharply. The reason is that the model with longer soft prompts tends to overfit the training data under the few-shot setting.

6 Conclusion

In this paper, we propose a novel **Soft** prompt learning framework with a **M**ultilingual **V**erbalizer (SoftMV) for XNLI. SoftMV applies the code-switched substitution strategy to generate multilingual questions for original questions constructed with soft prompts. We adopt the multilingual verbalizer to align the representations of original and augmented samples into the same semantic space with consistency regularization. Experimental results on XNLI demonstrate that SoftMV significantly outperforms the previous methods under the few-shot and full-shot cross-lingual transfer settings. The detailed analysis further confirms the effectiveness of each component in SoftMV.

7 Limitations

SoftMV is specifically designed for cross-lingual natural language inference. We believe that some of the ideas in our paper can be used in other tasks of XLU, which remains to be further investigated by subsequent research.

In addition, we conduct experiments on the XNLI dataset which consists of 15 languages. SoftMV outperforms the baseline methods under the cross-lingual transfer settings. However, the cross-lingual ability of SoftMV on other languages, especially those lacking relevant datasets, needs to be verified in future work.

Acknowledgements

The work was supported by the National Key Research and Development Program of China (No. 2019YFB1704003), the National Nature Science Foundation of China (No. 62021002), Tsinghua BNRist and Beijing Key Laboratory of Industrial Bigdata System and Application.

References

Wasi Ahmad, Haoran Li, Kai-Wei Chang, and Yashar Mehdad. 2021. Syntax-augmented multilingual BERT for cross-lingual transfer. 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 4538–4554, Online. Association for Computational Linguistics.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

M Saiful Bari, Tasnim Mohiuddin, and Shafiq Joty. 2021. UXLA: A robust unsupervised data augmentation framework for zero-resource cross-lingual NLP. 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 1978–1992, Online. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

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. *Advances in neural information processing systems*, 32.

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.

Xin Dong, Yaxin Zhu, Zuohui Fu, Dongkuan Xu, and Gerard de Melo. 2021. Data augmentation with adversarial training for cross-lingual NLI. 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 5158–5167, Online. Association for Computational Linguistics.

Karl Moritz Hermann and Phil Blunsom. 2014. Multilingual models for compositional distributed semantics. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 58–68, Baltimore, Maryland. Association for Computational Linguistics.

Xuming Hu, Lijie Wen, Yusong Xu, Chenwei Zhang, and Philip Yu. 2020. SelfORE: Self-supervised relational feature learning for open relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3673–3682, Online. Association for Computational Linguistics.

Xuming Hu, Chenwei Zhang, Fukun Ma, Chenyao Liu, Lijie Wen, and Philip S. Yu. 2021. Semi-supervised relation extraction via incremental meta self-training. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 487–496, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

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

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

Shuang Li, Xuming Hu, Li Lin, Aiwei Liu, Lijie Wen, and Philip S. Yu. 2023. A multi-level supervised contrastive learning framework for low-resource natural language inference. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:1771–1783.

Shuang Li, Xuming Hu, Li Lin, and Lijie Wen. 2022. Pair-level supervised contrastive learning for natural language inference. *arXiv preprint arXiv:2201.10927*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. 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 4582–4597, Online. Association for Computational Linguistics.

Li Lin, Yixin Cao, Lifu Huang, Shuang Li, Xuming Hu, Lijie Wen, and Jianmin Wang. 2022. What makes the story forward? inferring commonsense explanations as prompts for future event generation. In *Proc. of SIGIR*, pages 1098–1109.

Aiwei Liu, Honghai Yu, Xuming Hu, Shuang Li, Li Lin, Fukun Ma, Yawen Yang, and Lijie Wen. 2022a. Character-level white-box adversarial attacks against transformers via attachable subwords substitution. In *Proc. of EMNLP*.

Shuliang Liu, Xuming Hu, Chenwei Zhang, Shuang Li, Lijie Wen, and Philip S. Yu. 2022b. Hiure: Hierarchical exemplar contrastive learning for unsupervised relation extraction. In *Proc. of NAACL-HLT*, pages 5970–5980.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022c. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 61–68, Dublin, Ireland. Association for Computational Linguistics.

Bill MacCartney and Christopher D. Manning. 2008. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 521–528, Manchester, UK. Coling 2008 Organizing Committee.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Kunxun Qi, Hai Wan, Jianfeng Du, and Haolan Chen. 2022. Enhancing cross-lingual natural language inference by prompt-learning from cross-lingual templates. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1910–1923, Dublin, Ireland. Association for Computational Linguistics.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying LMs with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5203–5212, Online. Association for Computational Linguistics.

Libo Qin, Minheng Ni, Yue Zhang, and Wanxiang Che. 2021. Cosda-ml: multi-lingual code-switching data augmentation for zero-shot cross-lingual nlp. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3853–3860.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021b. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics.

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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Yusheng Su, Xiaozhi Wang, Yujia Qin, Chi-Min Chan, Yankai Lin, Huadong Wang, Kaiyue Wen, Zhiyuan Liu, Peng Li, Juanzi Li, Lei Hou, Maosong Sun, and Jie Zhou. 2022. On transferability of prompt tuning for natural language processing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3949–3969, Seattle, United States. Association for Computational Linguistics.

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

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Hui Wu and Xiaodong Shi. 2022. Adversarial soft prompt tuning for cross-domain sentiment analysis. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2438–2447, Dublin, Ireland. Association for Computational Linguistics.

Mengjie Zhao and Hinrich Schütze. 2021. Discrete and soft prompting for multilingual models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8547–8555, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Shots	α	lr	Epochs	Weight decay	Batch size
1	0.10	1e-05	70	0.01	12
2	0.10	1e-05	70	0.01	12
4	0.10	1e-05	70	0.01	12
8	0.15	1e-05	70	0.01	12
16	0.20	4e-06	70	0.01	12
32	0.15	7e-06	70	0.01	12
64	0.15	1e-06	70	0.01	12
128	0.20	1e-06	70	0.01	12
256	0.35	1e-06	70	0.01	12
Full	0.30	1e-06	70	0.01	12

Table 5: Hyperparameters used under different settings of XNLI.

A Training Details

A.1 Hyperparameters

Table 5 shows the hyperparameters used under different settings of XNLI. The model is trained for 70 epochs and the checkpoint that performs best on the development set is selected for performance evaluation.

A.2 Computing Device

All experiments are conducted on GeForce GTX 3090Ti. We use batch size 24 for a single GPU. Three GPUs are used for few-shot experiments. The full-shot experiments use 6 GPUs.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 7

☑ A2. Did you discuss any potential risks of your work?
Section 4, Appendix A

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract, 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, Section 5

☑ B1. Did you cite the creators of artifacts you used?
Section 4, Section 5

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Section 4, Section 5

☑ 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)?
Section 4, Appendix 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?
Section 4

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 4

☑ 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.
Section 4

C ☑ Did you run computational experiments?

Section 4, Section 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4, Appendix 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?
Section 4, Appendix 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 5

☑ 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.)?
Section 4, Appendix 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.