Enhancing Cross-lingual Prompting with Two-level Augmentation

Anonymous ACL submission

Abstract

Prompting approaches show promising results in few-shot scenarios. However, its strength for multilingual/cross-lingual problems has not been fully exploited. Zhao and Schütze (2021) made initial explorations in this direction by presenting that cross-lingual prompting outperforms cross-lingual finetuning. In this paper, we first conduct sensitivity analysis on the effect of each component in cross-lingual prompting and derive Universal Prompting across languages. Based on this, we propose a two-level augmentation framework to further improve the performance of prompt-based cross-lingual transfer. Notably, for XNLI, our method achieves 46.54% with only 16 English training examples per class, significantly better than 34.99% of finetuning.

1 Introduction

Although adapting Pre-trained Language Models (PLMs) (Devlin et al., 2019) to downstream NLP tasks via *finetuning* is the de facto mainstream paradigm under fully supervised settings (Wang et al., 2018), *prompting* (Liu et al., 2021; Lester et al., 2021; Radford et al., 2019; Brown et al., 2020) has demonstrated its superiority to *finetuning* in low-resource scenarios (Schick and Schütze, 2021a,b), where the annotated training data is scarce or even not available. Typically, *prompting* reformulates the classification task as a language modeling problem over manually-designed natural language prompts.

Despite the effectiveness of *prompting* on English tasks, its potential for cross-lingual and multilingual problems, which assume the availability of the training data in high-resource languages (e.g., *English*) only, is still under-explored. Zhao and Schütze (2021) is the pioneering work to apply *prompting* to cross-lingual NLP. However, their major efforts are spent on comparing different training strategies for cross-lingual prompting, and how the key ingredients of *prompting*, namely prompt-design and inference strategies, affect the cross-lingual transfer is not discussed.

To provide a practical guide for cross-lingual prompting, we conduct a sensitivity analysis upon Zhao and Schütze (2021) to explore the effects of each *prompting* component on the performance of cross-lingual transfer. Surprisingly, in contrast to the complicated designs in Zhao and Schütze (2021), we find that neither template translation nor verbalizer translation for inference is necessary, and the template-free prompting coupled with English-only inference, dubbed as "Universal Prompting" in this paper, generally performs well across different *few-shot* settings.

Based on such findings, we further propose a two-level augmentation framework to enhance the performance of cross-lingual prompting. Specifically, motivated by the fact that there is no explicit target-language guidance in Universal Prompting, we firstly propose to utilize multilingual verbalizers as an answer augmentation approach. Multilingual verbalizers introduce the label tokens in target languages, which provides additional supervision signals for prompting. By doing so, the model is enforced to learn the association between prompts and semantically equivalent label tokens in multiple languages. Besides, to alleviate the data scarcity issue in few-shot settings, we also develop in-batch data augmentation, which is based on mixup (Zhang et al., 2018; Sun et al., 2020) mechanism, to enhance the training without additional unlabeled data (Xie et al., 2020) or efforts on text manipulation (Wei and Zou, 2019).

In summary, our contributions are as follows:

- We develop a simple yet effective baseline called **Universal Prompting** for cross-lingual prompting.
- Based on Universal Prompting, we further propose a two-level augmentation framework to enhance the performance of prompt-based cross-lingual transfer.

		Prompt Templates	Verbalizers
EN (source)	Zhao and Schütze (2021)	`A . Question: B ? Answer: <mask> .`	Entailment: yes; Contradict: no; Neutral: maybe
	Universal Prompting	`A . B ? <mask> .`	Entailment: yes; Contradict: no; Neutral: maybe
TR (target)	Zhao and Schütze (2021)	`A . Soru: B ? Cevap: <mask> .`	Entailment: Evet; Contradict: hiçbir; Neutral: belki
	w/o Template Translation	`A . Question:B ?Answer: <mask> .`	Entailment: Evet; Contradict: hiçbir; Neutral: belki
	w/o Verbalizer Translation	`A . Soru: B ? Cevap: <mask> .`	Entailment: yes; Contradict: no; Neutral: maybe
	w/o Prompting Words	`A . B ? <mask> .`	Entailment: Evet; Contradict: hiçbir; Neutral: belki
	Universal Prompting	`A . B ? <mask> .`	Entailment: yes; Contradict: no; Neutral: maybe

Table 1: Prompt templates and verbalizers in English (EN) and Turkish (TR). A and B indicate two sentences of a sentence pair. For XNLI, A is the premise and B is the hypothesis. With the proposed Universal Prompting, we could treat source-language training and target-language inference in a unified fashion.

2 Pilot Experiments

In this section, we borrow the proposed solution in Zhao and Schütze (2021) to empirically investigate the elements in cross-lingual prompting. Note that, since soft prompting (SP) and mixed prompting (MP) rely on an external bidirectional LSTM to create soft prompt, we mainly investigate discrete prompting (DP) in this work for a clear and fair comparison.

2.1 Universal Prompting across Languages

Zhao and Schütze (2021) achieved prompt-based cross-lingual transfer by directly utilizing the translated prompting words and verbalizers for target-language inference. However, since the translated prompting words are not seen and the translated verbalizers are never modeled by the PLM during training on English, this may result in discrepancies between the source-language training and the target-language inference.

Starting from the above two aspects that result in such source-target discrepancies, we consider 3 possible variants with design choices different from Zhao and Schütze (2021) to alleviate the discrepancies to a certain degree. By combining these variations we end up with a Universal Prompting design, which can treat individual languages in a unified fashion. Table 1 summarizes our different design choices. [1]

2.2 Results

Our major experimental setup follows Zhao and Schütze (2021). Please refer to Section 4 for more details. In Table 2, we show that by alleviating discrepancies either in the aspect of verbalizers or templates, we could further improve the performance

Shots	Method	Accuracy
16	Zhao and Schütze (2021)	$38.81_{1.61}$
	w/o Template Translation	$39.15_{1.73}$
	w/o Verbalizer Translation	$42.32_{1.81}$
	w/o Pormpting Words	$39.87_{2.94}$
	Universal Prompting	$\mathbf{43.18_{2.77}}$
32	Zhao and Schütze (2021)	$41.42_{1.66}$
	w/o Template Translation	$41.72_{1.89}$
	w/o Verbalizer Translation	$46.50_{1.54}$
	w/o Pormpting Words	$43.66_{0.96}$
	Universal Prompting	$\mathbf{48.26_{1.34}}$
64	Zhao and Schütze (2021)	$46.42_{0.65}$
	w/o Template Translation	$46.75_{0.61}$
	w/o Verbalizer Translation	$\mathbf{53.07_{1.33}}$
	w/o Pormpting Words	$47.60_{1.09}$
	Universal Prompting	$52.19_{1.53}$

Table 2: The comparison results between Zhao and Schütze (2021) and its variants on XNLI. We calculate the average accuracy over 15 languages. The standard deviation over 5 runs is reported as the subscript.

of cross-lingual prompting[2]. Our proposed Universal Prompting across languages alleviates the discrepancy of prompt templates and verbalizers simultaneously, yielding a much stronger baseline than Zhao and Schütze (2021). This indicates that a null prompt (IV et al., 2021), combined with the English verbalizer for target-language inference generally performs well in multilingual tasks. We refer to this design as Universal Prompting (UP) in the following parts of our paper.

3 Method

Mask token in prompting methods is directly used for inference. In this section, we formalize our two-level augmentation approach for this important element of prompting. Our method leverages answer-level multilingual verbalizers and representation-level mixup simultaneously.

3.1 Answer-level Multilingual Verbalizers

The derived UP only considers the English verbalizer for source language training, and the translated

[1] Note that **w/o verbalizer translation** refers to not applying translated verbalizers during *inference*. In Section 3 we will show how to exploit the translated verbalizers as answer-level augmentation during *training*.

[2] As we employ a different evaluation method, the reproduced results of Zhao and Schütze (2021) are slightly different from the original ones. More details can be found in Section 4.

verbalizers in target languages are not exploited. Intuitively, their rich semantics could serve as high-quality paraphrases (Jiang et al., 2021) of the English verbalizer and provide additional supervision for training multilingual models. Motivated by this, we define a multilingual verbalizer for the English training data, which can be regarded as answer-level augmentation for masked language modeling. Formally, given the pre-built prompt $\boldsymbol{x}$ filled with input sentences, the training objective is to maximize the likelihood of verbalized label tokens in multiple languages:

$$\arg\max_{\boldsymbol{\theta}} \sum_{\boldsymbol{x}} \frac{1}{|\mathcal{L}|} \sum_{\ell \in \mathcal{L}} \log P\big(\langle\text{mask}\rangle = V_\ell(\boldsymbol{y}) | \boldsymbol{x}; \boldsymbol{\theta}\big) \quad (1)$$

where θ denotes parameters of the PLM. V_ℓ is the verbalizer in a certain language $\ell \in \mathcal{L}$, and it maps from the gold label to a specific word in language ℓ. [3] In comparison, UP only takes $\mathcal{L} = \{\text{EN}\}$, which is a monolingual verbalizer.

3.2 Representation-level Mixup

Manifold mixup (Verma et al., 2019) performs the interpolation in the latent space to construct virtual labeled data as augmentation. Based on manifold mixup, several mixup strategies have been designed to boost the performance of NLP tasks (Chen et al., 2020; Sun et al., 2020; Zhang and Vaidya, 2021). In this work, we propose to use mixup for cross-lingual prompting as a representation-level augmentation approach. To the best of our knowledge, this is the first endeavor to enhance prompting and multilingual learning with mixup. To formalize, let $\boldsymbol{m}_i = h(\boldsymbol{x}_i)$ and $\boldsymbol{m}_j = h(\boldsymbol{x}_j)$ as the last transformer layer's encoding of the mask tokens of two prompts $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, respectively. Then we perform linear interpolation to produce a virtual representation:

$$\hat{\boldsymbol{m}}_{ij} = \lambda h(\boldsymbol{x}_i) + (1-\lambda) h(\boldsymbol{x}_j) \quad (2)$$

where $\lambda \sim \beta(\alpha, \alpha)$. The corresponding target labels are linearly interpolated as well by:

$$\hat{\boldsymbol{y}}_{ij} = \lambda \boldsymbol{y}_i + (1-\lambda) \boldsymbol{y}_j \quad (3)$$

Considering an augmented multilingual verbalizer as in Section 3.1, the training objective of this particular virtual example would be:

$$\begin{aligned}\arg\max_{\boldsymbol{\theta}} \frac{1}{|\mathcal{L}|} \sum_{\ell \in \mathcal{L}} \big\{ &\lambda \log P\big(\langle\text{mask}\rangle = V_\ell(\boldsymbol{y}_i) | \hat{\boldsymbol{m}}_{ij}; \boldsymbol{\theta}\big) \\ &+ (1-\lambda) \log P\big(\langle\text{mask}\rangle = V_\ell(\boldsymbol{y}_j) | \hat{\boldsymbol{m}}_{ij}; \boldsymbol{\theta}\big) \big\}\end{aligned} \quad (4)$$

[3] For full lists of language sets we use, please refer to Appendix. A

The interpolation is performed in a dynamic in-batch fashion. For a batch drawn from the training set, we use every two adjacent examples to generate a virtual mask token representation. For more discussion about the mixup strategy for prompting methods, please refer to Appendix. B.

4 Experiments

In this section, we evaluate two multilingual tasks to demonstrate the effectiveness of our two-level augmentation framework.

4.1 Setup

Datasets We conduct experiments on two sentence-pair classification tasks: XNLI (Conneau et al., 2018; Williams et al., 2018) for cross-lingual natural language inference and PAWS-X (Yang et al., 2019) for multilingual paraphrase identification. For these two datasets, while the evaluation data is human-translated, the golden training data is only available in English.

Evaluation Following Zhao and Schütze (2021), we conduct our experiments by training the XLM-R base model (Conneau et al., 2020) on English. Then the model will be directly applied to other target languages, without using any training examples of the target language. To make a reasonable comparison between finetuning and prompting, we ensure finetuning to be better than a random guess on each language. Therefore, we randomly sample without replacement $K \in \{16, 32, 64, 128, 256\}$ per class for XNLI and $K \in \{256, 512\}$ per class for PAWS-X to construct the training set. Then we use the same number of shots from the development split to perform model selection to simulate a realistic few-shot setting (Perez et al., 2021).

The evaluation of few-shot cross-lingual transfer could be with large variance and depend on the selection of few shots (Zhang et al., 2021; Zhao et al., 2021; Keung et al., 2020). In our work, to faithfully reflect the performance of few-shot learning, we do not follow Zhao and Schütze (2021) to fix the training/development data but randomly sample separate training/development sets for different runs.

4.2 Results

Table 3 and 4 presents the accuracy on XNLI and PAWS-X dataset, respectively.

UP v.s. Finetuning On the XNLI dataset, even the simplest prompting method for cross-lingual transfer, namely UP, consistently outperforms the

	Method	EN	AR	BG	DE	EL	ES	FR	HI	RU	SW	TH	TR	UR	VI	ZH	Avg.
16shots	FT	35.62	35.11	34.85	35.07	35.08	35.21	34.95	34.89	34.52	35.07	34.92	34.79	35.02	35.02	34.71	34.99
	UP	47.68	42.01	45.50	44.51	46.68	36.61	46.81	40.29	45.43	42.06	44.21	41.04	40.61	45.79	38.42	43.18
	OURS	48.55	**46.24**	**47.95**	**48.00**	47.41	**47.47**	**48.61**	**44.36**	46.76	44.35	45.95	**45.83**	**44.80**	**47.31**	**44.55**	**46.54**
	W/O MV	**49.54**	41.55	46.84	45.53	47.59	34.63	48.55	42.39	**47.18**	43.95	**46.37**	43.82	43.32	46.52	40.09	44.52
	W/O MIXUP	48.38	45.59	47.74	47.72	**47.60**	44.38	47.83	42.44	46.69	**44.38**	44.65	45.52	43.48	46.65	40.83	45.59
32shots	FT	37.62	36.82	36.61	37.03	37.07	37.39	37.53	37.35	36.83	36.42	36.40	36.40	36.71	36.84	36.96	36.93
	UP	53.33	47.70	50.87	49.74	51.41	41.48	51.09	44.97	50.11	46.76	49.50	45.92	45.64	51.00	44.33	48.26
	OURS	52.79	**49.37**	**51.48**	**50.84**	**51.78**	50.05	**51.77**	**48.08**	**50.46**	47.30	49.35	**50.14**	**47.44**	**50.84**	**48.25**	**49.99**
	W/O MV	**53.75**	48.42	50.71	50.57	51.76	41.98	51.54	45.64	**50.46**	45.84	**49.65**	47.42	45.58	50.56	47.54	48.76
	W/O MIXUP	52.38	49.29	51.39	50.76	51.60	**50.21**	51.54	47.57	50.35	**47.56**	49.07	49.56	47.02	50.65	46.24	49.68
64shots	FT	42.97	40.70	41.29	41.68	42.09	42.46	42.23	40.59	40.38	39.96	40.65	40.84	40.24	42.09	40.53	41.25
	UP	57.76	51.67	54.85	54.99	54.69	51.63	54.96	47.97	53.32	48.12	51.91	49.89	47.86	54.14	49.13	52.19
	OURS	**59.97**	53.18	56.51	**56.67**	55.63	**56.79**	**56.97**	51.77	55.46	50.71	53.35	**54.21**	50.76	56.05	53.09	54.74
	W/O MV	59.17	**53.79**	**56.95**	56.53	**56.18**	55.35	56.48	**52.17**	**55.72**	**50.89**	**54.55**	53.35	**51.62**	**56.43**	**54.42**	**54.91**
	W/O MIXUP	59.56	53.06	55.98	55.65	55.16	56.67	56.66	51.44	55.18	49.99	52.90	53.76	49.80	55.43	53.70	54.33
128shots	FT	47.24	43.91	44.13	43.96	44.38	45.25	44.48	42.38	42.81	42.87	42.87	42.93	42.36	44.60	42.87	43.80
	UP	60.08	51.31	56.60	55.10	56.17	51.25	56.97	49.62	55.18	48.71	53.87	50.42	49.20	55.03	53.15	53.51
	OURS	**62.57**	54.91	58.72	**58.81**	**58.25**	**59.47**	58.76	**52.93**	**57.35**	50.95	54.30	**54.94**	51.47	57.80	54.99	**56.42**
	W/O MV	61.51	**55.31**	58.67	58.15	58.12	58.10	58.42	52.31	56.99	50.80	**55.40**	53.88	**51.74**	**57.96**	**56.12**	56.23
	W/O MIXUP	61.84	54.59	**58.77**	58.57	57.77	59.13	**58.89**	52.70	56.99	**52.05**	54.15	54.69	51.31	57.27	55.59	56.29
256shots	FT	59.49	52.87	55.92	55.51	55.07	57.44	56.32	51.75	54.19	49.88	52.38	53.68	50.38	55.37	53.95	54.28
	UP	65.08	56.57	61.03	60.65	60.74	59.21	61.01	55.18	59.41	53.73	57.66	57.62	54.08	60.58	58.71	58.75
	OURS	**67.97**	**59.54**	**63.59**	**63.26**	**62.34**	**64.80**	**63.93**	**58.39**	**61.87**	**55.83**	**59.19**	**60.32**	**56.00**	**62.41**	**61.29**	**61.38**
	W/O MV	65.80	58.07	62.04	61.33	61.05	63.03	62.36	56.16	60.14	54.17	58.23	57.62	54.12	60.52	59.81	59.63
	W/O MIXUP	67.40	58.02	62.33	62.18	61.35	63.61	62.93	56.89	60.75	54.68	58.06	59.00	54.74	61.17	59.33	60.16

Table 3: Zero-shot cross-lingual transfer accuracy on XNLI. FT:finetuning; UP: Universal Prompting; MV: multilingual verbalizer. Reported results are averaged with 5 random seeds.

	Method	EN	DE	ES	FR	JA	KO	ZH	Avg.
256shots	FT	63.18	60.81	60.95	61.39	58.60	58.48	59.78	60.46
	UP	65.50	62.21	63.24	62.82	54.11	54.30	55.99	59.74
	OURS	**71.87**	**68.59**	**69.10**	**69.02**	**60.41**	**60.88**	**62.75**	**66.09**
	W/O MV	69.06	66.26	66.47	65.79	59.28	58.34	60.77	63.71
	W/O MIXUP	70.95	67.14	67.58	67.63	59.01	60.44	61.16	64.84
512shots	FT	77.64	73.41	73.19	74.33	65.55	65.19	68.25	71.08
	UP	83.31	76.18	77.63	77.42	63.41	65.03	68.06	73.01
	OURS	**84.97**	**78.63**	79.60	**80.48**	**67.86**	68.13	**72.34**	**76.00**
	W/O MV	84.81	78.56	**79.67**	79.64	67.04	**68.34**	71.50	75.65
	W/O MIXUP	84.84	77.85	79.36	79.69	66.76	68.03	71.03	75.37

Table 4: Zero-shot cross-lingual transfer accuracy on PAWS-X. FT:finetuning; UP: Universal Prompting; MV: multilingual verbalizer. Reported results are averaged with 5 random seeds.

finetuning (FT) method by a large margin. Besides, our UP also surpasses FT on the majority of languages on the more challenging PAWS-X. These observations suggest that prompting is indeed a better solution of few-shot learning in cross-language scenarios and our UP can serve as a strong baseline of cross-lingual prompting.

Two-level Augmentation With the proposed two-level augmentation framework, our prompting method achieves consistent improvement over UP, indicating that multilingual verbalizers as answer-level augmentation and representation-level mixup are two meaningful ways to enhance cross-lingual prompting. The comparison results in Table 3 and Table 4 also exhibit consistent superiority of our method over cross-lingual finetuning. Even in the most resource-rich settings, compared to FT, our method still obtains 7.1% (256 shots) and 4.9% (512 shots) absolute gains on XNLI and PAWS-X.

Ablation Study The performance of our prompting method will become worse when we remove representation-level mixup or multilingual verbalizer, showing that both of the augmentation strategies defined at representation-level and answer-level contribute positively to the improvement. We also notice that the negative effects brought by OURS W/O MV are generally larger, showing that the guidance from multiple target languages is more helpful for cross-lingual prompting.

Inference Strategy Our augmentation framework can be naturally extended by designing more sophisticated inference strategies. Interestingly, we find that English-only inference is still comparable to these strategies. More discussions can be found in Appendix C.

5 Conclusion

In this paper, we first derive Universal Prompting, a simple but effective baseline for cross-lingual prompting. The proposed two-level augmentation framework further enhance cross-lingual prompting on two sentence-pair classification tasks. In the future, we will consider verifying the effectiveness of prompting and the proposed augmentation framework in cross-lingual sequence tagging or text generation tasks.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris 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 *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel 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.

Robert L Logan IV, Ivana Balavzevi'c, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. 2021. Cutting down on prompts and parameters: Simple few-shot learning with language models. *ArXiv*, abs/2106.13353.

Zhengbao Jiang, J. Araki, Haibo Ding, and Graham Neubig. 2021. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

Amit Jindal, Arijit Ghosh Chowdhury, Aniket Didolkar, Di Jin, Ramit Sawhney, and Rajiv Ratn Shah. 2020. Augmenting NLP models using latent feature interpolations. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6931–6936, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Phillip Keung, Yichao Lu, Julian Salazar, and Vikas Bhardwaj. 2020. Don't use English dev: On the zero-shot cross-lingual evaluation of contextual embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 549–554, Online. Association for Computational Linguistics.

Teven Le Scao and Alexander Rush. 2021. How many data points is a prompt worth? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2627–2636, Online. Association for Computational Linguistics.

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

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ArXiv*, abs/2107.13586.

Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2021. On the stability of fine-tuning {bert}: Misconceptions, explanations, and strong baselines. In *International Conference on Learning Representations*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. *ArXiv*, abs/2105.11447.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

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.

Lichao Sun, Congying Xia, Wenpeng Yin, Tingting Liang, Philip Yu, and Lifang He. 2020. Mixup-transformer: Dynamic data augmentation for NLP tasks. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3436–3440, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. 2019. Manifold mixup: Better representations by interpolating hidden states. In *ICML*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. 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, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3687–3692, Hong Kong, China. Association for Computational Linguistics.

Hongyi Zhang, Moustapha Cissé, Yann Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. *ArXiv*, abs/1710.09412.

Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q Weinberger, and Yoav Artzi. 2021. Revisiting few-sample {bert} fine-tuning. In *International Conference on Learning Representations*.

Wancong Zhang and I. Vaidya. 2021. Mixup training leads to reduced overfitting and improved calibration for the transformer architecture. *ArXiv*, abs/2102.11402.

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.

Mengjie Zhao, Yi Zhu, Ehsan Shareghi, Ivan Vulić, Roi Reichart, Anna Korhonen, and Hinrich Schütze. 2021. A closer look at few-shot crosslingual transfer: The choice of shots matters. 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 5751–5767, Online. Association for Computational Linguistics.

A Additional Implementation Details

Implementation Package Our implementation is based on PyTorch (Paszke et al., 2019) and Huggingface Transformer (Wolf et al., 2019) framework.

Model Details XLM-R base model, containing 270M parameters, is pretrained on 2.5TB of filtered CommonCrawl on 100 languages. It contains 12 Transformer layers with hidden space dimensions of 768 and 12 attention heads in each layer.

Computing Infrastructure All of our experiments are conducted on a single *Tesla V100-SXM2 32G*. Gradient accumulation steps of 4 is used for prompting to overcome resource limitations.

Hyperparameter Settings Our major hyperparameter settings follow Zhao and Schütze (2021). A fixed learning rate (1e-5) is used for all of our experiments without any learning rate schedule to compare finetuning with prompting (Le Scao and Rush, 2021). We use a smaller batch size of 8 for finetuning and prompting because it achieves slightly better performance. We use the max sequence length of 256. The model is trained for 50 epochs and we select the checkpoint by development accuracy for testing as suggested in Mosbach et al. (2021); Zhang et al. (2021). The α value for β distribution in representation-level mixup is set to 1.2 for all of the experiments.

Prompting The language sets $\mathcal{L}$ used for multilingual verbalizers are determined by the language availability of the dataset. Specifically, for XNLI, $\mathcal{L}$ = {EN, AR, BG, DE, EL, ES, FR, HI, RU, SW, TH, TR, UR, VI, ZH}. For PAWS-X, $\mathcal{L}$ = {EN, DE, ES, FR, JA, KO, ZH}

For simplicity, the verbalizers of target languages are translated by Google Translate. Similar with XNLI, we use "paraphrase $\rightarrow$ yes" and "non-paraphrase $\rightarrow$ no" as the verbalizer of PAWS-X in English. Table 5 presents the full multilingual verbalizer we use for the PAWS-X dataset.

We discuss Universal Prompting across languages for multilingual sentence-pair classification tasks in Section 2. Moreover, we believe the same notion of alleviating source-target discrepancies in terms of prompt template and verbalizer is also applicable for cross-lingual single-sentence classification or text generation tasks, which is left for future work.

Language	Verbalizer
EN	Paraphrase $\rightarrow$ yes Non-paraphrase $\rightarrow$ no
DE	Paraphrase $\rightarrow$ Ja Non-paraphrase $\rightarrow$ Nein
ES	Paraphrase $\rightarrow$ sí Non-paraphrase $\rightarrow$ no
FR	Paraphrase $\rightarrow$ Oui Non-paraphrase $\rightarrow$ non
JA	Paraphrase $\rightarrow$ はい Non-paraphrase $\rightarrow$ ない
ZH	Paraphrase $\rightarrow$ 是 Non-paraphrase $\rightarrow$ 否
KO	Paraphrase $\rightarrow$ 예 Non-paraphrase $\rightarrow$ 아니

Table 5: The multilingual verbalizer for PAWS-X.

B Additional Discussion about Mixup Strategy for Prompting

Previous mixup methods for NLP models perform the interpolation at the input embedding level (Zhang and Vaidya, 2021), hidden representation level (Jindal et al., 2020; Chen et al., 2020) or the [CLS] token (Zhang and Vaidya, 2021). However, none of them is directly applicable for prompting-based methods. In prompting-based methods, the most important hidden space representation for classification is encoded at the position of mask tokens. Different training data may have different sequence lengths and their mask tokens may be put at different positions. Previous practices of hidden representation level mixup will result in the interpolation between the representation of a mask token and a normal token, which is meaningless in prompting methods. Therefore, we find that the most intuitive way is to apply the interpolation in the last transformer layer's representations of mask tokens. Then the interpolated representation is fed into the masked language modeling head.

Note that our proposed representation-level mixup of mask tokens is also directly applicable for monolingual prompting. It would also be interesting to apply it to more settings, which is left for future study.

C Inference Strategy

A natural extension for our method is to leverage the multilingual verbalizer in some way for target-language inference as well. For comparisons, we heuristically devise the following inference strategies :

(1) English Verbalizer The English verbalizer is still used when transferring to target languages.

Strategy Num	Accuracy
1	$\mathbf{56.42}_{\mathbf{1.37}}$
2	$56.31_{1.15}$
3	$56.23_{1.09}$
4	$56.33_{1.11}$
5	$56.39_{1.21}$

Table 6: Test accuracy by using different inference strategies. The accuracy is averaged by 15 testing languages of XNLI of 5 random seeds.

This strategy is used to produce results in Table 3 and 4. To formalize:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} P\big(\langle\text{mask}\rangle = V_{EN}(\boldsymbol{y})|\boldsymbol{x};\boldsymbol{\theta}\big) \tag{5}$$

(2) Target Language Verbalizer The verbalizer in the corresponding target language is used, which is the practice of Zhao and Schütze (2021). To formalize:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} P\big(\langle\text{mask}\rangle = V_{target}(\boldsymbol{y})|\boldsymbol{x};\boldsymbol{\theta}\big) \tag{6}$$

(3) Taking Maximum over the Multilingual Verbalizer In this strategy, we will take the maximum probability over the whole multilingual verbalizer. To formalize:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y},\boldsymbol{\ell}} P\big(\langle\text{mask}\rangle = V_{\ell}(\boldsymbol{y})|\boldsymbol{x};\boldsymbol{\theta}\big) \tag{7}$$

(4) Taking Sum over the Multilingual Verbalizer In this strategy, we will take the sum of probability over the whole multilingual verbalizer. To formalize:

$$\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}} \sum_{\boldsymbol{\ell}\in\mathcal{L}} P\big(\langle\text{mask}\rangle = V_{\ell}(\boldsymbol{y})|\boldsymbol{x};\boldsymbol{\theta}\big) \tag{8}$$

(5) Bilingual Verbalizer In this strategy, we will take the sum of probability over the target language verbalizer and the English verbalizer. To formalize, the predicted label $\hat{y}$ is given by:

$$\begin{aligned}\hat{\boldsymbol{y}} = \arg\max_{\boldsymbol{y}}\{&P\big(\langle\text{mask}\rangle = V_{EN}(\boldsymbol{y})|\boldsymbol{x};\boldsymbol{\theta}\big)\\ &+P\big(\langle\text{mask}\rangle = V_{target}(\boldsymbol{y})|\boldsymbol{x};\boldsymbol{\theta}\big)\}\end{aligned} \tag{9}$$

We use the checkpoint of XLM-R trained by 128 shots on the XNLI dataset and make inference with different strategies. Table 6 shows the accuracy by employing different inference strategies. We show that with our two-level augmentation framework, the inference is quite robust to the utilization of the verbalizer. This can probably be attributed to answer-level multilingual verbalizers, which help to model label tokens in multiple languages. We choose to simply employ English-only inference due to its simplicity and slightly better performance to produce results in Tables 3 and 4.