# Language Fusion for Parameter-Efficient Cross-lingual Transfer

Anonymous ACL submission

## Abstract

Limited availability of multilingual text corpora for training language models often leads to poor performance on downstream tasks due to undertrained representation spaces for languages other than English. This 'under-representation' has motivated recent cross-lingual transfer methods to leverage the English representation space by e.g. mixing English and non-English tokens at input or extending model parameters, which in turn increases computational complexity. To address this, we introduce Fusion for Language Representations (FLARE) in adapters, a method designed to improve both the representation quality and downstream performance for languages other than English. FLARE integrates source and target language representations within the bottlenecks of low-rank LoRA adapters using lightweight linear transformations. This maintains parameter efficiency as the method does not require additional parameters, while improving transfer performance, further narrowing the performance gap to English. Another key advantage of the proposed latent representation fusion is that it does not increase the number of input tokens, thus maintaining computational efficiency. Moreover, FLARE provides flexibility to integrate various types of representations, e.g., we show that it is possible to fuse latent translations extracted from machine translation models. Our results demonstrate FLARE's effectiveness on natural language understanding tasks, reducing the performance gap to English across all tasks.[1]

## 1 Introduction

Representation degradation for 'non-English' languages poses a challenge in the context of multilingual pretrained language models (mPLMs)[2].

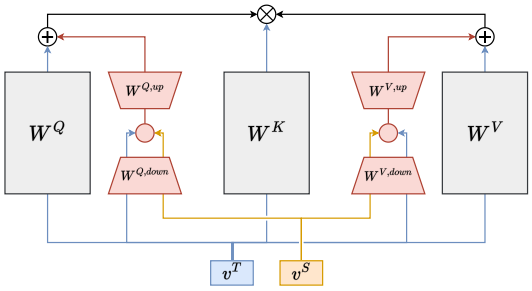


Figure 1: Fusion of source and target representations in LoRA adapters inserted within the query and value matrices. The representations are fused in the adapter bottlenecks and the outputs are added $\oplus$ to the query and value outputs before softmax $\otimes$ activation.

Large-scale English text corpora are widely available for self-supervised pretraining, resulting in superior representation quality and downstream task performance when compared to low(er)-resource languages (Lauscher et al., 2020; Yang et al., 2022). Training mPLMs on massively multilingual text data creates a unified representation space that enables cross-lingual information transfer. Despite the substantial improvements, the imbalance in pretraining resources still substantially reduces downstream performance (Winata et al., 2022).

Cross-lingual transfer (termed XLT henceforth) aims to narrow this performance gap by transferring task-specific knowledge acquired in high-resource languages to lower-resource languages (Ruder et al., 2019). Given the dominance of English in pretraining corpora, machine translations (MT) are frequently utilized to avoid processing non-English data (Shi et al., 2010; Artetxe et al., 2020, 2023). Techniques utilizing source and target language representation spaces include language mixup (Yang et al., 2022), and concatenating multilingual input sequences for in-context XLT (Kim et al., 2023; Tanwar et al., 2023; Villa-Cueva et al.,

[1] Our code repository is available at https://anonymous.4open.science/r/FLARE-7984

[2] The domination of the English representation space is observed independent of model architectures, including encoder-only, decoder-only and encoder-decoder transformer (Wu and Dredze, 2020; Lee et al., 2022a; Yang et al., 2022; Wendler et al., 2024; Tang et al., 2024).

2024). These approaches, while improving XLT, typically focus on representations in a specific mPLM layer or require extensive training and computational resources by extending the input length. Additionally, they typically rely on high-quality MT output for source language input. Despite the widespread use of discrete machine translations, only few studies explore enhancing the 'internal' information extracted from MT models (Ponti et al., 2021), and MT output is typically not used to model sub-sentential interaction between source and target language representations.

When adapting mPLMs to new tasks and languages, the choice of adaptation method is crucial for downstream performance. Parameter-efficient fine-tuning (PEFT) methods are designed to acquire new knowledge while minimizing the number of extra parameters required and keeping the large underlying mPLM frozen (Hu et al., 2021). In particular, bottleneck-style adapters extract relevant features from new data by compressing model representations with the assumption that task information can be captured in a lower-dimensional space (Houlsby et al., 2019). This directly aligns with the XLT objectives, providing resource-efficient language and task adaptation capabilities and support for infusing model representations with new knowledge. Similarly, low-rank adapters (LoRA) also create such 'representation bottlenecks'; they get inserted into the query and value attention modules, and exemplify a widely adopted PEFT approach in large language models (Hu et al., 2021). In XLT, adapters are extensively used for acquiring task and language knowledge (Pfeiffer et al., 2020), yet the extent of knowledge transfer within adapters themselves remains underexplored.

In this study, we thus introduce Fusion for Language Representations (FLARE) within adapter bottlenecks to improve parameter-efficient XLT. As illustrated in Figure 1, we propose token-wise fusion of source and target language representations within each transformer block. Our findings suggest that even lightweight linear transformations, such as addition or multiplication, enhance XLT performance, as they allow for the interaction of source and target language representations within the adapter bottlenecks. A key advantage of our method lies in its parameter efficiency, as the fusion operations are located within the adapter bottlenecks, thereby not introducing additional parameters while enhancing performance. Our experiments across natural language inference, sentiment classification, and question answering tasks, using encoder-only and encoder-decoder mPLMs, demonstrate that our fusion technique effectively reduces the cross-lingual transfer gap for XLM-R Large to 8.1% across all evaluated tasks. Further experiments illustrate that computational efficiency can be further enhanced by using latent translations as source language inputs in FLARE, and demonstrate the versatility of the method, which is orthogonal to the choice of mPLMs and MT systems.

**Contributions. 1)** We introduce the FLARE method, fusion for language representations in bottleneck adapters for parameter-efficient cross-lingual transfer. **2)** Our approach effectively narrows the transfer performance gap between English and other languages across various downstream tasks. **3)** We demonstrate the adaptability of our approach by incorporating machine translation encoder representations directly into the mPLM.

## 2 Related Work

**PEFT in Multilingual Language Models and Cross-Lingual Transfer.** PEFT aims to incorporate task or language-specific knowledge into mPLMs without updating all model weights (Pfeiffer et al., 2020). Most prominent techniques include sparse fine-tuning by selectively updating model parameters (Ansell et al., 2022), and inserting adapter modules that reduce trainable parameters to a small fraction of total weights of the underlying mPLM (Houlsby et al., 2019). Furthermore, PEFT modules are composable, and thus information combination from multiple modules is possible (Wang et al., 2022; Lee et al., 2022b).

Bottleneck adapters project model representations into a lower-dimensional space and then back to their original dimensions, creating a bottleneck that regulates information flow (Houlsby et al., 2019). During this adaptation process, the weights of the (m)PLM remain frozen. Following the same assumption that task-specific knowledge can be compressed in a low-dimensional space, low-rank adapters (LoRA) are widely utilized for fine-tuning language models (Hu et al., 2021). They are inserted into the attention modules of transformer architectures, maximizing the capacity to adapt to new task-specific information, while preserving parameter-efficiency. In this study, we extend the task and knowledge acquisition capabilities of adapters by leveraging such adapter bottlenecks (as created e.g. by LoRA) for XLT.

**Cross-lingual Representation Transfer.** Enhancing performance for languages underrepresented in the mPLMs' pretraining data often involves aligning and combining representations from various languages to facilitate XLT (Oh et al., 2022). By concatenating multilingual input sequences, mPLMs leverage a shared representation space across both source and target language inputs (Kim et al., 2023; Tanwar et al., 2023; Villa-Cueva et al., 2024). Techniques such as mixtures of task and language adapters have been implemented to merge language representation spaces effectively (Lee et al., 2022b). In projection-based approaches, target language representations are projected onto a high-resource language (e.g., English), to enhance feature extraction in the high-resource language, before re-projecting back to the target language (Xu et al., 2023). Yang et al. (2022) introduced a mixup method, combining source and target representations in a specific layer of the mPLM during task fine-tuning. Building on this concept, Cao et al. (2023) used cross-attention with semantic and token-level alignment loss terms, aiming to transfer knowledge from the source to the target language. Our work contributes to this research stream by enhancing *parameter-efficient XLT* and introducing *representation-agnostic fusion* methods.

Representation fusion is also applied to integrate information across different modalities (Fang et al., 2021; Ramnath et al., 2021). For instance, Qu et al. (2024) employed feature routing in cross-modal vision-language tasks, guiding language model representations through the LoRA bottleneck using the last hidden state of a vision model. Our work differs in its scope and fusion methodology: FLARE extracts significantly richer representations from the source and target languages by capturing layer-wise representations for each transformer block in the mPLMs. Moreover, by ensuring dimensional alignment, we perform token-wise representation fusion within adapter bottlenecks, thereby transferring finer-grained information across languages.

## 3 Methodology

### 3.1 Language Representation Fusion

Our methodology is based on the hypothesis that incorporating English with target language representations enhances cross-lingual knowledge transfer and distills task-relevant information into the target language. We assume (MT-created) parallel corpora $\mathcal{P} = \{(x^S, x^T)\}$ during task fine-tuning,

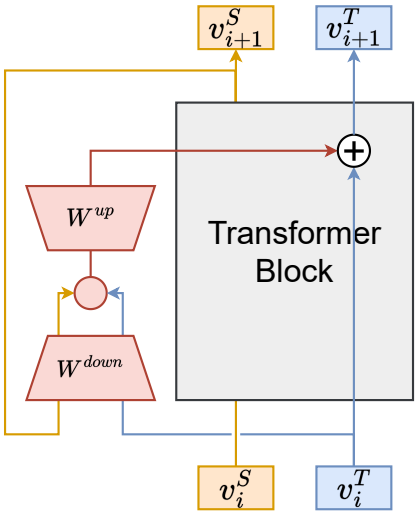


Figure 2: During the forward pass with fusion adapters, source language representations $x^S$ are fused with target language representations $x^T$ in each transformer block $i$. Source representations are extracted by inferencing the mPLM without the fusion adapters.

where $x$ are instances in the respective source and target language. Our methodology particularly focuses on employing machine-translated 'silver' parallel data, akin to *translate-train* and *translate-test* settings, as we believe this approach is the most realistic in practice. We contend that transferring information during task fine-tuning is more resource-efficient compared to extensive pretraining on large-scale self-supervised text corpora.

A straightforward and effective method for aligning multilingual representations is to concatenate source and target language input sequences $x^{S,T} = [x^S; x^T]$ where $x \in \mathbb{R}^{2m}$, with $m$ representing the sequence length of both source and target languages. This so-called *input-level fusion* enables cross-lingual knowledge transfer across all layers of the mPLM, facilitating in-context learning, which typically does not require additional training (Villa-Cueva et al., 2024). However, this approach is computationally expensive due to increased input sequence lengths and encounters scalability issues related to the context length limitations in mPLMs.

To address these limitations, we propose FLARE, a method for representation-level language fusion within low-rank bottleneck adapters; see Figure 1. Source language representations $v_i^S$, extracted from the frozen mPLM without adapters, and target language representation $v_i^T$ at transformer block $i$ are down-projected using $W^{down}$ and combined with fusion function $\phi$ to create a fused representation $h = \phi\left(v_i^S W^{down}, v_i^T W^{down}\right)$, where

$h \in \mathbb{R}^{m \times r}$ with sequence length $m$ and bottleneck dimensions $r$. Following a standard LoRA procedure, this fused low-rank representation is then up-projected and added to the frozen attention outputs $v^0$ to form the target language output representation $v^T_{i+1} = hW^{up} + v^0$ of the attention block. This enhances the target language adaptation by directing the model's attention to task-relevant information. The down-projection within the bottleneck adapters is applied to both target and source language representations, exploiting the unified embedding space acquired during self-supervised pretraining for cross-lingual adaptation.[3]

A key advantage of representation fusion is the reduction in computational complexity, thereby enhancing parameter efficiency for both task and language adaptation. By processing multilingual inputs separately and only fusing highly compressed representations within adapter bottlenecks, our method avoids the computational overhead associated with quadratic scaling in attention computations for model dimensions $d$, thus enhancing resource efficiency. Furthermore, the memory requirements are limited to the last hidden states obtained from the output of each transformer block.

Moreover, our fusion approach is agnostic regarding the source language representation. This flexibility allows directly leveraging representations extracted from the MT encoder $\mathcal{M}$ as 'latent translations' for fusion. We extract a single representation from the MT model $v^T = \mathcal{M}\left(x^T\right)$, where $v^T \in \mathbb{R}^{m \times d_{\mathcal{M}}}$, which serves as a latent translation. We project these MT model dimensions to align with the mPLM using a projection layer $W^{proj}$. Consequently, the up-projected representation $v^T W^{proj}$ is fused with the target language representation within the adapter bottlenecks of each mPLM layer; see Figure 3. This FLARE MT method enhances resource efficiency by bypassing a forward pass in the mPLM, which is required when using discrete text in the source language, and preserves the inherent translation uncertainty within the embeddings by avoiding discretization in the MT decoder, thus mitigating potential translation errors (Ponti et al., 2021; Unanue et al., 2023).

[3]Assuming that new task information can be learned within low-rank adapters, we posit that task-specific cross-lingual knowledge can be effectively transferred within adapter bottlenecks. This enhances efficiency, and also compresses and aligns task-relevant information, simplifying the complexity of representations $r \ll d$. This setup enables the application of lightweight transformations that merge information from both source and target representations.

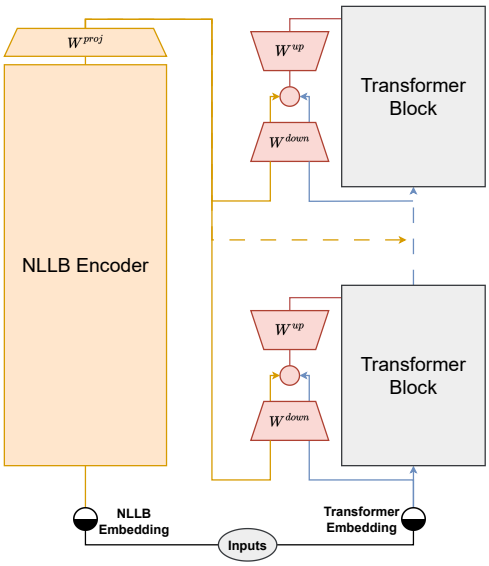


Figure 3: Illustration of the FLARE MT variant where projected encoder representations from an MT model are directly fused with target language representations within the fusion adapters in the mPLM. Encoder representations from the MT model serve as latent translations, avoiding discretization in the decoder.

### 3.2 Fusion Functions

To fuse cross-lingual representations in bottleneck adapters, we evaluate both linear and non-linear transformations that do not require additional model parameters, alongside cross-attention. We extract token-wise representations from source and target language sequences, capturing rich contextual information at the token level. Extracting source language and target language representations from the same underlying mPLM ensures matching hidden dimensions $d$ in each transformer layer, facilitating subsequent representation fusion in the low-rank bottleneck adapters.

The down-projected representations in the adapter bottlenecks for source and target languages are denoted as $S = v^S W^{down}$ and $T = v^T W^{down}$, where $S$ and $T$ are representations of dimensions $\mathbb{R}^{m \times r}$. These representations are subsequently combined at the token level through the following fusion functions:

1. element-wise addition (*add*): $S + T$
2. element-wise multiplication (*mul*): $S \circ T$
3. *cross-attention*:[4] $\text{softmax}\left(\frac{W_a^Q S \left(W_a^K T\right)'}{\sqrt{r}}\right) W_a^V T$

[4]Although cross-attention modules add parameters to the adapters, the low bottleneck dimensions $r$, typically smaller than 64, minimize the parameter count in comparison to the model's internal dimensions $d$. Specifically, we utilize a single cross-attention head to maintain efficiency.

$W_a^Q$, $W_a^K$ and $W_a^V$ are the weight matrices of the query, key and value projections in the adapter $a$, respectively, and $'$ denotes the matrix transpose.

We extend the linear fusion functions using non-linear transformations through rectified linear units $ReLU\,(S)$ and $ReLU\,(T)$ (Qu et al., 2024). This addition improves feature extraction capabilities by selectively enabling information flow in token representations. Given the inherent misalignment of multilingual input sequences at the token level, extracting token-level representations for subsequent fusion may introduce alignment issues. We hypothesize that the adapter projections $W^{down}$ aid the alignment of multilingual representations. Further correcting for misalignment between source and target language representations, non-linear transformation functions can restrict propagating misaligned information, which ultimately might improve downstream task performance.

### 3.3 Training

For task adaptation in the target language, we insert LoRA adapters into query and value weight matrices of the mPLM previously fine-tuned on English task data (referred to as the *base model*). In FLARE, these adapters implement fusion function $\phi$ that combines source and target language input representations into a single fused representation, as illustrated in Figure 1. Consistent with standard PEFT training, only the task head and LoRA parameters and output layer are trainable, while all other parameters remain frozen.

During the forward pass, detailed in Figure 2, representations from both the source and target languages are extracted at each transformer block. Layer-wise source language representations are obtained from the base model and stacked in matrix $V^S \in \mathbb{R}^{l \times m \times d}$, where $l$ represents the number of layers in the mPLM. Target language representations are obtained during the forward pass through the base model with LoRA adapters. In each layer, source and target language representations are transformed and compressed to lower dimensions $r \ll h$ in the adapter's down-projection $W^{down}$. The shared down-projection layers, applied to both source and target language representations before subsequent fusion, reduce the model's reliance on the English representation space. The final steps include the application of a fusion function and standard up-projection, as already described in Sections 3.1 and 3.2.

## 4 Experimental Setup

### 4.1 Underlying Models and Baselines

**mPLMs.** Our experiments are based on various mPLMs including XLM-R Base (270M parameters) and Large (550M) (Conneau et al., 2020), and mT5-XL (3.7B) (Xue et al., 2021). Additionally, experiments with LLaMA 3 (8B) (AI@Meta, 2024) are discussed in Appendix D.

**Fine-Tuning Setup.** We follow a modular XLT approach where the mPLM is fine-tuned on English task data and subsequently adapted using task data in the target language (Zhao et al., 2021). Unless stated otherwise, models are fine-tuned using $r = 64$ and $\alpha = 128$ in the LoRA configurations, while the hyperparameter configurations of each model are detailed in Table 5 in the appendix.

**Baselines.** We benchmark FLARE against zero-shot cross-lingual transfer, translate-test, and translate-train baselines, as well as input-level fusion models trained with the same LoRA configurations as the FLARE variants. Model checkpoints are selected on validation data that was machine translated from English to the respective target languages. More details on the baselines are provided below:

*Zero-Shot XLT.* The base model fine-tuned on English task data is directly evaluated on test data in the target languages without further training.

*Translate-Test.* Test sets in each target language are translated into English using NLLB (Team et al., 2022). Subsequently, the base model is evaluated on these machine-translated test sets.

*Translate-Train.* The base model is fine-tuned on machine-translated task data in the respective target languages. This setting assumes that no gold translations are available during training. Consequently, training data comprises instances translated from English to the target language using NLLB.

For fusion, we obtain the required 'silver' parallel data also through MT (using NLLB). The training set consists of parallel sets of English and MT-ed instances, whereas the validation and test sets consist of parallel target language instances and corresponding machine translations into English. We posit that the assumed absence of gold translations both during training and during inference is the most realistic evaluation of FLARE models.

Finally, to compare representation-level fusion with input-level fusion, we append source and target language texts in the input prompt of the mPLM, effectively doubling the sequence length

| Model | XNLI | TyDiQA | NusaX | Avg. |
|---|---|---|---|---|
| ***Zero-Shot Cross-Lingual Transfer*** *(models are trained on English data)* | | | | |
| XLM-R Base | 74.08 | 48.90 / 36.64 | 58.07 | 58.31 |
| XLM-R Large | 78.40 | 65.08 / 54.22 | 76.41 | 71.49 |
| mT5-XL | 80.50 | 64.11 / 50.76 | 72.88 | 70.27 |
| ***Translate-Test*** *(test data is translated to English)* | | | | |
| XLM-R Base | 75.25 | 48.76 / 36.79 | 76.30 | 64.77 |
| XLM-R Large | 77.04 | 65.51 / 54.17 | 75.80 | 70.89 |
| mT5-XL | 79.13 | 64.36 / 51.18 | 75.48 | 70.79 |
| ***Translate-Train*** *(models are trained on training data translated to the target language)* | | | | |
| XLM-R Base | 77.30 | 50.09 / 37.66 | 71.04 | 64.07 |
| w/ input-level fusion | 74.58 | - | 76.68 | - |
| w/ FLARE MT | 77.14 | 48.94 / 37.20 | 72.05 | **64.09** |
| w/ FLARE | 73.40 | 50.04 / 37.74 | 73.61 | 63.63 |
| XLM-R Large | 79.40 | 65.20 / 53.74 | 77.47 | 72.11 |
| w/ input-level fusion | 78.48 | - | 78.39 | - |
| w/ FLARE MT | 81.45 | 65.34 / 53.98 | 77.16 | 72.76 |
| w/ FLARE | 80.23 | 65.30 / 54.30 | 79.99 | **73.34** |
| mT5-XL | 82.99 | 64.87 / 52.42 | 80.58 | 74.07 |
| w/ input-level fusion | 79.66 | - | 74.87 | - |
| w/ FLARE MT | 82.84 | 64.78 / 52.12 | 80.41 | 73.90 |
| w/ FLARE | 83.25 | 64.90 / 52.48 | 80.72 | **74.22** |

Table 1: Average accuracy results across languages included in the XNLI, TyDiQA, and NusaX datasets. Performance metrics are: Accuracy for XNLI, F1/Exact Match for TyDiQA, and Micro F1 for NusaX. FLARE results are based on the best performing fusion functions:*add relu* for XNLI and NusaX, as well as *mul* on TyDiQA.

(Kim et al., 2023; Villa-Cueva et al., 2024).[5]

### 4.2 Evaluation Tasks and Datasets

**XNLI** consists of machine-translated sentence pairs that are translated from English to 15 languages (Conneau et al., 2018). The task involves determining whether a sentence entails, contradicts, or is neutral to a given premise.

**NusaX** is a human-annotated sentiment classification dataset that spans 11 Indonesian languages, including low-resource languages (Winata et al., 2023). With 500 labeled instances for each language, the dataset evaluates few-shot adaptation.

**TyDiQA-GoldP** is a human-annotated extractive QA dataset covering 8 languages (Clark et al., 2020). The task is to identify the answer spans within the context passages.

Additional information on evaluation languages and datasets used for source language fine-tuning are available in Table 9 in the appendix.

### 4.3 Machine Translations

We utilize NLLB's 3.3B variant (Team et al., 2022) as the main MT model, with greedy decoding to obtain translations (Artetxe et al., 2023). To ensure consistency in our experimental setup, we also translate languages that are not directly supported by NLLB. Specifically, Madurese (mad) and Ngaju (nij) are translated using the Indonesian language identifier, as these languages are not supported by NLLB[6] (Winata et al., 2023).

For translating extractive QA datasets, we enclose the answer spans within marker tokens prior to translation with NLLB (Chen et al., 2023). This method allows us to determine the position of the translated answer spans by locating these marker tokens in the translated text. Instances that fail to retain the answer span marker tokens in the translated output are excluded from the evaluation process.

| Model | XNLI | NusaX |
|---|---|---|
| *Translate-Train (fusion models are trained on training data translated to the target language and evaluated using* ***gold translations*** *in the source language)* | | |
| XLM-R Base | | |
| w/ input-level fusion | 84.63 | 87.87 |
| w/ FLARE | 84.62 | 75.43 |
| XLM-R Large | | |
| w/ input-level fusion | 87.19 | 90.93 |
| w/ FLARE | 88.15 | 84.66 |
| mT5-XL | | |
| w/ input-level fusion | 89.67 | 90.57 |
| w/ FLARE | 86.57 | 80.72 |

Table 2: Average scores for the *translate-train* setting with gold English translations during inference across languages included in the XNLI, and NusaX datasets, representing optimal translation quality.

[5]The context length for input-level fusion models is doubled. Due to memory and context length limitations, these models could not be evaluated for TyDiQA; see later.

[6]We note that Toba Batak (bbc) is unsupported by NLLB and excluded from the evaluation due to translation artifacts resulting in random classification performance.

| Fusion Function | TyDiQA | NusaX |
|---|---|---|
| *Translate-Train (models are trained on data translated to the target language)* | | |
| add | 65.04 / 53.48 | 79.89 |
| mul | 65.30 / 54.30 | 78.78 |
| add+relu | 65.06 / 53.78 | 79.99 |
| cross-attention | 65.04 / 53.64 | 78.17 |

Table 3: Average scores for different fusion functions based on XLM-R Large using FLARE.

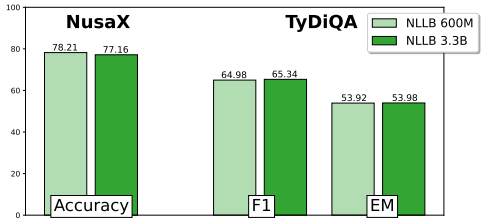


Figure 4: Average performance differences on NusaX and TyDiQA for XLM-R Large using FLARE MT with MT models of different size.

## 5 Results and Discussion

**Main Results** displayed in Table 1 confirm our hypothesis that task-specific knowledge can be efficiently transferred from English to other languages within adapter bottlenecks. FLARE consistently surpasses the zero-shot, translate-test, and translate-train baselines across various tasks, demonstrating robust performance with machine-translated training data in the target language and machine-translated source language data during inference. Moreover, the results from the few-shot adaptation scenario on NusaX suggest that FLARE does not require extensive labeled task data. It improves downstream performance in few-shot settings on lower-resource languages. While input-level fusion shows competitive results for XLM-R Base, both translate-train and FLARE outperform input-level fusion for larger mPLMs. Beyond performance benefits, FLARE reduces the average training time on XNLI by more than 30% when compared to input-level fusion.

When comparing FLARE with the FLARE MT variant which utilizes latent translations, it becomes evident that the mPLM's task-specific enriched source representations enhance downstream performance. In settings where extracting task-specific knowledge for the source representations from the mPLM is challenging, such as when faced with issues of translation quality, the richer translation information from the MT model's encoder representations can enhance downstream performance.

**Impact of Translation Quality.** Translation quality is an important factor when combining source and target language representations. The results in Table 2 show the upper performance boundary when fusion models are exposed to gold translations during inference. Providing gold translations for source language inputs closes the cross-lingual transfer gap, matching source language performance. This demonstrates that FLARE can optimally combine the available information, and its performance scales with translation quality, which is the most decisive factor for downstream performance across fusion models.

For extractive QA tasks, MT quality limits model applicability. For these tasks, FLARE matches or surpasses the translate-train and translate-test baselines, indicating that the lower performance boundary matches strong baselines. In contrast, performance of input-level fusion substantially deteriorates when evaluated using machine-translated inputs, underscoring its reliance on the quality of English text inputs. Yet, when provided with gold translation data, input-level fusion matches or exceeds source language performance.

The FLARE MT variant relies on latent translations, which contain rich translation information. Consequently, the MT model size, serving as a proxy for translation quality, has a lower impact on performance. The results in Figure 4 show that performance with the NLLB 600M variant is comparable to, or even better than, that with NLLB 3.3B. This suggests that down-projecting latent translations may incur information loss.

**Impact of Fusion Function.** Table 3 presents the average performance of fusion functions inside LoRAs of XLM-R Large. The results suggest that adding non-linearity to the fusion functions does not provide decisive performance benefits over simpler linear transformations. Notably, the functions *add*, *mul*, and *add+relu* show the best performance. Despite the additional parameters available in cross-attention, the technique does not yield superior downstream performance. In sum, given that the optimal fusion function appears to be task-dependent, these functions can be regarded as hyperparameters that can also be fine-tuned based on validation data.

**Impact of Adapter Capacity.** Increasing the bottleneck size within LoRA improves FLARE's performance, albeit with diminishing returns for

| Model | $r$ | TyDiQA | NusaX |
|---|---|---|---|
| *Translate-Train (models are trained on training data translated to the target language)* | | | |
| XLM-R Base | 8 | 51.05 / 38.11 | 63.40 |
| w/ FLARE | | 51.12 / 39.08 | 66.46 |
| XLM-R Large | | 64.87 / 53.81 | 77.79 |
| w/ FLARE | | 65.03 / 53.96 | 79.21 |
| XLM-R Base | 64 | 50.09 / 37.66 | 70.29 |
| w/ FLARE | | 50.04 / 37.74 | 73.61 |
| XLM-R Large | | 65.20 / 53.74 | 77.47 |
| w/ FLARE | | 65.30 / 54.30 | 79.99 |
| XLM-R Base | 128 | 49.77 / 37.78 | 70.46 |
| w/ FLARE | | 50.42 / 38.63 | 73.12 |
| XLM-R Large | | 65.26 / 53.97 | 77.36 |
| w/ FLARE | | 66.18 / 55.46 | 79.35 |

Table 4: Average scores for varying adapter bottleneck size $r$ in LoRA; based on XLM-R Large, using FLARE with the *add+relu* fusion function (Section 3.2).

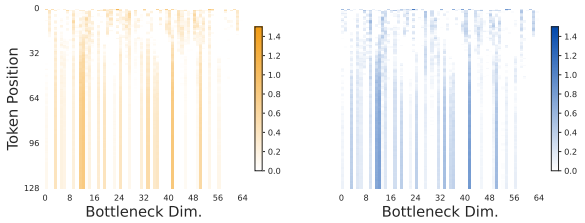


Figure 5: Average activation values for English and Acehnese in the first bottleneck query layer in XLM-R Large for the NusaX test set; *add+relu* fusion.

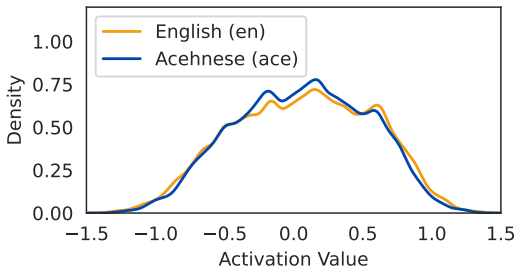


Figure 6: Average activations in the bottleneck adapters across all XLM-R Large layers for the NusaX test set.

$r = 128$ on datasets like NusaX; see Table 4. This suggests that even highly compressed language representations are sufficient to facilitate cross-lingual transfer in the representation space. Moreover, the required adapter capacity is dependent on task complexity: more complex tasks require finer-grained representations for optimal fusion performance. Despite the incremental performance gains with larger adapter capacities, even LoRA adapters with $r = 8$ already yield considerable benefits with FLARE. Interestingly, FLARE can leverage larger adapter capacities more effectively compared to regular LoRA adapters without fusion.

**Layer-wise Language Activation.** Figure 6 shows that the magnitudes of source and target language activations across the entire XLM-R Large are comparable. This indicates that FLARE does not overly rely on either source or target representations during fusion. Further, Figure 5 displays the average activations for English and Acehnese in the first adapter bottleneck: this confirms that both source and target languages maintain similar activation magnitudes. Hence, subsequent Acehnese representations are infused with the English representations from this initial transfer, integrating balanced source and target language information. Detailed activations for individual instances are illustrated in Figure 7, which show positional activation differences and demonstrate the alignment of source and target languages for information transfer.

**On Latent MT Fusion.** FLARE MT outperforms zero-shot and translate-test baselines and shows competitive performance with regular FLARE. This indicates that errors in discrete translations directly affect downstream performance. In contrast to regular FLARE, the MT encoder representations used in FLARE MT include task-agnostic language information, and therefore do not transfer task knowledge to the target languages. Nonetheless, it provides a resource-efficient alternative to regular FLARE by avoiding the need for decoding in the MT and eliminating the forward pass in the mPLM, making it especially valuable in scenarios where translation quality is limited. The detailed results for XNLI in Table 6 (appendix) show that FLARE MT is particularly beneficial for lower-resource languages, such as Swahili and Urdu, compared to FLARE, when exposed to large amounts of training data.

## 6 Conclusion

We introduced Fusion for Language Representations (FLARE) for parameter-efficient cross-lingual transfer (XLT). Our experimental results demonstrated that FLARE outperforms strong XLT baselines on natural language understanding tasks. With gold translations, it matches model performance in English, while not reducing performance below translate-train baselines for lower-quality translations. We also showed that FLARE is representation-agnostic: it can directly incorporate latent translations from an MT model in place of translated English text. This further improves resource-efficiency and enhances knowledge transfer for lower-quality translations.

## 7 Limitations

The proposed FLARE method by design relies on textual data availability for both source and target languages. In practical scenarios, such as the standard translate-train settings evaluated in our study, machine translation models are utilized to generate either the source or target language inputs. Consequently, the performance of FLARE is dependent upon the quality of these machine translations, as we also investigated empirically in this work. This dependency poses some significant challenges, particularly for tasks that require precise positional alignment, like extractive question-answering, where the quality of machine translations affects downstream performance and model applicability.

Furthermore, our evaluation exclusively employs English as the high-resource source language for representation fusion. While English is predominantly used in mPLM pretraining corpora, exploring other high-resource languages that share linguistic similarities with the target languages could potentially yield similar or improved cross-lingual transfer performance.

Finally, our choice of base multilingual LMs has been motivated by the current state-of-the-art (SotA) in the field of multilingual NLP and XLT to low-resource languages for NLU tasks. The main models are SotA encoder-only (XLM-R) and encoder-decoder mPLMs (mT5), while we also provide some preliminary results with a representative decoder-only LLM (Llama 3). However, we note that the LLM technology and its adaptation to XLT for NLU in lower-resource languages has not been proven to be fully mature yet (Lin et al., 2024; Razumovskaia et al., 2024).

## References


AI@Meta. 2024. Introducing meta llama 3.

Alan Ansell, Edoardo Ponti, Anna Korhonen, and Ivan Vulić. 2022. Composable sparse fine-tuning for cross-lingual transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1778–1796, Dublin, Ireland. Association for Computational Linguistics.

Mikel Artetxe, Vedanuj Goswami, Shruti Bhosale, Angela Fan, and Luke Zettlemoyer. 2023. Revisiting machine translation for cross-lingual classification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6489–6499, Singapore. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2020. Translation artifacts in cross-lingual transfer learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7674–7684, Online. Association for Computational Linguistics.

Tingfeng Cao, Chengyu Wang, Chuanqi Tan, Jun Huang, and Jinhui Zhu. 2023. Sharing, teaching and aligning: Knowledgeable transfer learning for cross-lingual machine reading comprehension. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 455–467, Singapore. Association for Computational Linguistics.

Yang Chen, Chao Jiang, Alan Ritter, and Wei Xu. 2023. Frustratingly easy label projection for cross-lingual transfer. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5775–5796, Toronto, Canada. Association for Computational Linguistics.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

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.

Yuwei Fang, Shuohang Wang, Zhe Gan, Siqi Sun, and Jingjing Liu. 2021. Filter: An enhanced fusion method for cross-lingual language understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12776–12784.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Sunkyoung Kim, Dayeon Ki, Yireun Kim, and Jinsik Lee. 2023. Boosting cross-lingual transferability in multilingual models via in-context learning. *Preprint*, arXiv:2305.15233.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online. Association for Computational Linguistics.

En-Shiun Lee, Sarubi Thillainathan, Shravan Nayak, Surangika Ranathunga, David Adelani, Ruisi Su, and Arya McCarthy. 2022a. Pre-trained multilingual sequence-to-sequence models: A hope for low-resource language translation? In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 58–67, Dublin, Ireland. Association for Computational Linguistics.

Jaeseong Lee, Seung-won Hwang, and Taesup Kim. 2022b. FAD-X: Fusing adapters for cross-lingual transfer to low-resource languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 57–64, Online only. Association for Computational Linguistics.

Peiqin Lin, Shaoxiong Ji, Jörg Tiedemann, André F. T. Martins, and Hinrich Schütze. 2024. MaLA-500: Massive language adaptation of large language models. *Preprint*, arXiv:2401.13303.

Jaehoon Oh, Jongwoo Ko, and Se-Young Yun. 2022. Synergy with translation artifacts for training and inference in multilingual tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6747–6754, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Edoardo Maria Ponti, Julia Kreutzer, Ivan Vulić, and Siva Reddy. 2021. Modelling latent translations for cross-lingual transfer. *Preprint*, arXiv:2107.11353.

Ayu Purwarianti and Ida Ayu Putu Ari Crisdayanti. 2019. Improving bi-lstm performance for indonesian sentiment analysis using paragraph vector. In *2019 International Conference of Advanced Informatics: Concepts, Theory and Applications (ICAICTA)*, pages 1–5.

Tingyu Qu, Tinne Tuytelaars, and Marie-Francine Moens. 2024. Introducing routing functions to vision-language parameter-efficient fine-tuning with low-rank bottlenecks. *Preprint*, arXiv:2403.09377.

Kiran Ramnath, Leda Sari, Mark Hasegawa-Johnson, and Chang Yoo. 2021. Worldly wise (WoW) - cross-lingual knowledge fusion for fact-based visual spoken-question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1908–1919, Online. Association for Computational Linguistics.

Evgeniia Razumovskaia, Ivan Vulić, and Anna Korhonen. 2024. Analyzing and adapting large language models for few-shot multilingual NLU: are we there yet? *Preprint*, arXiv:2403.01929.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019. A survey of cross-lingual word embedding models. *J. Artif. Int. Res.*, 65(1):569–630.

Lei Shi, Rada Mihalcea, and Mingjun Tian. 2010. Cross language text classification by model translation and semi-supervised learning. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1057–1067, Cambridge, MA. Association for Computational Linguistics.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. *Preprint*, arXiv:2402.16438.

Eshaan Tanwar, Subhabrata Dutta, Manish Borthakur, and Tanmoy Chakraborty. 2023. Multilingual LLMs are better cross-lingual in-context learners with alignment. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6292–6307, Toronto, Canada. Association for Computational Linguistics.

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. *Preprint*, arXiv:2207.04672.

Inigo Jauregi Unanue, Gholamreza Haffari, and Massimo Piccardi. 2023. T3L: Translate-and-Test Transfer Learning for Cross-Lingual Text Classification. *Transactions of the Association for Computational Linguistics*, 11:1147–1161.

Emilio Villa-Cueva, A. Pastor López-Monroy, Fernando Sánchez-Vega, and Thamar Solorio. 2024. Adaptive cross-lingual text classification through in-context one-shot demonstrations. *Preprint*, arXiv:2404.02452.

Yaqing Wang, Sahaj Agarwal, Subhabrata Mukherjee, Xiaodong Liu, Jing Gao, Ahmed Hassan Awadallah, and Jianfeng Gao. 2022. AdaMix: Mixture-of-adaptations for parameter-efficient model tuning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5744–5760, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in english? on the latent language of multilingual transformers. *Preprint*, arXiv:2402.10588.

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.

Genta Winata, Shijie Wu, Mayank Kulkarni, Thamar Solorio, and Daniel Preotiuc-Pietro. 2022. Cross-lingual few-shot learning on unseen languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 777–791, Online only. Association for Computational Linguistics.

Genta Indra Winata, Alham Fikri Aji, Samuel Cahyawijaya, Rahmad Mahendra, Fajri Koto, Ade Romadhony, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, Pascale Fung, Timothy Baldwin, Jey Han Lau, Rico Sennrich, and Sebastian Ruder. 2023. NusaX: Multilingual parallel sentiment dataset for 10 Indonesian local languages. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 815–834, Dubrovnik, Croatia. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

Shaoyang Xu, Junzhuo Li, and Deyi Xiong. 2023. Language representation projection: Can we transfer factual knowledge across languages in multilingual language models? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3692–3702, Singapore. 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.

Huiyun Yang, Huadong Chen, Hao Zhou, and Lei Li. 2022. Enhancing cross-lingual transfer by manifold mixup. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*.

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 Detailed Evaluation Results

**Figure 7** displays average activations within the first adapter bottlenecks in the XLM-R Large model using FLARE and the *add+relu* fusion function. This visualization highlights the positional alignment process between English and Acehnese token representations, with varying activation values across different sequence positions reflecting the dynamics of language representation fusion.

**Table 6** shows the results for the XNLI dataset for each language in zero-shot XLT, translate-test, translate-train settings, including translate-train with gold translations in the source language. The results confirm that FLARE consistently improves XTL performance in the translate-train setting across different languages without particular bias towards typological relatedness to English or frequency in pretraining corpora.

**Table 7** details the results for the TyDiQA dataset for each language in the zero-shot XLT, translate-test, and translate-train settings. The outcomes demonstrate that FLARE performance extends to tasks including positional information, such as extractive question-answering.

**Table 8** outlines the performance for the NusaX dataset for each language in zero-shot XLT, translate-test, translate-train, and translate-train settings with gold translations in the source language. Even with few training samples, our FLARE method demonstrates consistent performance improvements across the low-resource languages included in the NusaX dataset.

## B Training Details

Our evaluation results are averaged across *two random seeds*. Initially, we fully fine-tune XLM-R Base and XLM-R Large models on English task data. For mT5-XL, fine-tuning is conducted using LoRA adapters set with $r = 64$ and $\alpha = 128$, which are subsequently integrated into the model's weights prior to task fine-tuning in the target languages. Hyperparameter configurations for full-tuning each mPLM are provided in Table 5.

The total computation time for the experimental results exceeds 4,000 GPU hours. Experiments are conducted on NVIDIA A100 and H100 GPUs. All models are trained using half-precision.

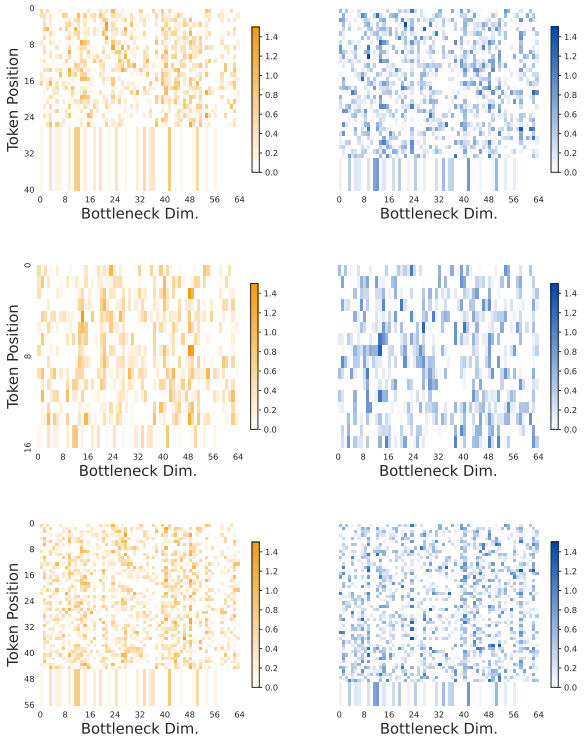


Figure 7: Activation values for individual instances included in the NusaX test set. English and Acehnese activation values are extracted from the first bottleneck query layer in XLMR-Large, which is trained with the *add+relu* fusion function.

## C Another Ablation: Representation Fusion during Training Only

To investigate the importance of utilizing source language representations during inference, we modified FLARE to restrict representation fusion to the training phase only. Specifically, we limited the fusion with source language representations to 50% of the training instances and excluded source language data during inference. This evaluates cross-lingual transfer capabilities based on instance-independent patterns learned from source language representations during training. Our findings reveal that fusion adapters struggle to learn patterns that are independent of specific instances from source language representations during training. As a result, when implemented in the XLM-R Large model on the NusaX test set, the performance of the *train-only* FLARE variant decreased by 30%. Crucially, this significant drop underscores the importance of incorporating source language representations during inference to achieve effective cross-lingual adaptation.

| Model | Hparam | Values |
|---|---|---|
| XLMR-Base | epochs | {10, 10, 20} |
| | batch size | 32 |
| | sequence length | {128, 512, 128} |
| | learning rate | 2e-5 |
| XLMR-Large | epochs | {10, 10, 20} |
| | batch size | 32 |
| | sequence length | {128, 512, 128} |
| | learning rate | 2e-5 |
| mT5-XL | epochs | {10, 10, 40} |
| | batch size | 64 |
| | sequence length | {128, 512, 128} |
| | learning rate | 2e-4 |

Table 5: Hyperparameter configurations for each mPLM across the XNLI, TyDiQA, and NusaX datasets. Values listed in curly braces represent the specific settings used for each dataset in sequential order: {XNLI, TyDiQA, NusaX}.

## D Preliminary Results on LLaMA 3

To validate whether the performance benefits of FLARE extend to decoder-only model architectures, we conducted further experiments using the LLaMA 3 8B model (AI@Meta, 2024). We finetune the base model for 3 epochs on the English NusaX dataset using LoRA adapters ($r = 64$, $\alpha = 128$). Subsequently, we adapted this base model to the Indonesian languages included in the NusaX dataset using FLARE in the *translate-train* setting. We train the base model for 3 epochs with a learning rate of $2e^{-4}$, maintaining the same LoRA configurations as mentioned above. The results demonstrate that language representation fusion is effectively applicable to decoder-only architectures. The LLaMA 3 8B model with FLARE achieved an F1-Score of 76.58, surpassing the *translate-test* performance, which was 76.18. While these outcomes are in line with our findings for other model architectures, they suggest that the LLaMA 3 base model is undertrained and it has not been created to support multilingual NLP applications directly. Consequently, the results are not directly comparable with those included in Table 1, and they warrant further investigation (also related to improving target language adaptation of LLMs in general), which is out of scope of this particular work.

| Model | en | ar | bg | de | el | es | fr | hi | ru | sw | th | tr | ur | vi | zh | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Zero-Shot Cross-lingual Transfer* | | | | | | | | | | | | | | | | |
| XLM-R Base | 84.83 | 72.18 | 77.23 | 76.85 | 75.67 | 78.92 | 78.20 | 70.06 | 75.17 | 64.27 | 71.52 | 72.42 | 65.95 | 74.69 | 73.27 | 74.08 |
| XLM-R Large | 87.94 | 77.45 | 81.72 | 81.26 | 81.58 | 82.99 | 82.40 | 74.31 | 78.66 | 67.70 | 76.29 | 76.95 | 69.88 | 78.30 | 78.60 | 78.40 |
| mT5-XL | 90.00 | 79.72 | 84.01 | 83.55 | 83.05 | 84.87 | 84.49 | 77.72 | 81.38 | 75.85 | 77.70 | 80.46 | 73.85 | 79.90 | 80.48 | 80.50 |
| *Translate-Test (translate test data to English using NLLB 3.3B)* | | | | | | | | | | | | | | | | |
| XLM-R Base | 84.83 | 74.69 | 77.74 | 78.86 | 78.42 | 80.48 | 78.86 | 72.87 | 76.25 | 70.90 | 71.60 | 76.41 | 66.85 | 76.57 | 72.99 | 75.25 |
| XLM-R Large | 87.94 | 76.51 | 81.28 | 81.12 | 81.36 | 82.48 | 81.74 | 74.73 | 77.58 | 71.88 | 71.96 | 77.62 | 68.52 | 77.90 | 73.95 | 77.04 |
| mT5-XL | 90.00 | 79.06 | 83.17 | 83.29 | 82.71 | 84.09 | 83.49 | 76.67 | 80.54 | 73.15 | 74.69 | 79.64 | 69.80 | 80.26 | 77.31 | 79.13 |
| *Translate-Train (models are trained on training data translated to the target language)* | | | | | | | | | | | | | | | | |
| XLM-R Base | 84.83 | 75.43 | 79.50 | 79.08 | 78.18 | 80.68 | 79.90 | 73.11 | 78.20 | 79.90 | 76.87 | 75.67 | 69.40 | 78.06 | 78.22 | 77.30 |
| w/ input-level fusion | 84.83 | 74.73 | 77.80 | 78.08 | 76.59 | 78.94 | 78.38 | 71.22 | 75.37 | 70.32 | 72.26 | 75.93 | 65.71 | 75.79 | 72.97 | 74.58 |
| w/ FLARE MT | 84.83 | 76.71 | 80.34 | 79.72 | 78.46 | 80.74 | 80.00 | 73.73 | 78.62 | 71.20 | 77.09 | 75.89 | 70.50 | 78.16 | 78.84 | 77.14 |
| w/ FLARE | 84.83 | 75.03 | 75.51 | 78.36 | 74.47 | 76.77 | 78.50 | 69.68 | 74.63 | 70.64 | 69.42 | 72.63 | 64.89 | 74.03 | 73.07 | 73.40 |
| XLM-R Large | 87.94 | 76.51 | 83.95 | 81.12 | 82.91 | 84.31 | 81.74 | 78.12 | 81.26 | 71.88 | 78.54 | 80.56 | 75.03 | 81.76 | 73.95 | 79.40 |
| w/ input-level fusion | 87.94 | 79.60 | 82.30 | 81.46 | 82.36 | 83.77 | 81.26 | 76.85 | 78.70 | 73.15 | 74.85 | 80.06 | 70.70 | 79.40 | 74.19 | 78.48 |
| w/ FLARE MT | 87.94 | 80.90 | 84.91 | 83.75 | 83.97 | 85.03 | 83.91 | 78.20 | 82.16 | 76.39 | 80.72 | 81.26 | 75.17 | 81.60 | 82.34 | 81.45 |
| w/ FLARE | 87.94 | 79.94 | 83.59 | 82.46 | 82.75 | 84.39 | 82.44 | 80.72 | 78.10 | 73.65 | 78.10 | 80.06 | 73.99 | 80.66 | 82.34 | 80.23 |
| mT5-XL | 90.00 | 82.24 | 85.95 | 85.43 | 85.33 | 86.29 | 86.23 | 80.62 | 83.91 | 78.96 | 81.12 | 83.01 | 77.07 | 82.81 | 82.95 | 82.99 |
| w/ input-level fusion | 90.00 | 79.74 | 83.65 | 83.65 | 82.95 | 84.51 | 83.81 | 77.50 | 81.18 | 73.47 | 75.93 | 79.82 | 70.74 | 80.70 | 77.54 | 79.66 |
| w/ FLARE MT | 90.00 | 82.50 | 85.77 | 85.49 | 84.95 | 85.85 | 85.61 | 80.48 | 83.39 | 79.06 | 80.66 | 83.17 | 77.50 | 82.53 | 82.79 | 82.84 |
| w/ FLARE | 90.00 | 82.59 | 86.43 | 85.61 | 85.39 | 86.13 | 86.17 | 81.08 | 84.09 | 79.58 | 81.10 | 83.41 | 77.52 | 83.32 | 83.11 | 83.25 |
| *Translate-Train (fusion models are trained on training data translated to the target language and evaluated using gold translations in the source language)* | | | | | | | | | | | | | | | | |
| XLM-R Base w/ input-level fusion | 84.83 | 84.85 | 84.79 | 84.79 | 84.71 | 84.67 | 84.25 | 84.63 | 84.31 | 84.53 | 84.63 | 84.51 | 84.87 | 84.75 | 84.47 | 84.63 |
| w/ FLARE | 84.83 | 84.63 | 84.63 | 84.53 | 84.67 | 84.55 | 84.57 | 84.35 | 84.39 | 84.65 | 84.87 | 84.87 | 84.79 | 84.67 | 84.49 | 84.62 |
| XLM-R Large w/ input-level fusion | 87.94 | 88.41 | 88.54 | 88.46 | 88.36 | 88.28 | 88.02 | 88.38 | 85.91 | 86.23 | 85.91 | 85.85 | 86.05 | 85.85 | 86.45 | 87.19 |
| w/ FLARE | 87.94 | 88.10 | 88.06 | 88.04 | 88.12 | 88.02 | 88.08 | 88.40 | 88.12 | 88.46 | 88.16 | 88.14 | 88.22 | 88.04 | 88.16 | 88.15 |
| mT5-XL w/ input-level fusion | 90.00 | 90.04 | 89.80 | 89.54 | 89.70 | 89.78 | 89.50 | 89.80 | 89.52 | 89.56 | 89.84 | 89.66 | 89.38 | 89.52 | 89.70 | 89.67 |
| FLARE | 90.00 | 88.62 | 88.74 | 88.80 | 85.34 | 87.83 | 86.19 | 84.31 | 86.12 | 89.66 | 88.49 | 89.56 | 79.22 | 85.33 | 83.73 | 86.57 |

Table 6: Average scores per language in the XNLI dataset. Model performance is evaluated using the Accuracy metric.

| Model | en | ar | ben | fi | ind | ko | ru | sw | tel | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|
| *Zero-Shot Cross-lingual Transfer* | | | | | | | | | | |
| XLM-R Base | 62.17 / 51.36 | 51.35 / 35.94 | 50.31 / 36.67 | 42.48 / 27.39 | 58.07 / 46.95 | 40.26 / 28.77 | 41.41 / 29.97 | 58.66 / 47.44 | 48.62 / 40.00 | 48.90 / 36.64 |
| XLM-R Large | 72.26 / 57.96 | 62.78 / 51.25 | 72.14 / 60.00 | 51.31 / 37.90 | 68.34 / 58.84 | 53.49 / 40.03 | 56.20 / 44.03 | 69.59 / 59.04 | 86.79 / 82.63 | 65.08 / 54.22 |
| mT5-XL | 74.97 / 61.14 | 58.26 / 40.31 | 71.00 / 53.33 | 55.21 / 40.13 | 67.76 / 55.93 | 53.77 / 41.03 | 56.00 / 43.30 | 68.64 / 57.02 | 82.26 / 75.00 | 64.11 / 50.76 |
| *Translate-Test (translate test data to English using NLLB 3.3B)* | | | | | | | | | | |
| XLM-R Base | 62.17 / 51.36 | 51.17 / 36.25 | 47.98 / 35.00 | 42.59 / 27.39 | 58.43 / 47.26 | 40.84 / 30.77 | 41.72 / 30.24 | 58.70 / 47.44 | 48.61 / 40.00 | 48.76 / 36.79 |
| XLM-R Large | 72.26 / 57.96 | 62.40 / 50.62 | 73.25 / 60.00 | 52.62 / 38.54 | 68.78 / 58.84 | 55.09 / 41.03 | 55.64 / 42.97 | 69.38 / 58.70 | 86.91 / 82.63 | 65.51 / 54.17 |
| mT5-XL | 74.97 / 61.14 | 58.23 / 40.00 | 71.00 / 53.33 | 54.99 / 40.13 | 69.76 / 57.93 | 53.77 / 41.03 | 56.72 / 44.03 | 68.22 / 58.02 | 82.15 / 75.00 | 64.36 / 51.18 |
| *Translate-Train (models are trained on training data translated to the target language)* | | | | | | | | | | |
| XLM-R Base | 62.17 / 51.36 | 52.08 / 36.88 | 50.78 / 38.33 | 44.48 / 27.39 | 58.96 / 47.56 | 40.41 / 29.06 | 42.56 / 31.30 | 59.54 / 48.12 | 51.94 / 42.63 | 50.09 / 37.66 |
| w/ FLARE MT | 62.17 / 51.36 | 51.36 / 37.19 | 48.19 / 35.00 | 43.13 / 28.03 | 59.78 / 48.48 | 38.48 / 29.06 | 42.68 / 31.30 | 58.75 / 47.78 | 49.18 / 40.79 | 48.94 / 37.20 |
| w/ FLARE | 62.17 / 51.36 | 52.37 / 37.19 | 49.87 / 36.67 | 43.98 / 27.71 | 59.67 / 48.78 | 40.45 / 29.92 | 42.55 / 30.24 | 58.88 / 47.44 | 52.53 / 43.95 | 50.04 / 37.74 |
| XLM-R Large | 72.26 / 57.96 | 61.77 / 49.38 | 72.44 / 58.33 | 51.90 / 39.17 | 68.07 / 59.45 | 55.46 / 41.03 | 55.57 / 42.71 | 68.78 / 57.00 | 87.61 / 82.89 | 65.20 / 53.74 |
| w/ FLARE MT | 72.26 / 57.96 | 61.99 / 48.75 | 71.58 / 58.33 | 52.62 / 37.90 | 68.39 / 59.76 | 56.42 / 43.59 | 56.17 / 43.24 | 68.60 / 57.34 | 86.95 / 82.89 | 65.34 / 53.98 |
| w/ FLARE | 72.26 / 57.96 | 61.77 / 49.69 | 71.17 / 58.33 | 52.55 / 39.81 | 68.35 / 60.06 | 56.10 / 41.88 | 55.69 / 43.24 | 69.02 / 57.68 | 87.78 / 83.68 | 65.30 / 54.30 |
| mT5-XL | 74.97 / 61.14 | 60.34 / 43.44 | 70.01 / 58.33 | 54.44 / 38.85 | 68.63 / 57.01 | 56.46 / 45.30 | 55.94 / 41.11 | 69.56 / 58.70 | 83.54 / 76.58 | 64.87 / 52.42 |
| w/ FLARE MT | 74.97 / 61.14 | 59.96 / 43.15 | 69.86 / 56.66 | 55.63 / 39.52 | 68.80 / 57.11 | 56.17 / 44.97 | 55.63 / 42.24 | 69.29 / 57.53 | 82.94 / 75.82 | 64.78 / 52.12 |
| w/ FLARE | 74.97 / 61.14 | 57.55 / 41.25 | 71.95 / 57.67 | 55.97 / 40.45 | 68.88 / 57.01 | 56.61 / 45.36 | 56.31 / 43.77 | 68.48 / 57.68 | 83.41 / 76.65 | 64.90 / 52.48 |

Table 7: Average scores per language in the TyDiQA dataset. Model performance is evaluated using the F1 / EM metrics.

| Model | en | ace | ban | bjn | bug | ind | jav | mad | min | nij | sun | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Zero-Shot Cross-lingual Transfer* | | | | | | | | | | | | |
| XLM-R Base | 91.00 / 90.00 | 55.75 / 52.64 | 64.25 / 61.87 | 69.00 / 66.01 | 43.00 / 32.49 | 89.50 / 88.62 | 75.25 / 71.73 | 50.75 / 42.75 | 63.50 / 61.28 | 51.75 / 43.75 | 62.25 / 59.60 | 62.50 / 58.07 |
| XLM-R Large | 91.75 / 91.21 | 70.00 / 69.53 | 78.25 / 77.78 | 81.50 / 80.77 | 50.75 / 47.99 | 92.50 / 91.82 | 86.50 / 85.71 | 73.25 / 72.28 | 82.00 / 80.52 | 74.00 / 72.93 | 85.50 / 84.73 | 77.42 / 76.41 |
| mT5-XL | 90.75 / 90.21 | 73.25 / 72.93 | 77.00 / 76.24 | 80.25 / 79.62 | 34.75 / 30.84 | 91.50 / 90.61 | 88.25 / 87.35 | 61.75 / 61.38 | 78.50 / 77.90 | 65.00 / 65.16 | 87.50 / 86.73 | 73.78 / 72.88 |
| *Translate-Test (translate test data to English using NLLB 3.3B)* | | | | | | | | | | | | |
| XLM-R Base | 91.00 / 90.00 | 77.25 / 76.88 | 77.25 / 77.52 | 85.25 / 84.57 | 72.50 / 72.28 | 87.50 / 86.63 | 85.25 / 84.67 | 55.50 / 55.32 | 83.00 / 82.22 | 59.00 / 58.61 | 84.75 / 84.31 | 76.72 / 76.30 |
| XLM-R Large | 91.75 / 91.21 | 77.00 / 76.46 | 74.50 / 74.19 | 84.25 / 83.78 | 70.00 / 69.82 | 87.75 / 86.88 | 84.25 / 83.69 | 58.25 / 58.36 | 82.75 / 82.27 | 56.25 / 56.54 | 86.50 / 86.04 | 76.15 / 75.80 |
| mT5-XL | 90.75 / 90.21 | 76.75 / 76.18 | 74.00 / 73.43 | 82.25 / 81.55 | 69.50 / 69.13 | 87.00 / 85.99 | 84.25 / 83.50 | 61.00 / 60.63 | 83.00 / 82.32 | 59.50 / 59.37 | 83.50 / 82.68 | 76.08 / 75.48 |
| *Translate-Train (models are trained on training data translated to the target language)* | | | | | | | | | | | | |
| XLM-R Base | 91.00 / 90.00 | 67.50 / 67.48 | 71.50 / 70.73 | 79.50 / 78.38 | 65.25 / 65.29 | 88.25/ 86.48 | 83.50 / 81.52 | 55.00 / 48.54 | 80.75 / 79.35 | 60.00 / 55.95 | 78.00 / 76.77 | 72.92 / 71.04 |
| w/ input-level fusion | 91.00 / 90.00 | 80.75 / 79.97 | 73.00 / 72.67 | 82.50 / 80.45 | 72.75 / 72.28 | 90.50 / 89.63 | 85.50 / 84.66 | 63.25 / 62.80 | 82.75 / 82.09 | 58.75 / 56.96 | 85.75 / 85.25 | 77.55 / 76.68 |
| w/ FLARE MT | 91.00 / 90.00 | 73.25 / 73.28 | 70.00 / 69.19 | 80.00 / 78.81 | 63.50 / 63.03 | 90.75 / 89.58 | 84.00 / 82.23 | 57.00 / 52.80 | 81.75 / 80.10 | 57.00 / 53.03 | 79.25 / 78.47 | 73.65 / 72.05 |
| w/ FLARE | 91.00 / 90.00 | 72.00 / 71.93 | 75.00 / 74.13 | 80.25 / 78.77 | 75.50 / 74.77 | 89.00 / 87.30 | 82.50 / 80.41 | 58.75 / 55.32 | 80.50 / 78.86 | 61.25 / 57.33 | 78.50 / 77.28 | 75.32 / 73.61 |
| XLM-R Large | 91.75 / 91.21 | 73.00 / 72.29 | 77.50 / 76.14 | 83.00 / 81.74 | 63.00 / 60.73 | 90.50 / 89.25 | 87.75 / 86.51 | 74.75 / 73.14 | 82.00 / 80.51 | 73.50 / 71.14 | 84.25 / 83.22 | 78.92 / 77.47 |
| w/ input-level fusion | 91.75 / 91.21 | 78.00 / 77.63 | 75.00 / 74.72 | 82.25 / 81.77 | 71.50 / 71.12 | 90.25 / 89.38 | 90.00 / 89.44 | 67.00 / 66.07 | 79.00 / 78.31 | 68.00 / 67.69 | 88.25 / 87.75 | 78.92 / 78.39 |
| w/ FLARE MT | 91.75 / 91.21 | 73.50 / 72.59 | 77.25 / 76.84 | 82.50 / 81.95 | 56.00 / 53.88 | 91.50 / 90.33 | 87.75 / 86.50 | 72.50 / 71.56 | 84.00 / 82.67 | 73.00 / 70.66 | 85.75 / 84.67 | 78.38 / 77.16 |
| w/ FLARE | 91.75 / 91.21 | 79.25 / 78.77 | 78.75 / 78.15 | 81.50 / 80.32 | 70.25 / 70.08 | 92.00 / 91.27 | 88.75 / 87.85 | 73.50 / 72.67 | 86.25 / 85.31 | 74.00 / 72.07 | 84.00 / 83.42 | 80.82 / 79.99 |
| mT5-XL | 90.75 / 90.21 | 81.25 / 80.11 | 82.50 / 80.94 | 86.25 / 85.13 | 68.50 / 66.58 | 90.75 / 89.57 | 91.00 / 89.90 | 75.00 / 72.61 | 85.25 / 83.67 | 72.00 / 68.99 | 89.25 / 88.29 | 82.18 / 80.58 |
| w/ input-level fusion | 90.75 / 90.21 | 81.50 / 80.95 | 79.25 / 78.57 | 83.50 / 82.88 | 72.25 / 71.87 | 90.25 / 89.28 | 40.00 / 33.80 | 70.00 / 69.10 | 84.25 / 83.57 | 71.75 / 71.14 | 88.50 / 87.55 | 76.12 / 74.87 |
| w/ FLARE MT | 90.75 / 90.21 | 82.00 / 81.14 | 83.00 / 81.39 | 85.25 / 84.07 | 65.25 / 64.62 | 90.75 / 89.61 | 90.75 / 89.73 | 72.00 / 69.05 | 85.75 / 84.89 | 73.25 / 71.57 | 89.25 / 88.01 | 81.72 / 80.41 |
| w/ FLARE | 90.75 / 90.21 | 84.25 / 83.80 | 81.50 / 80.55 | 85.00 / 84.06 | 65.75 / 64.70 | 89.75 / 88.32 | 91.50 / 90.50 | 76.50 / 74.36 | 85.00 / 83.64 | 71.00 / 69.29 | 89.00 / 88.00 | 81.92 / 80.72 |
| *Translate-Train (fusion models are trained on training data translated to the target language and evaluated using gold translations in the source language)* | | | | | | | | | | | | |
| XLM-R Base w/ input-level fusion | 91.00 / 90.00 | 90.50 / 90.04 | 89.25 / 88.45 | 89.50 / 88.23 | 89.50 / 88.51 | 91.00 / 89.83 | 91.00 / 90.19 | 86.25 /84.76 | 89.75 / 88.55 | 84.75 / 82.79 | 88.50 / 87.38 | 89.00 / 87.87 |
| w/ FLARE | 91.00 / 90.00 | 75.25 / 74.79 | 78.75 / 77.66 | 81.00 / 79.66 | 87.25 / 85.84 | 90.75 / 89.52 | 84.00 / 82.06 | 57.00 / 51.11 | 81.75 / 80.25 | 61.25 / 57.10 | 77.50 / 76.31 | 77.45 / 75.43 |
| XLM-R Large w/ input-level fusion | 91.75 / 91.21 | 91.75 / 91.24 | 91.75 / 91.08 | 91.25 / 90.55 | 91.25 / 90.69 | 92.50 / 91.99 | 91.50 / 90.88 | 91.75 / 91.23 | 91.75 / 91.07 | 90.75 / 90.07 | 91.00 / 90.52 | 91.52 / 90.93 |
| w/ FLARE | 91.75 / 91.21 | 89.75 / 89.24 | 89.75 / 88.98 | 84.00 / 82.55 | 90.75 / 90.07 | 91.00 / 90.22 | 89.00 / 88.15 | 73.00 / 71.20 | 88.50 / 87.58 | 75.00 / 72.93 | 86.50 / 85.71 | 85.72 / 84.66 |
| mT5-XL w/ input-level fusion | 90.75 / 90.21 | 92.00 / 91.39 | 91.00 / 90.39 | 92.00 / 91.47 | 92.00 / 91.54 | 91.75 / 90.88 | 90.25 / 89.49 | 90.00 / 88.87 | 91.50 / 90.86 | 90.25 / 89.20 | 92.25 / 91.60 | 91.30 / 90.57 |
| w/ FLARE | 90.75 / 90.21 | 84.25 / 83.80 | 81.50 / 80.55 | 85.00 / 84.06 | 65.75 / 64.70 | 89.75 / 88.32 | 91.50 / 90.50 | 76.50 / 74.36 | 85.00 / 83.64 | 71.00 / 69.29 | 89.00 / 88.00 | 81.92 / 80.72 |

Table 8: Average scores per language in the NusaX dataset. Model performance is evaluated using the Accuracy / Micro F1 metrics.

| Task | Language | ISO Code | Source |
|---|---|---|---|
| XNLI | Arabic | ar | Crowd-sourced (Williams et al., 2018) |
| | Bulgarian | bg | |
| | Chinese | zh | |
| | French | fr | |
| | German | de | |
| | Greek | el | |
| | Hindi | hi | |
| | Russian | ru | |
| | Spanish | es | |
| | Swahili | sw | |
| | Thai | th | |
| | Turkish | tr | |
| | Urdu | ur | |
| | Vietnamese | vi | |
| TyDiQA | Arabic | ar | Wikipedia (Clark et al., 2020) |
| | Bengali | ben | |
| | Finnish | fi | |
| | Indonesian | ind | |
| | Korean | ko | |
| | Russian | ru | |
| | Swahili | sw | |
| | Telugu | tel | |
| NusaX | Acehnese | ace | SmSA (Purwarianti and Crisdayanti, 2019) |
| | Balinese | ban | |
| | Banjarese | bjn | |
| | Buginese | bug | |
| | Indonesian | ind | |
| | Javanese | jav | |
| | Madurese | mad | |
| | Minangkabau | min | |
| | Ngaju | nij | |

Table 9: Overview of languages and corresponding source data used in the experiments, categorized by task.