# Respond in my Language: Mitigating Language Inconsistency in Response Generation based on Large Language Models

**Liang Zhang**[1]*, **Qin Jin**[1]†, **Haoyang Huang**[2], **Dongdong Zhang**[2], **Furu Wei**[2]
[1]School of Information, Renmin University of China
[2]Microsoft Research Asia, China
{zhangliang00,qjin}@ruc.edu.cn
{haohua,dozhang,fuwei}@microsoft.com

## Abstract

Large Language Models (LLMs) show strong instruction understanding ability across multiple languages. However, they are easily biased towards English in instruction tuning, and generate English responses even given non-English instructions. In this paper, we investigate the language inconsistent generation problem in monolingual instruction tuning. We find that instruction tuning in English increases the models' preference for English responses. It attaches higher probabilities to English responses than to responses in the same language as the instruction. Based on the findings, we alleviate the language inconsistent generation problem by counteracting the model preference for English responses in both the training and inference stages. Specifically, we propose Pseudo-Inconsistent Penalization (PIP) which prevents the model from generating English responses when given non-English language prompts during training, and Prior Enhanced Decoding (PED) which improves the language-consistent prior by leveraging the untuned base language model. Experimental results show that our two methods significantly improve the language consistency of the model without requiring any multilingual data[1].

## 1 Introduction

Large Language Models (LLMs) have received increasing research attention for their convincing language understanding and generation abilities (Brown et al., 2020; Openai, 2022; OpenAI, 2023; Touvron et al., 2023a,b). They also demonstrate intrinsic capabilities of multilingual understanding (Armengol-Estapé et al., 2022; Yuan et al., 2023) and cross-task generalization after instruction tuning (Ouyang et al., 2022). However, it is observed that LLMs tend to generate in the wrong language after monolingual instruction tuning (Hu et al., 2023). As shown in Figure 1, Llama2-Chat, which is an instruction-tuned model of Llama2 (Touvron et al., 2023b) in English, struggles to generate responses in the same/consistent language with the user instruction. To make LLMs follow multilingual instructions, many works (Li et al., 2023b; Chen et al., 2023c,b; Ranaldi and Pucci, 2023; Lai et al., 2023) have attempted to perform instruction tuning over multiple languages. Despite their achievements, these approaches can be costly to implement, since they involve collecting multilingual instruction-tuning data and retraining the models over larger datasets (Chen et al., 2023b).


Figure 1: Llama2-Chat-7B fails to respond to the user in the consistent language.

However, as shown in Figure 1, the model retains a notable degree of multilingual understanding even after monolingual instruction tuning, which suggests that the model should have the potential to follow multilingual instructions. Therefore, instead of leveraging multilingual resources for fine-tuning, we focus on resolving the problem of language inconsistency in response generation caused by monolingual instruction tuning, thus enabling

*Work done during an internship at MSRA.
†Corresponding author.
[1]https://github.com/zhangliang-04/Respond_in_my_language



*Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4177–4192

the model to follow multilingual instructions.

To this end, we delve into the process of English instruction tuning, and find that this monolingual training process increases the model's prior probability of English responses. As a result, the model tends to assign a higher probability to English responses regardless of the instruction language. Based on this observation, we propose two distinct methods, adopted during the training or inference stage respectively, to control the model's preference for English responses. Specifically, we propose an auxiliary training objective **P**seudo-**I**nconsistent **P**enalization (PIP) that penalizes the model from generating English responses for pseudo-inconsistent instructions. In the inference stage, we propose **P**rior **E**nhanced **D**ecoding (PED) that increases the language consistent probability from the untuned base model. We evaluate the proposed methods across multilingual instruction following, language understanding and machine translation. Experimental results show that our proposed methods significantly increase the rate of generating language-consistent responses without hurting the ability of language understanding and machine translation.

The main contributions of this work include:

- We analyze the language inconsistency problem in response generation by large language models after they are instruction-tuned.
- We propose solutions to alleviate this problem from both training and inference perspectives. Neither method requires additional multilingual-instruction following data.
- We validate the effectiveness of our proposed methods from multiple aspects including instruction following, language understanding, and machine translation.

## 2 Related Works

**Multilingual Instruction Tuning** Many works try to extend the English capabilities of large language models to other languages through multilingual instruction tuning (Chen et al., 2023c; Li et al., 2023b; Chen et al., 2023b). Early efforts focused on collecting and constructing multilingual instruction-following data. For instance, Phoenix (Chen et al., 2023c) gathers existing multilingual assets from various sources (Dom and Steven, 2023; Peng et al., 2023). Bactrian-X (Li et al., 2023b) obtain multilingual responses by providing ChatGPT (Openai, 2022) with machine-translated instructions. Chen et al. (2023b) explore multilingual instruction tuning under a budget-constraint scenario. Okapi (Lai et al., 2023) introduces RLHF (Ouyang et al., 2022) in multilingual instruction tuning to align with human preference. Later studies attempt to introduce additional training objectives to enhance the multilingual alignment in LLMs. X-LLM (Ranaldi and Pucci, 2023) and x-CrossLlama (Zhu et al., 2023) construct translation-based instructions with parallel corpora (Goyal et al., 2022; Schwenk et al., 2021). Li et al. (2023a) enhance multilingual correspondence through multilingual contrastive learning. PLUG (Zhang et al., 2023) employs English as a pivot language to improve multilingual instruction following. These works require multilingual data to achieve multilingual instruction following. In contrast, we focus on addressing the language inconsistency in monolingual instruction tuning, and can enable multilingual instruction-following without the need for multilingual data.

**Off-target Problem** A closely related topic to language inconsistency is the off-target problem, where multilingual machine translation models generate translations in the wrong language, disregarding the control signal (Ha et al., 2016; Gu et al., 2019; Aharoni et al., 2019; Rios et al., 2020; Zhang et al., 2020; Wu et al., 2021; Yang et al., 2021). Researches have been conducted to explain and solve the off-target problem. For instance, Gu et al. (2019) consider that the model learns spurious correlations between the control signal and decoded sentences. Chen et al. (2023a) suggest that the off-target problem stems from the encoder's failure to capture discriminative control signal and alleviate it by separating shared tokens across languages. Zan et al. (2023) impose unlikelihood sampling on constructed off-target samples to prevent the model from generating in the wrong languages. Sennrich et al. (2023) use contrastive decoding (Li et al., 2023c) to alleviate hallucination and off-target problems in machine translation. The key difference between language inconsistency and the off-target problem is that language inconsistency refers to the model's failure to maintain the language of the input instruction without explicit specification. We focus on generating responses in consistent language without specifically identifying them in this paper.

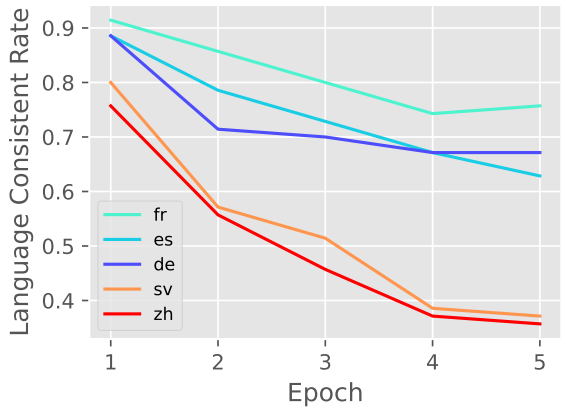



Figure 2: The language consistent rate of other languages during instruction tuning on English Alpaca. Base model: Llama2-7B.

**Zero-shot Cross-lingual Generation** Zero-shot cross-lingual generation is another related topic to our study that concentrates on knowledge transferring across languages in generation tasks. It has been widely observed that zero-shot cross-lingual generation encounters challenges due to language inconsistent generation (Xue et al., 2021; Maurya et al., 2021; Vu et al., 2022; Pfeiffer et al., 2023; Li and Murray, 2023; Chirkova et al., 2023). Research efforts have been made to address the issue. For example, Maurya et al. (2021) and Vu et al. (2022) attribute this issue as catastrophic forgetting and alleviate it by freezing model components and parameter-efficient prompt tuning (Lester et al., 2021). Li and Murray (2023) find that monolingual training encourages the encoder to learn language-invariant representations which are detrimental to cross-lingual generation. They suggest that incorporating an auxiliary language can help regularize the model and mitigate this issue. Chirkova et al. (2023) conducts experimental studies and suggests that a reduced learning rate can be beneficial for cross-lingual generation. These studies are conducted based on encoder-decoder models such as mT5 (Xue et al., 2021) and NLLB (Team et al., 2022), lacking analysis of the current mainstream decoder-only large language models (Brown et al., 2020; Touvron et al., 2023a,b). In addition, their evaluation tasks contain only short multilingual sentences such as multilingual QA and summarization. In contrast, we focus on multilingual instruction following under the zero-shot setting, which presents a greater challenge in generating consistent and long-form text in multiple languages.

## 3 Method

### 3.1 Instruction tuning

Instruction tuning guides large language models to follow human instructions in a supervised manner (Ouyang et al., 2022). Formally, given an instruction following dataset $D = \{(X_i, Y_i)\}_{i=1}^{N}$, the loss function of instruction tuning is as follows:

$$\mathcal{L}_{\text{it}} = -\frac{1}{N}\sum_{i} \log P(Y_i|X_i;\theta) \quad (1)$$

where $X_i, Y_i$ refer to the instruction and response respectively, and $\theta$ denotes the parameters of the model, which is omitted by default for convenience of expression. By minimizing $\mathcal{L}_{\text{it}}$, the probability of generating corresponding response $P(Y|X)$ gets higher than that of a mismatched response $P(\widetilde{Y}|X)$, and the model thus learns to generate a proper response.

### 3.2 Language Consistency in Instruction Tuning

Intuitively, we expect the generated responses should be in the same language as the input instructions. For example, given an instruction $X^l$ in a non-English language $l$, we expect the model to generate the response $Y^l$ rather than $Y^{en}$ in English, though $Y^l$ and $Y^{en}$ are semantically similar. However, we find that the language consistency of other languages can be compromised during English instruction tuning. As illustrated in Figure 2, the ratio of generating language-consistent responses keeps decreasing during the English instruction tuning. It indicates that the model gradually loses the ability to generate responses in consistent languages, which is not what we expect.

To understand why this could happen during English instruction tuning, we examine the process of training from a Bayesian perspective. Supposing the model is trained on English instruction tuning dataset $D^{en} = \{(X_i^{en}, Y_i^{en})\}_{i=1}^{N}$. By minimizing the loss in Equation (1), the model is optimized to generate an appropriate response and increase $P(Y^{en}|X^{en})$. According to the Bayesian rule, $P(Y^{en}|X^{en})$ can be formulated as:

$$P(Y^{en}|X^{en}) = \frac{P(X^{en}|Y^{en})P(Y^{en})}{P(X^{en})} \quad (2)$$

From Equation (2), we can observe that the increase of $P(Y^{en}|X^{en})$ will potentially lead to the increase of $P(Y^{en})$, which is the prior probability

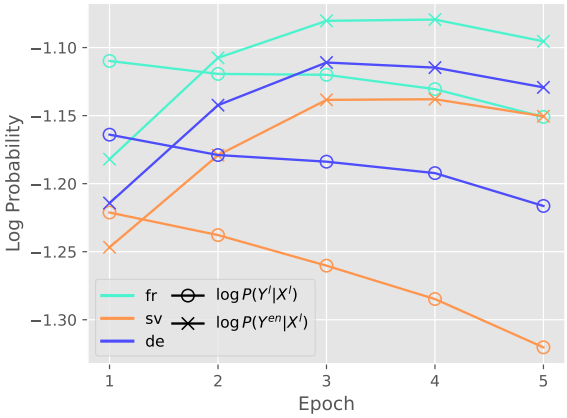



Figure 3: The log probability of Alpaca responses in different languages. Probability of English responses $P(Y^{en}|X^l)$ (×) becomes greater than non-English responses $P(Y^l|X^l)$ (○) during the instruction tuning progresses. We aggregate the response level probability by averaging on tokens: $\log P(Y|X) = \frac{1}{N}\sum_i^N \log P(y_i|X)$

of generating English response $Y^{en}$ regardless of the input instruction. Compared to $Y^{en}$, $Y^l$ is not encouraged since the model is not optimized on instruction data in language $l$. Thus, the increase of $P(Y^{en})$ could lead to $P(Y^{en}|X^l)$ becoming greater than $P(Y^l|X^l)$. It means the model grants more likelihood towards English responses even given instructions in language $l$.

To verify this hypothesis, we plot the change curve of $P(Y^{en}|X^l)$ and $P(Y^l|X^l)$ during the training process. As illustrated in Figure 3, the probability of language-inconsistent response $P(Y^{en}|X^l)$ keeps increasing as the instruction tuning proceeds, and becomes greater than the probability of language-consistent response $P(Y^l|X^l)$. It suggests that the instruction tuning process indeed leads to encouraging the generation of English responses for non-English instruction.

### 3.3 Proposed Methods

To address the above language inconsistency issue, we need to make sure that the model favors responding using the same language as the instruction, that is, to ensure $P(Y^l|X^l) > P(Y^{en}|X^l)$. To achieve this, we propose the Pseudo Inconsistent Penalty (PIP) that penalizes $P(Y^{en}|X^l)$ during English instruction tuning, and Prior Enhanced Decoding (PED) that increases $P(Y^l|X^l)$ in the inference phrase.

**Pseudo Inconsistent Penalization (PIP)** aims to offset the preference for English response learned in instruction tuning. A direct approach is to penalize the occurrence of $Y^{en}$ given $X^l$. However, $X^l$ is not available in the monolingual setting. To replace $X^l$, we construct pseudo inconsistent instructions by adding language identifier prompt $R^l$ to the English instructions, where $R^l$ conveys the semantic of "response in language $l$". It is based on the intuition that we expect the model to respond in the specific language when it is instructed to. We use maximum unlikelihood estimation (Welleck et al., 2020) to penalize the English response $Y^{en}$ in the dataset given the pseudo inconsistent instruction $(X_i^{en}, R^l)$. Formally, the loss function of PIP is as follows:

$$\mathcal{L}_{\text{pip}} = -\frac{1}{N}\sum_i \log(1 - P(Y_i^{en}|X_i^{en}, R^l)) \quad (3)$$

We perform PIP along with the English instruction tuning. The total loss is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{it}} + \mathcal{L}_{\text{pip}} \quad (4)$$

**Prior Enhanced Decoding (PED)** In addition to reducing the preference of $Y^{en}$ during training, we also propose Prior Enhanced Decoding (PED) that enhances $Y^l$ in the inference stage. PED integrates the confidence scores from the base model during auto-regressive inference. It leverages the strong language prior knowledge in the base model learned from the large-scale pre-training. Formally, PED calculates the confidence score of the $j$-th token at step $i$ in the inference process as follows:

$$\begin{aligned} s_{\text{it}}(t_j) &= -\log P(t_j, Y_{<i}, X; \theta) \\ s_{\text{base}}(t_j) &= -\log P(t_j, Y_{<i}, X; \theta_{\text{base}}) \\ s_{\text{ped}}(t_j) &= s_{\text{it}}(t_j) + s_{\text{base}}(t_j) \end{aligned} \quad (5)$$

where $t_j \in V$ denotes the $j$-th token in the vocabulary $V$, $\theta$ denotes the parameters of the instruction tuned model, $\theta_{\text{base}}$ denotes the parameters of the base model, and $s_*(t_j)$ refers to the confidence score of predicting the token $t_j$. In practice, we notice that $s_{\text{base}}$ are more evenly distributed across $V$ compared to $s_{\text{it}}$. This means that the top-ranking tokens from $s_{\text{ped}}$ would probably be the same as those from $s_{\text{it}}$. To address this, we narrow down the candidate set to tokens that receive high confidence in $s_{\text{base}}$. Specifically, we follow Li et al. (2023c) to select the candidate set at each step $i$ as:

$$V_{\text{cand}} = \{t_j \in V : s_{\text{base}}(t_j) \geq \alpha \max_t s_{\text{base}}(t)\}$$

where $\alpha$ is a hyper-parameter to control the minimum confidence compared to the most likely token. The final confidence score is determined as follows:

$$s_{\text{final}}(t_j) = \begin{cases} s_{\text{ped}}(t_j) & \text{if } t_j \in V_{\text{cand}} \\ -\inf & \text{otherwise} \end{cases} \quad (6)$$

## 4 Experimental Setup

In this section, we introduce the experimental setup including the base models, training dataset, evaluation settings, and baseline methods.

### 4.1 Base Models

**Llama2** is an open-source foundation language model proposed by Touvron et al. (2023b). Due to its wide application (Wang et al., 2023a; Zhang et al., 2023) and potential multilingual ability (Yuan et al., 2023), we choose the 7B version of Llama2 as our base model for instruction tuning.

**Llama2-Chat** is a dialogue optimized model based on Llama2 (Touvron et al., 2023b). It is fine-tuned through several successive SFT and RLHF (Ouyang et al., 2022) procedures, mostly in English. It thus shows limited generation ability in other languages. Nevertheless, Llama2-Chat still possesses some multilingual understanding and machine translation abilities (Sennrich et al., 2023). As a result, we perform the inference-based methods on Llama2-Chat and verify to what extent these methods mitigate the language inconsistency problem. We also use the Llama2-Chat 7B version by default.

**BLOOM** is a multilingual large language model which explicitly support 46 languages (Workshop et al., 2023). It is widely used for building multilingual models. We choose the 7.1B version of BLOOM for instruction tuning.

### 4.2 Training Dataset

For training-based methods, we employ the Stanford Alpaca[2] (Taori et al., 2023) dataset for instruction tuning. It consists of 52K English instruction-response pairs built through Self-Instruct (Wang et al., 2023b) and filtering offensive content with online moderation API[3].

### 4.3 Evaluation Settings

We evaluate the model with respect to instruction following, language understanding, and machine translation in this work.

***Instruction Following*** We choose Vicuna Benchmark[2] (Zheng et al., 2023) to assess models' abilities to follow human instructions. It comprises 80 English instructions covering various domains written by humans. We exclude 10 instructions related to coding and math, since they yield unreliable results when detecting the response languages. We translate the remaining 70 instructions into 10 languages including French (fr), German (de), Swedish (sv), Chinese (zh), Japanese (ja), Korean (ko), Arabic (ar), Spanish (es), Portuguese (pt) and Vietnamese (vi) with Baidu Translation API[4]. We denote the translated version of Vicuna with 10 languages as M-Vicuna.

We evaluate the instruction-following ability in terms of the correct use of language and the response quality. A language detector[5] is adopted to automatically detect the language used in each response, and report the ratio of correct usage of the same language as the input instruction. We denote this metric as the Language Consistency Rate (LCR). For response quality, we follow the same evaluation setting as in (Chen et al., 2023b), where GPT-3.5-Turbo[6] is prompted to score a response with points in the range of [0,3]. We report the average score of responses for each language, and multiply by 100 for better formatting.

***Language Understanding*** Since the training-based methods may harm the existing knowledge learned from pre-training, we evaluate these models on language understanding benchmarks as well. We choose ARC (Clark et al., 2018) and MMLU (Hendrycks et al., 2020) for English evaluation. For non-English languages, we use the translated version of MMLU and ARC released by (Lai et al., 2023). We report the average multiple-choice accuracy across 9 languages including English, French, Chinese, German, Swedish, Arabic, Spanish, Portuguese, and Vietnamese, and denoted as M-MMLU and M-ARC respectively.

***Machine Translation*** We evaluate the inference-based method on machine translation. We choose Flores-101 (Goyal et al., 2022) to test the zero-shot translation performance from English to 10 target languages the same with M-Vicuna. We adopt the same instruction as in (Sennrich et al., 2023) to prompt the model to perform translation. Language consistent rate (LCR) and spBLEU (Goyal et al., 2022) are reported as metrics. We calculate sp-

[2]Released under Apache-2.0 license.

[3]https://platform.openai.com/docs/api-reference/moderations/object

[4]https://api.fanyi.baidu.com/

[5]https://pypi.org/project/langdetect/

[6]https://platform.openai.com/docs/models/gpt-3-5

| Method | Base Model | Train Lang. | Instruction Following | | Language Understanding | |
|---|---|---|---|---|---|---|
| | | | LCR | GPT | M-MMLU | M-ARC |
| Vanilla training | Llama2-7B | en | 54.00 | 87.27 | 37.35 | 38.37 |
| Reducing $LR_{5e^{-5}}$ | Llama2-7B | en | 65.86 | 106.71 | 37.40 | 38.25 |
| Reducing $LR_{1e^{-5}}$ | Llama2-7B | en | 69.86 | 111.57 | 32.29 | 37.03 |
| ALT | Llama2-7B | en,zh | **98.43** | **162.00** | 37.12 | 38.03 |
| PIP (Ours) | Llama2-7B | en | 96.43 | 155.27 | **37.92** | **38.98** |
| Vanilla training | BLOOM-7B1 | en | 74.57 | 113.43 | 25.57 | **36.42** |
| PIP (Ours) | BLOOM-7B1 | en | **89.57** | **126.71** | **26.32** | 35.84 |

Table 1: Evaluation of the training-based methods across instruction following and language understanding. We report the average scores across all languages. Performance of each language is shown in Appendix C.

BLEU using sacreBLEU[7] (Post, 2018)

### 4.4 Compared Training Baselines

**Vanilla training.** We fine-tune Llama2-7B (Touvron et al., 2023b) directly with the loss function in Equation (1).

**Reducing LR.** Chirkova et al. (2023) suggests that reducing the learning rate of monolingual fine-tuning can alleviate language inconsistency. We adjust the learning rate from $1e^{-4}$ to $1e^{-5}$ and $5e^{-5}$ respectively as baselines.

**Auxiliary Language Training (ALT).** Li and Murray (2023) suggests training with an auxiliary language can alleviate the failure in cross-lingual generalization. As a baseline, we jointly train with Chinese alpaca data released by Lai et al. (2023). The Chinese dataset is translated from English Alpaca through GPT-3.5-turbo API. We sample the same amount of English and Chinese instructions and keep the whole training budget the same with monolingual fine-tuning following (Chen et al., 2023b).

### 4.5 Compared Inference Baselines

**Vanilla inference.** We directly provide the multilingual instructions to the Llama2-Chat-7B model in the default prompt template and do not specify additional prompting.

**Language Prompt (LP).** The assumed language is specified by appending the language prompt in the instruction. We attempt several language prompts on Llama2-Chat-7B and use $TLP_{out}$ as default since it performs best in guiding the language use of the model. More details about the language prompt are presented in Appendix A.1.

**In-Context Learning (ICL).** We provide one-shot in-context example (Brown et al., 2020) to encourage the model to respond in consistent language. The prompt structure is presented in Appendix A.2.

**Language Contrastive Decoding (LCD).** We implement language contrastive decoding following (Sennrich et al., 2023). Since in almost all inconsistent cases, the model generates English responses. We construct negative sentences by appending prompt `"Answer in English:"`. For positive sentences, we append the same prompt as LP.

## 5 Experiment Results and Analyses

### 5.1 Evaluation of Training-based Method

Table 1 compares the training-based methods on instruction tuning in terms of language consistency rate and GPT3.5 evaluation. We observe that directly training Llama2-7B on English Alpaca achieves only 54% language consistency rate on average when providing non-English instructions. Employing a lower learning rate such as $1e^{-5}$ indeed helps improve language consistency. However, this improvement is relatively modest, and it can lead to a decline in language understanding capabilities, which indicates that learning rate adjustment is not an ideal solution to resolve the language inconsistency problem. Auxiliary training with Chinese data can significantly alleviate the language inconsistency issue. However, it requires additional language resources, and also harms language understanding due to the curse of multilinguality (Conneau et al., 2020).

In contrast, our proposed PIP method significantly enhances the language consistency rates of the LLMs after English instruction tuning, while preserving the language understanding capability. Notably, PIP attains this substantial improvement without using any non-English resources. Furthermore, the GPT-3.5 evaluation confirms that PIP also improves the quality of the multilingual response. These results demonstrate that our method

[7] Signature:#:1|c:mixed|e:no|tok:flores101|s:exp|v:2.3.1

| Method | Lang. specify | Instruction Following | | Machine Translation | |
|---|---|---|---|---|---|
| | | LCR | GPT | LCR | spBLEU |
| Vanilla inference | ✗ | 18.14 | 52.29 | - | - |
| LP | ✓ | 55.86 | 64.71 | 92.28 | 21.39 |
| ICL (Brown et al., 2020) | ✓ | 63.86 | 145.57 | 82.56 | 21.96 |
| LCD (Sennrich et al., 2023) | ✓ | 72.43 | 149.86 | 92.26 | 21.49 |
| PED (Ours) | ✗ | 72.57 | 147.14 | - | - |
| PED+LP (Ours) | ✓ | **91.86** | **181.43** | **93.74** | **22.28** |

Table 2: Evaluation on inference-based methods. All methods are performed on Llama2-Chat. -: machine translation is not applicable without specifying target languages. We report the average scores across all languages. Performance of each language is shown in Appendix C.

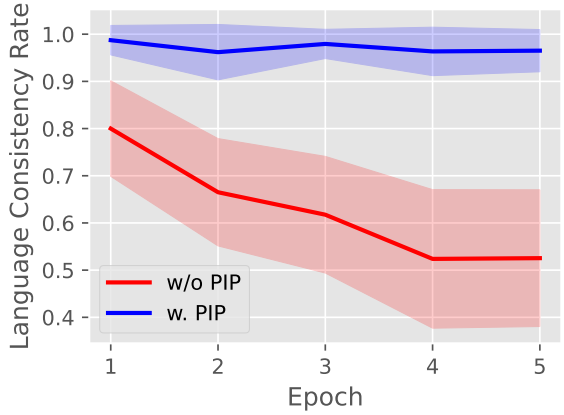



Figure 4: Language consistency during training. The transparent area displays the variance of 10 languages.

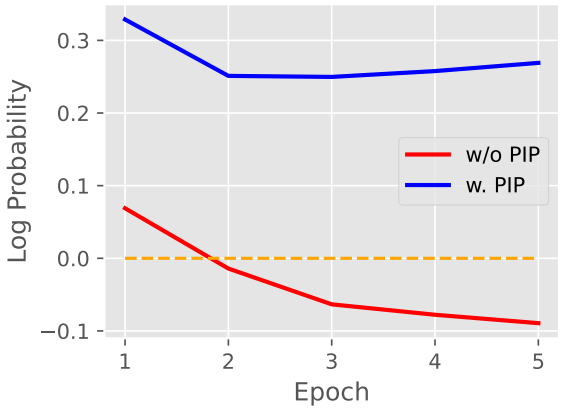



Figure 5: Average $\log P(Y^l|X^l) - \log P(Y^{en}|X^l)$ across all languages.

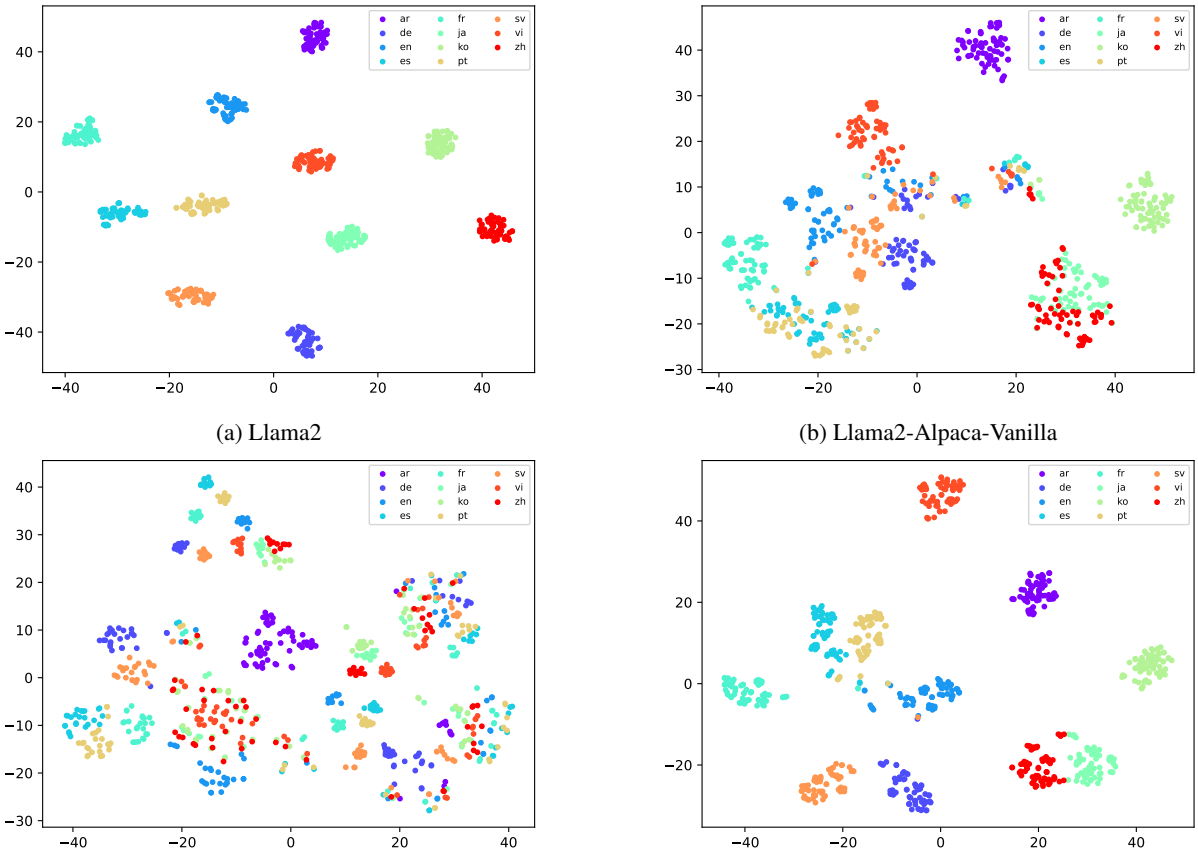
successfully mitigates the negative impact that monolingual instruction tuning brings to other languages.

### 5.2 Evaluation of Inference-based Method

Table 2 shows the performances of inference-based methods applied on Llama2-Chat-7B. We find that Llama2-Chat-7B exhibits a more severe language inconsistency issue compared to Llama2-Alpaca, with only 18.1% of responses using the correct language. This could be attributed to Llama2-Chat undergoing more extensive English fine-tuning (Touvron et al., 2023b). While using language prompts improves consistency to 55.9%, it is not a natural and user-friendly approach as it requires specifying the language in the instruction. ICT and LCD further improves consistency to 63.9% and 72.4%, but they also rely on language specification. Our proposed PED method achieves 72.6% language consistency without requiring any language specification. It further achieves 91.9% when combined with language specification, and attains the highest average scores in quality assessments. These results demonstrate that our proposed PED stands out for its ability to not only significantly improve language consistency in a user-friendly manner without the need for specifying languages but also enhance response quality. Although the PED encourages the model to generate responses in a consistent language, it also shows improvement in machine translation, where the input and output languages are different. We believe this is because the language prior introduced by PED is context-dependent. It can adapt to the semantic context and encourage the desired language. Therefore, the PED method not only enhances language consistency but also has a positive impact on the model's ability to handle tasks involving language switching.

### 5.3 Visualization of the Training Process

Figure 4 visualizes the average language consistency rate across 10 languages during the whole training process. With our proposed PIP, the model can maintain a stable language consistency during the whole training process, otherwise the language consistency will continue to decrease as the training




(a) Llama2

(b) Llama2-Alpaca-Vanilla

(c) Llama2-Chat

(d) Llama2-Alpaca-PIP

Figure 6: Instruction representation visualization with t-SNE on M-Vicuna.

progresses. In Figure 5, we illustrate the average change curve of $\log P(Y^l|X^l) - \log P(Y^{en}|X^l)$ across 10 languages during the training. Our proposed PIP ensures that responses in a consistent language obtain higher probabilities than responses in English. In contrast, without PIP, responses with inconsistent language will gain higher probabilities as training progresses. These visualizations again demonstrate that PIP indeed prevents the model from being biased towards English and enhances the model to maintain language consistency.

### 5.4 Visualization of Multilingual Instruction Representations.

We use t-SNE (Van der Maaten and Hinton, 2008) to visualize the instruction representations generated by the model when processing M-Vicuna. We treat the last-layer hidden state corresponding to the last token as the representation of the instruction. As shown in Figure 6, we can see that the instruction representations produced by Llama2-7b are completely distinguishable by language. It indicates the base model's ability to differentiate between languages. However, the Llama2-Alpaca and Llama2-Chat models, which are fine-tuned to follow English instructions, show a reduced language discrimination ability. This loss of language discrimination could lead to their inability to differentiate input languages, resulting in responses that are inconsistent with the input language. In contrast, the model trained with the PIP method well preserves the language discriminability of Llama2-7b sentence vectors. This visualization from the perspective of sentence vectors confirms the effectiveness of our method.

## 6 Conclusion

In this paper, we investigate the language inconsistency problem in response generation based on large language models. We find that monolingual instruction tuning increases the prior probability of English responses, and thus leads to language inconsistent generation when providing instructions in other languages. To address this issue, we propose Pseudo Inconsistent Penalization (PIP) to prevent the models from preferring to generate English responses during training, and Prior Enhanced Decoding (PED) to improve the likelihood

of language-consistent responses during inference. Experiment results demonstrate that our methods significantly enhance the language consistency rate without requiring any multilingual data while maintaining language understanding and machine translation capabilities of the large language models.

## Limitations

Our approach is constrained by the multilingual capabilities of the base language model. If the base model has limitations in its multilingual proficiency, these will be reflected in the outcomes of our methods, such as generating low-quality multilingual responses. Therefore, while our framework offers significant improvements in language consistency, its performance is still dependent on the foundational multilingual competencies of the chosen language model.

## Ethical Impact

Our research mitigates the English bias in large language models to enhance their applicability in multilingual contexts. This is particularly beneficial for user groups with limited data and computational resources, providing more equitable access to advanced language processing technologies. However, it also brings potential challenges in safe alignment under zero-shot multilingual scenarios, which requires careful consideration and further research. Our work marks a step towards more inclusive and versatile AI language systems.

## Acknowledgements

We thank all reviewers for their insightful comments and suggestions. This work was partially supported by the National Natural Science Foundation of China (No. 62072462) and the Beijing Natural Science Foundation (No. L233008).

## References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. 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 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Jordi Armengol-Estapé, Ona de Gibert Bonet, and Maite Melero. 2022. On the multilingual capabilities of very large-scale English language models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3056–3068, Marseille, France. European Language Resources Association.

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.

Liang Chen, Shuming Ma, Dongdong Zhang, Furu Wei, and Baobao Chang. 2023a. On the off-target problem of zero-shot multilingual neural machine translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9542–9558, Toronto, Canada. Association for Computational Linguistics.

Pinzhen Chen, Shaoxiong Ji, Nikolay Bogoychev, Barry Haddow, and Kenneth Heafield. 2023b. Monolingual or multilingual instruction tuning: Which makes a better alpaca.

Zhihong Chen, Feng Jiang, Junying Chen, Tiannan Wang, Fei Yu, Guiming Chen, Hongbo Zhang, Juhao Liang, Chen Zhang, Zhiyi Zhang, et al. 2023c. Phoenix: Democratizing chatgpt across languages. *arXiv preprint arXiv:2304.10453*.

Nadezhda Chirkova, Sheng Liang, and Vassilina Nikoulina. 2023. Empirical study of pretrained multilingual language models for zero-shot cross-lingual generation.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard 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.

Eccleston Dom and Tey Steven. 2023. Sharegpt. `https://sharegpt.com`.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 Evaluation Benchmark for Low-Resource and Multilingual Machine Translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O.K. Li. 2019. Improved zero-shot neural machine translation via ignoring spurious correlations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1258–1268, Florence, Italy. Association for Computational Linguistics.

Thanh-Le Ha, Jan Niehues, and Alex Waibel. 2016. Toward multilingual neural machine translation with universal encoder and decoder. In *Proceedings of the 13th International Conference on Spoken Language Translation*, Seattle, Washington D.C. International Workshop on Spoken Language Translation.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Jinyi Hu, Yuan Yao, Chongyi Wang, Shan Wang, Yinxu Pan, Qianyu Chen, Tianyu Yu, Hanghao Wu, Yue Zhao, Haoye Zhang, Xu Han, Yankai Lin, Jiao Xue, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2023. Large multilingual models pivot zero-shot multimodal learning across languages.

Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback.

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.

Chong Li, Shaonan Wang, Jiajun Zhang, and Chengqing Zong. 2023a. Align after pre-train: Improving multilingual generative models with cross-lingual alignment.

Haonan Li, Fajri Koto, Minghao Wu, Alham Fikri Aji, and Timothy Baldwin. 2023b. Bactrian-x: A multilingual replicable instruction-following model with low-rank adaptation. *arXiv preprint arXiv:2305.15011*.

Tianjian Li and Kenton Murray. 2023. Why does zero-shot cross-lingual generation fail? an explanation and a solution. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12461–12476, Toronto, Canada. Association for Computational Linguistics.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023c. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12286–12312, Toronto, Canada. Association for Computational Linguistics.

Kaushal Kumar Maurya, Maunendra Sankar Desarkar, Yoshinobu Kano, and Kumari Deepshikha. 2021. ZmBART: An unsupervised cross-lingual transfer framework for language generation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2804–2818, Online. Association for Computational Linguistics.

Openai. 2022. Chatgpt. `https://openai.com/blog/chatgpt`.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Jonas Pfeiffer, Francesco Piccinno, Massimo Nicosia, Xinyi Wang, Machel Reid, and Sebastian Ruder. 2023. mmt5: Modular multilingual pre-training solves source language hallucinations. *arXiv preprint arXiv:2305.14224*.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Leonardo Ranaldi and Giulia Pucci. 2023. Does the english matter? elicit cross-lingual abilities of large language models. In *Proceedings of the 3rd Workshop on Multi-lingual Representation Learning (MRL)*, pages 173–183.

Annette Rios, Mathias Müller, and Rico Sennrich. 2020. Subword segmentation and a single bridge language affect zero-shot neural machine translation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 528–537, Online. Association for Computational Linguistics.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021. WikiMatrix: Mining 135M parallel sentences in 1620 language pairs from Wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1351–1361, Online. Association for Computational Linguistics.

Rico Sennrich, Jannis Vamvas, and Alireza Mohammadshahi. 2023. Mitigating hallucinations and off-target machine translation with source-contrastive and language-contrastive decoding. *arXiv preprint arXiv:2309.07098*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models.

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

Tu Vu, Aditya Barua, Brian Lester, Daniel Cer, Mohit Iyyer, and Noah Constant. 2022. Overcoming catastrophic forgetting in zero-shot cross-lingual generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9279–9300, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Raghavi Chandu, David Wadden, Kelsey MacMillan, Noah A Smith, Iz Beltagy, et al. 2023a. How far can camels go? exploring the state of instruction tuning on open resources. *arXiv preprint arXiv:2306.04751*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023b. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. 2020. Neural text generation with unlikelihood training. In *International Conference on Learning Representations*.

BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev,

Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov, Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. 2023. Bloom: A 176b-parameter open-access multilingual language model.

Liwei Wu, Shanbo Cheng, Mingxuan Wang, and Lei Li. 2021. Language tags matter for zero-shot neural machine translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3001–3007, Online. Association for Computational Linguistics.

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

Yilin Yang, Akiko Eriguchi, Alexandre Muzio, Prasad Tadepalli, Stefan Lee, and Hany Hassan. 2021. Improving multilingual translation by representation and gradient regularization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7266–7279, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Fei Yuan, Shuai Yuan, Zhiyong Wu, and Lei Li. 2023. How multilingual is multilingual llm?

Changtong Zan, Liang Ding, Li Shen, Yibin Lei, Yibing Zhan, Weifeng Liu, and Dacheng Tao. 2023. Unlikelihood tuning on negative samples amazingly improves zero-shot translation. *arXiv preprint arXiv:2309.16599*.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In

*Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online. Association for Computational Linguistics.

Zhihan Zhang, Dong-Ho Lee, Yuwei Fang, Wenhao Yu, Mengzhao Jia, Meng Jiang, and Francesco Barbieri. 2023. Plug: Leveraging pivot language in cross-lingual instruction tuning.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Wenhao Zhu, Yunzhe Lv, Qingxiu Dong, Fei Yuan, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Extrapolating large language models to non-english by aligning languages. *arXiv preprint arXiv:2308.04948*.

# A Prompt Structures

## A.1 Language Prompt

We attempt the following prompts to specify the language to respond:
**System Language Prompt (SLP)**: use system prompt to specifying the language.
**English Language Prompt (ELP)**: specifying the language in English. $\text{ELP}_{\text{out}}$ means put the prompt outside the `[INST]` block.
**Target Language Prompt (TLP)**: specifying the language in the target language. $\text{TLP}_{\text{out}}$ means put the prompt outside the `[INST]` block.
Table 3 shows examples for each of the above-mentioned prompts and the average language consistency rate on M-Vicuna. We use $\text{TLP}_{\text{out}}$ to specify the language by default since it performs the best.

| Prompt | Example | LCR |
|---|---|---|
| SLP | `[INST] «SYS» You are a helpful assistant who always speaks in French. «/SYS» {instruction} [/INST]` | 42.29 |
| ELP | `[INST] {instruction} Answer in French. [/INST]` | 31.43 |
| $\text{ELP}_{\text{out}}$ | `[INST] {instruction} [/INST] Answer in French:` | 42.38 |
| TLP | `[INST] {instruction} Répondre en français. [/INST]` | 48.28 |
| $\text{TLP}_{\text{out}}$ | `[INST] {instruction} [/INST] Répondre en français:` | **55.86** |

Table 3: Different language prompts we attempted on Llama2-Chat-7B. The language consistency rate is reported by testing on M-Vicuna.

## A.2 In-Context Learning Prompts

We present the prompt structure for in-context learning. For the instruction following task, we randomly select one instruction-response pair from English Alpaca and translate it into the target language as the in-context example. We find that the prompt structure used in in-context learning has a significant impact on the language consistency of the response. We tries the following two prompt structures, and use $\text{ICL}_2$ by default.
For the machine translation task, we use the one-shot prompt structure provided by Sennrich et al. (2023).

| Prompt | Example | LCR |
|---|---|---|
| $\text{ICL}_1$ | `[INST] {example instruction} [/INST] {example response}`<br>`[INST] {instruction} [/INST]` | 13.43 |
| $\text{ICL}_2$ | `[INST] Please answer questions in the same language.`<br>`Here is an example:`<br>`Question: {example instruction}`<br>`Answer in the same language: {example response}`<br>`Question: {instruction} [/INST]`<br>`Answer in the same language:` | **63.86** |

Table 4: Two in-context learning prompts we attempted on Llama2-Chat-7B. The language consistency rate is evaluated on M-Vicuna.

# B Implementation Details

For all training-based methods, we train the model on the English Standford Alpaca (Taori et al., 2023) for 5 epochs with an overall batch size of 128 and a learning rate of $1e^{-4}$ by default. We warm up the

model in the first 3% steps and decay the learning rate to 0 linearly at the end of training. We adopt LoRA (Hu et al., 2021) for parameter-efficient fine-tuning with a rank of 64. For PIP, we randomly chose one of the 10 non-English languages for each instruction to construct pseudo-inconsistent samples for unlikelihood optimization. The prompt structure of $R^l$ is TLP (See Appendix A.1 for details). All the models are trained on 4 NVIDIA RTX A6000 GPUs with 48G memory within 15 hours. We perform the greedy search for all inference-based methods, and set the maximum decoding length as 1024 tokens by default. For PED, we search $\alpha$ between $\{1e^{-1}, 1e^{-2}, 1e^{-3}\}$ and adopt $1e^{-2}$ for instruction following and $1e^{-3}$ for machine translation.

## C Performance of Specific Languages

We report the performance of each language in Table 5, 6, and 7. We notice that our PIP method slightly affects the quality of English response generation, as it suppresses English responses during training. However, it is worth noting that this impact is relatively small compared to the improvement brought to other languages, and it also hardly affects English comprehension ability. We argue that it is worthwhile to make slight concessions on the quality of English generation in exchange for stronger language consistency.

| Method | Language Consistency Rate | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | en | fr | es | pt | vi | ja | ar | sv | ko | de | zh |
| Vanilla training | 100.0 | 75.7 | 62.9 | 71.4 | 45.7 | 47.1 | 37.1 | 37.1 | 60.0 | 67.1 | 35.7 |
| Reducing $LR_{1e^{-5}}$ | 100.0 | 88.6 | 81.4 | 80.0 | 68.6 | 58.6 | 41.4 | 75.7 | 74.3 | 81.4 | 48.6 |
| Reducing $LR_{5e^{-5}}$ | 100.0 | 85.7 | 85.7 | 87.1 | 48.6 | 58.6 | 30.0 | 71.4 | 60.0 | 81.4 | 50.0 |
| ALT | 100.0 | 98.6 | 100.0 | 100.0 | 100.0 | 91.4 | 97.1 | 100.0 | 100.0 | 98.6 | 98.6 |
| PIP | 100.0 | 98.6 | 98.6 | 97.1 | 98.6 | 95.7 | 98.6 | 97.1 | 100.0 | 95.7 | 84.3 |
| GPT3.5 Evaluation | | | | | | | | | | | |
| Vanilla training | 198.6 | 127.1 | 114.3 | 117.1 | 70.0 | 72.9 | 38.6 | 65.7 | 90.0 | 117.1 | 60.0 |
| Reducing $LR_{1e^{-5}}$ | 207.1 | 154.3 | 147.1 | 142.9 | 107.1 | 85.7 | 45.7 | 128.6 | 91.4 | 141.4 | 71.4 |
| Reducing $LR_{5e^{-5}}$ | 201.4 | 148.6 | 157.1 | 151.4 | 72.9 | 91.4 | 31.4 | 124.3 | 80.0 | 137.1 | 72.9 |
| ALT | 201.4 | 180.0 | 181.4 | 177.1 | 157.1 | 150.0 | 108.6 | 175.7 | 141.4 | 174.3 | 174.3 |
| PIP | 194.3 | 180.0 | 181.4 | 171.4 | 154.3 | 152.9 | 100.0 | 167.1 | 138.6 | 170.0 | 137.0 |
| Multilingual MMLU | | | | | | | | | | | |
| Okapi* | - | 30.7 | 30.9 | - | - | - | - | - | - | 31.7 | - |
| Vanilla training | 45.4 | 39.7 | 40.2 | 38.9 | 34.0 | - | 28.2 | 37.0 | - | 38.5 | 34.3 |
| Reducing $LR_{1e^{-5}}$ | 40.2 | 32.9 | 33.6 | 33.2 | 29.8 | - | 26.5 | 32.0 | - | 33.5 | 29.0 |
| Reducing $LR_{5e^{-5}}$ | 45.1 | 39.7 | 40.2 | 38.4 | 34.0 | - | 29.3 | 36.9 | - | 38.7 | 34.4 |
| ALT | 45.0 | 38.9 | 39.6 | 38.6 | 34.6 | - | 28.9 | 36.5 | - | 38.0 | 33.9 |
| PIP | 45.9 | 39.7 | 40.2 | 39.8 | 34.8 | - | 29.3 | 37.1 | - | 39.2 | 35.3 |
| Multilingual ARC | | | | | | | | | | | |
| Okapi* | - | 39.6 | 38.3 | - | - | - | - | - | - | 36.0 | - |
| Vanilla training | 50.7 | 41.8 | 40.9 | 41.5 | 31.5 | - | 25.7 | 39.0 | - | 38.1 | 36.1 |
| Reducing $LR_{1e^{-5}}$ | 48.0 | 40.5 | 38.9 | 39.7 | 30.4 | - | 26.3 | 37.7 | - | 36.6 | 35.1 |
| Reducing $LR_{5e^{-5}}$ | 50.4 | 41.1 | 40.1 | 41.4 | 31.2 | - | 26.3 | 39.7 | - | 38.5 | 35.6 |
| ALT | 48.6 | 41.1 | 40.6 | 41.3 | 30.9 | - | 25.6 | 39.0 | - | 38.0 | 37.2 |
| PIP | 50.2 | 42.3 | 40.8 | 42.1 | 32.1 | - | 27.4 | 39.6 | - | 39.3 | 37.2 |

Table 5: Performance of training-based methods in each language. Base model: Llama2-7B. *: results reported by Lai et al. (2023).

| Method | Language Consistency Rate | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | en | fr | es | pt | vi | ja | ar | sv | ko | de | zh |
| Vanilla training | 100.0 | 98.6 | 100.0 | 97.1 | 98.6 | 55.7 | 97.1 | 15.7 | 21.4 | 68.6 | 92.9 |
| PIP | 98.6 | 100.0 | 100.0 | 100.0 | 100.0 | 75.7 | 100.0 | 52.9 | 91.4 | 81.4 | 94.3 |
| GPT3.5 Evaluation | | | | | | | | | | | |
| Vanilla training | 184.3 | 160.0 | 161.4 | 158.6 | 165.7 | 61.4 | 160.0 | 11.4 | 22.9 | 70.0 | 162.9 |
| PIP | 172.9 | 167.1 | 165.7 | 170.0 | 167.1 | 81.4 | 140.0 | 41.4 | 90.0 | 77.1 | 167.1 |
| Multilingual MMLU | | | | | | | | | | | |
| Vanilla training | 25.1 | 25.6 | 25.6 | 25.5 | 25.2 | - | 25.2 | 25.8 | - | 26.6 | 25.6 |
| PIP | 25.3 | 26.2 | 25.8 | 26.3 | 26.6 | - | 25.8 | 26.8 | - | 27.1 | 26.8 |
| Multilingual ARC | | | | | | | | | | | |
| Vanilla training | 42.6 | 40.5 | 39.9 | 42.6 | 35.6 | - | 35.2 | 25.8 | - | 26.9 | 38.7 |
| PIP | 42.1 | 40.6 | 40.2 | 42.1 | 35.1 | - | 33.7 | 23.9 | - | 26.4 | 38.5 |

Table 6: Performance of training-based methods in each language. Base model: BLOOM-7B1.

| Method | Language Consistency Rate | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | en | fr | es | pt | vi | ja | ar | sv | ko | de | zh |
| Vanilla inference | 100.0 | 25.7 | 70.0 | 44.3 | 4.3 | 0.0 | 0.0 | 17.1 | 0.0 | 20.0 | 0.0 |
| LP | 100.0 | 54.3 | 94.3 | 81.4 | 27.1 | 44.3 | 72.9 | 32.9 | 24.3 | 47.1 | 80.0 |
| ICL | 72.9 | 58.6 | 61.4 | 60.0 | 78.6 | 67.1 | 87.1 | 47.1 | 81.4 | 37.1 | 60.0 |
| LCD | 100.0 | 88.6 | 98.6 | 90.0 | 60.0 | 54.3 | 87.1 | 51.4 | 42.9 | 68.6 | 82.9 |
| PED | 91.4 | 70.0 | 88.6 | 90.0 | 74.3 | 64.3 | 94.3 | 52.9 | 78.6 | 35.7 | 77.1 |
| PED+LP | 91.4 | 91.4 | 97.1 | 97.1 | 90.0 | 77.1 | 100.0 | 91.4 | 87.1 | 97.1 | 90.0 |
| GPT3.5 Evaluation | | | | | | | | | | | |
| Vanilla inference | 292.9 | 75.7 | 204.3 | 127.1 | 8.6 | 0.0 | 0.0 | 48.6 | 0.0 | 58.6 | 0.0 |
| LP | 292.9 | 160.0 | 272.9 | 235.7 | 61.4 | 114.3 | 85.7 | 90.0 | 51.4 | 131.4 | 215.7 |
| ICL | 201.4 | 158.6 | 165.7 | 160.0 | 165.7 | 167.1 | 95.7 | 117.1 | 167.1 | 100.0 | 158.6 |
| LCD | 292.9 | 197.1 | 265.7 | 244.3 | 78.6 | 101.4 | 84.3 | 108.6 | 55.7 | 145.7 | 217.1 |
| PED | 250.0 | 185.7 | 235.7 | 244.3 | 148.6 | 114.3 | 95.7 | 108.6 | 118.6 | 71.4 | 148.6 |
| PED+LP | 230.0 | 232.9 | 260.0 | 257.1 | 161.4 | 147.1 | 100.0 | 204.3 | 134.3 | 134.3 | 182.9 |
| Machine Translation (LCR) | | | | | | | | | | | |
| LP | - | 98.0 | 97.5 | 96.3 | 95.9 | 97.0 | 96.7 | 94.1 | 82.6 | 95.0 | 69.7 |
| ICL | - | 97.2 | 97.6 | 93.8 | 97.1 | 79.1 | 80.8 | 90.7 | 41.5 | 90.4 | 57.3 |
| LCD | - | 98.6 | 97.4 | 96.5 | 96.5 | 97.3 | 97.1 | 94.6 | 82.4 | 95.2 | 67.0 |
| PED+LP | - | 96.1 | 95.7 | 94.2 | 95.5 | 97.5 | 97.5 | 91.0 | 95.4 | 91.0 | 83.6 |
| Machine Translation (spBLEU) | | | | | | | | | | | |
| LP | - | 37.5 | 24.9 | 36.5 | 20.6 | 15.8 | 3.7 | 29.6 | 8.6 | 25.9 | 10.8 |
| ICL | - | 37.0 | 25.3 | 37.1 | 21.7 | 17.5 | 3.7 | 29.4 | 10.0 | 26.1 | 12.0 |
| LCD | - | 37.6 | 24.9 | 36.7 | 20.8 | 15.8 | 3.9 | 29.7 | 8.6 | 26.0 | 10.9 |
| PED+LP | - | 37.1 | 24.7 | 37.2 | 20.7 | 17.7 | 4.9 | 30.3 | 11.1 | 26.0 | 13.2 |

Table 7: Performance of inference-based methods in each language. Base model: Llama2-Chat-7B.