Exploring The Effectiveness of In-Context Methods in Human-Aligned Large Language Models Across Languages

Ubaidillah Ariq Prathama [1,*], Ayu Purwarianti [2], and Samuel Cahyawijaya [3]

[1, 2] School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung, Indonesia

[3] Cohere, London, United Kingdom

E-mail: 23524058@std.stei.itb.ac.id[1], ayu@staff.stei.itb.ac.id[2], and samuelcahyawijaya@cohere.com[3]

ABSTRACT

Most of past studies about in-context methods like in-context learning (ICL), cross-lingual ICL (X-ICL), and in-context alignment (ICA) come from older, unaligned large language models (LLMs). However, modern human-aligned LLMs are different; they come with chat-style prompt templates, are extensively human-aligned, and cover many more languages. We re-examined these in-context techniques using two recent, human-aligned multilingual LLMs. Our study covered 20 languages from seven different language families, representing high, mid, and low-resource levels. We tested how well these methods generalized using two tasks: topic classification (SIB-200) and machine reading comprehension (Belebele). We found that utilizing prompt templates significantly improves the performance of both ICL and X-ICL. Furthermore, ICA proves particularly effective for mid- and low-resource languages, boosting their F1-score by up to 6.1%. For X-ICL, choosing a source language that is linguistically similar to the target language, rather than defaulting to English, can lead to substantial gains, with improvements reaching up to 21.98%. Semantically similar ICL examples continue to be highly relevant for human-aligned LLMs, providing up to a 31.42% advantage over static examples. However, this gain decreases when using machine translation model to translate query from target language. These results collectively suggest that while modern human-aligned LLMs definitely benefit from in-context information, the extent of these gains is highly dependent on careful prompt design, the language's resource level, language pairing, and the overall complexity of the task.

Keywords: Large Language Models (LLMs), In-Context Learning (ICL), Cross-Lingual ICL (X-ICL), In-Context Alignment (ICA)

1. Introduction

Large Language Models (LLMs) have recently shown impressive ability to perform well across many different tasks, topics, and languages [1], [2], [3], [4], [5], [6]. However, they don't perform as well in situations with limited resources, like low-resource languages or specialized topics. This shows there's a big gap in how well LLMs handle these low-resource scenarios. To fix this, it's crucial to adapt and align LLMs so they can understand and create text in a wider range of languages and topics [7], [8], [9], [10], [11], [12], [13]. For instance, studies have shown that aligning multilingual representations using parallel data can boost multilingual abilities without needing specific task fine-tuning [14], [15], [16], [17]. However, many of these methods are hard to implement on a large scale because they require adjusting parameters with vast amounts of data. Another approach focuses on making LLMs generalize efficiently in test-time in low-resource settings by providing information within the context. This has led to the development of various in-context methods such as in-context learning (ICL) [18], [19], [20], cross-lingual ICL (X-ICL) [21], [22], [23], [24], and in-context alignment (ICA) [12], [25] which effectively improve how well LLMs generalize.

Due to the fast pace at which LLMs are developing, in-context methods haven't yet been thoroughly tested with the newest, human-aligned LLMs [1], [2], [3], [4], [5], [6]. These newer models are different from older LLMs [18],

* Corresponding author.

Received: June 23rd, 2025. Revised: July 3rd, 2025. Accepted: July 8th, 2025.

Available online: July 8th, 2025.

DOI: https://doi.org/10.12962/j24068535.v23i1.a1323

[26], [27] because they undergo more rigorous post-training [28], [29], [30], [31], [32], [33], [34], [35]. This extra training gives them advanced capabilities beyond just generating language, such as reasoning, acting as agents, and aligning with human values. This also brings in new in-context features like chat template; system prompt; and various special tokens [36], [37], [38], [39]. Given these significant differences from previous LLMs, it's unclear whether the past findings about in-context methods still apply to the latest human-aligned LLMs.

To really understand in-context methods in the latest human-aligned LLMs, we're looking at what makes them different from older models. First, modern LLMs use a specific chat template that includes sections for the system, user, and assistant. How these separate parts of the prompt affect in-context methods haven't been studied before. So, we're exploring how to adapt in-context methods to work with these new components. Second, the newest LLMs support many more languages and are better at generalizing to low-resource languages. Our research will specifically examine various resource levels across diverse regions, covering a total of 20 languages. Third, since current LLMs are great at generalizing across different languages and tasks, we're re-evaluating how effective techniques for retrieving semantically similar in-context learning (ICL) examples are. In short, our contributions cover:

1. **Enhancing Human-Aligned LLMs with In-Context Methods:** The study shows that applying in-context methods to the latest human-aligned LLMs significantly boosts their performance, achieving up to a 55% improvement over zero-shot, particularly for topic classification tasks.
2. **Impact of In-Context Information Placement:** The study explored how putting in-context information in different parts of a prompt affects outcomes. It turns out that the placement has only a small impact on LLM performance. While it improves In-Context Learning (ICL) and Cross-Lingual ICL (X-ICL), it actually decreases zero-shot performance compared to traditional prompt.
3. **Influence of Language Resource on In-Context Methods:** The study also looked into how choosing specific languages affects in-context methods. In-Context Alignment (ICA) led to an up to 6.1% performance increase for low- and mid-resource languages. Additionally, picking a source language similar to the target language significantly improved X-ICL performance by up to 21.98%.
4. **Effectiveness of ICL Example Retrieval Techniques:** The study compares various techniques for retrieving ICL examples, semantically similar examples remained highly effective in human-aligned LLMs, providing up to a 31.42% gain over static examples. However, this effectiveness decreased when a machine translation model was used to translate query from target language.

2. Literature Review

Historically, LLMs like GPT-3 [18] with its 175 billion parameters were built using extensive unsupervised pre-training on vast amounts of text. Later, models such as BLOOM [26] and XGLM [27] expanded on this by incorporating diverse multilingual data to generalize across different languages. In contrast, human-aligned LLMs represent a significant evolution. These models now integrate sophisticated post-training techniques like instruction-tuning [15], [28], [30], preference-tuning [9], [32], [35], and reinforcement learning from human feedback (RLHF) [40], [41], [42]. Many recent open-source LLMs leverage these methods. For example, Meta's Llama 3 [5] uses instruction-tuning on over 10 million annotated examples, while Google's Gemma 3 [6] employs human preference-tuning. This human-centric approach also extends to multilingual models. Qwen 2.5 [43] is trained with specific human preference alignment in various languages, and Aya Expanse [44] breaks new ground by using multilingual RLHF. This is especially notable because it tackles the challenge of a lack of high-quality non-English preference data, pushing LLM alignment beyond models primarily focused on English.

Recent work on steering large language models increasingly exploits the three-role chat template to inject structured reasoning signals. Constitutional AI frames an explicit list of behavioral principles as the system message, lets the model self-critique under that constitution in hidden assistant turns, then produces a revised assistant reply to the original user prompt, thereby hard-coding alignment goals into the dialogue flow [36]. ReAct casts the environment's observations as incremental user messages and interleaves "Thought" and "Action" strings inside the single assistant role, so the agent can reason, act via API calls, and reflect in one continuous chat trace [37]. Planning Tokens fine-tunes models to emit a special planning symbol at the start of every reasoning step, effectively inserting a self-generated mini-system instruction that guides each subsequent assistant sentence without changing the external user input [38]. Pause Tokens prepend a learned sequence of <pause> markers between the user prefix and the first assistant word, giving the network extra forward passes before it speaks and preserving the same delayed template during pre-training, fine-tuning, and inference [39].

ICL first introduced by [18], allows LLMs to execute tasks using a small number of examples and instructions directly within the input, foregoing the need for model weight adjustments. In monolingual applications, ICL's effectiveness has been amplified by methods like chain-of-thought prompting, which encourages step-by-step reasoning [45] and retrieval techniques that select semantically similar examples [46]. The use of sentence encoder for semantic textual similarity improves LLMs performance in sentiment analysis, table-to-text generation, and question-answering compared to random examples selection. The application of ICL to multiple languages, known as X-ICL, was pioneered by [21], demonstrating that LLMs can achieve strong results in multilingual tasks with minimal examples. X-ICL mimics ICL performance, but with advantage in low-resource languages as ICL examples is not always available in low-resource languages. Echoing findings from monolingual ICL, subsequent research has shown that retrieving semantically similar examples also enhances cross-lingual performance [12].

Recent advancements in ICL have spurred the development of in-context alignment (ICA) strategies. For instance, X-InSTA [25] markedly improved performance over random prompt choices across 44 cross-lingual pairs by ensuring semantic consistency in examples and aligning labels between languages. They formulate ICA by combining the well-known semantic similar X-ICL examples and adding as simple translation of every possible label for each task between source and target language. This method helps LLMs to reason better across languages with up to 23% f1-score improvement in MARC, CLS, and HatEval. Building on this, [12] extended the research to 25 languages with fewer resources, finding that aligning queries could be more effective than aligning labels and boosting performance for these languages in zero-shot scenarios. They modified ICA using sentences semantically similar to query from parallel corpora. Instead of using source-target language pair of label task, they use these sentences in the prompt. This method shows improvement compared to X-InSTA, as previous method may make the model classify in shifted label space due to the prompt design. However, a significant limitation of this body of work is that its evaluation has been confined to LLMs not specifically fine-tuned with human feedback. This study aims to address this gap by evaluating the efficacy of these methods on more recent, human-aligned LLMs.

3. Methodology

3.1. Overview

Previous research has consistently demonstrated that ICL and X-ICL can significantly boost model performance across various tasks [18], [21]. Building on this, [25] introduced in-context label alignment to further enhance ICL's effectiveness, a concept later refined by [12] with the addition of in-context query alignment. Our current work extends the foundational methods proposed in [12].

Our prompt construction relies on four core components: the task instruction I^{task}, the user's query in target language q^{tgt}, an alignment example A, and ICL examples ICL. To create the alignment example, we leverage a parallel exemplar dataset $D^{para} = \left\{ \left(s_1^{src}, s_1^{tgt} \right), \cdots, \left(s_m^{src}, s_m^{tgt} \right) \right\}$, where each pair consists of a source and its corresponding target sentences. From this dataset, we select the top-k parallel sentence pairs by maximizing the monolingual similarity between the user's query q^{tgt} and sentence in target language s_i^{tgt}. For a given target language L^{tgt}, these chosen pairs are then formatted into an alignment prompt structured as, "In L^{tgt}, s_1^{src} means s_1^{tgt}, $\cdots$, and s_k^{src}means s_k^{tgt}". Similarly, we retrieve ICL examples from the specific task dataset $D^{task} = \{(e_1, y_1), \cdots, (e_n, y_n)\}$, where e_i and y_i denote the input and label example. We select e_i semantically similar to q^{tgt}. For monolingual ICL, we use monolingual similarity, while for X-ICL, we employ multilingual similarity. We can summarize this into:

Table 1: Chat-template for in-context methods

Traditional Prompt	**Chat-Template**
I^{task}	**User:** $I^{task} \oplus e_i$
$e_1 \rightarrow y_1$	**Assistant:** y_1
$\vdots$	$\vdots$
$e_k \rightarrow y_k$	**User:** $I^{task} \oplus e_k$
$q^{tgt} \rightarrow y_{pred}$	**Assistant:** y_k
	User: $I^{task} \oplus q^{tgt}$
	Assistant: y_{pred}

$$A = f(q^{tgt}, D^{para}, k) \tag{1}$$

$$ICL = f(q^{tgt}, D^{task}, k) \tag{2}$$

We create both alignment prompts shown in (1) and ICL examples shown in (2) for every query. The quantity of examples to retrieve is determined by k, and a semantic textual similarity model is used for this retrieval. The alignment is formatted as a single string, while the ICL consists of a list containing k pairs of inputs and their corresponding labels.

3.2. Prompt Construction

Recent LLMs are often designed with specific chat templates that delineate roles for the system, user, and assistant. These templates are crucial for enhancing the LLMs' ability to learn from provided examples and generate outputs that align more closely with human expectations [47]. Our research investigated the impact of five different formatting strategies on performance. We tested these against three variations of chat-template prompts and two variations of traditional prompts serving as baselines. The variables definition refers to Section 3.1.

Traditional (Tr). We used the traditional prompt format directly, as shown in Table 1, without any modifications. The alignment prompt was placed at the very beginning. The model was then expected to predict its response y_{pred} directly from the designated label space.

Traditional with Chat-Template (TC). This strategy involved modifying the traditional prompt. The alignment information was still added at the beginning, but the entire prompt was then formatted to fit within the user's chat-template structure. The model's prediction y_{pred} was expected to appear as a distinct part of the assistant's response.

First Turn Chat-Template (FT). We adopted the standard chat-template format from Table 1. The alignment information was appended directly to the beginning of the first user turn in the conversation sequence. Equation (3) shows the modification of first user turn.

$$\boldsymbol{User}\text{:}\;\; A \oplus I^{task} \oplus e_i \tag{3}$$

Last Turn Chat-Template (LT). Similar to FT, this strategy also used the chat-template format from Table 1. However, the alignment information was appended to the beginning of the last user turn. Equation (4) shows the modification of last user turn.

$$\boldsymbol{User}\text{:}\;\; A \oplus I^{task} \oplus q^{tgt} \tag{4}$$

Separate Turn Chat-Template (ST). For this method, we introduced an additional user and assistant turn at the very beginning of the chat-template format (from Table 1). In this initial user turn, we provided the alignment content along with an explicit instruction: "Use this information to answer the task below," guiding the model to leverage the provided context for its response. Equation (5) shows the modification of first user and asisstant turn.

$$\begin{aligned} \boldsymbol{User}\text{:}&\;\; A \oplus I^{align} \\ \boldsymbol{Assistant}\text{:}&\;\; Ok \end{aligned} \tag{5}$$

Table 2: Languages detail used in experiment

Lang Code	Language	Script	Lang Family	Region	Joshi's Class	CC Crawl %	Res Level
rus_Cyrl	Russian	Cyrillic	Indo-European	Europe 2	4	5.9294	High
zho_Hans	Chinese Simplified	Han	Sino-Tibetan	Asia 3	5	5.4135	High
deu_Latn	German	Latin	Indo-European	Europe 1	5	5.1154	High
jpn_Jpan	Japanese	Japanese	Japonic	Asia 3	5	4.8624	High
spa_Latn	Spanish	Latin	Indo-European	Europe 1	5	4.4426	High
fra_Latn	French	Latin	Indo-European	Europe 1	5	4.2269	High
ind_Latn	Indonesian	Latin	Austronesian	Asia 3	3	1.2009	Medium
kor_Hang	Korean	Hangul	Koreanic	Asia 3	4	0.8028	Medium
ukr_Cyrl	Ukrainian	Cyrillic	Indo-European	Europe 2	4	0.6240	Medium
hin_Deva	Hindi	Devanagari	Indo-European	Asia 2	4	0.2027	Medium
ben_Beng	Bengali	Bengali	Indo-European	Asia 2	3	0.1107	Medium
arb_Arab	Arabic	Arabic	Afro-Asiatic	Asia 1	5	0.0774	High
mkd_Cyrl	Macedonian	Cyrillic	Indo-European	Europe 2	1	0.0387	Low
urd_Arab	Urdu	Arabic	Indo-European	Asia 2	3	0.0300	Medium
jav_Latn	Javanese	Latin	Austronesian	Asia 3	1	0.0023	Low
snd_Arab	Sindhi	Arabic	Indo-European	Asia 2	1	0.0016	Low
sun_Latn	Sundanese	Latin	Austronesian	Asia 3	1	0.0012	Low
yor_Latn	Yoruba	Latin	Atlantic-Congo	Africa	2	0.0008	Low
ibo_Latn	Igbo	Latin	Atlantic-Congo	Africa	1	0.0006	Low
fuv_Latn	Nigerian Fulfulde	Latin	Atlantic-Congo	Africa	0	-	Low

It's important to note the variations in the number of prompt types across different settings. In a zero-shot setting without alignment, all chat-template prompts behaved identically since their differences primarily involve ICL and alignment placement, resulting in only two distinct variations (traditional vs. chat-template types). However, in zero-shot with alignment, the "Separate Turn Chat-Template" became distinct due to its unique turn structure, leading to three variations. Finally, in ICL and X-ICL settings without alignment, all chat-template prompts effectively became the same because their distinguishing feature (alignment placement) was not active, leaving us with three distinct variations.

3.3. Language Categorization

We've established our language resource levels by considering two key metrics: Joshi's Class and the CC Crawl percentage (from CC-MAIN-2025-13). Joshi's Class, as outlined in [48], indicates a language's relative priority or resource availability, with higher values signifying greater priority. The CC Crawl percentage, on the other hand, quantifies how often and extensive content in a given language has been indexed online. Based on these, we classify languages as high-resource if they have a Joshi's Class of 5 or 4 and a CC Crawl percentage exceeding 1%. Languages are deemed low-resource if their Joshi's Class is 0 or 1 *or* their CC Crawl percentage falls below 0.001%. All other languages are categorized as mid-resource. This dual-threshold approach ensures our high-, mid-, and low-resource classifications accurately reflect languages that are not only theoretically well-studied but also practically accessible at an internet scale, making our analysis more representative of real-world data availability.

3.4. Experimental Setup

A. Language Selection

For our experiments, we carefully selected a diverse set of 20 languages, which you can find detailed in Table 2. This language set is remarkably broad, encompassing nine distinct writing systems, including nine Latin-script languages, three Arabic, three Cyrillic, and one each for Devanagari, Bengali, Hangul, Japanese, and Han.

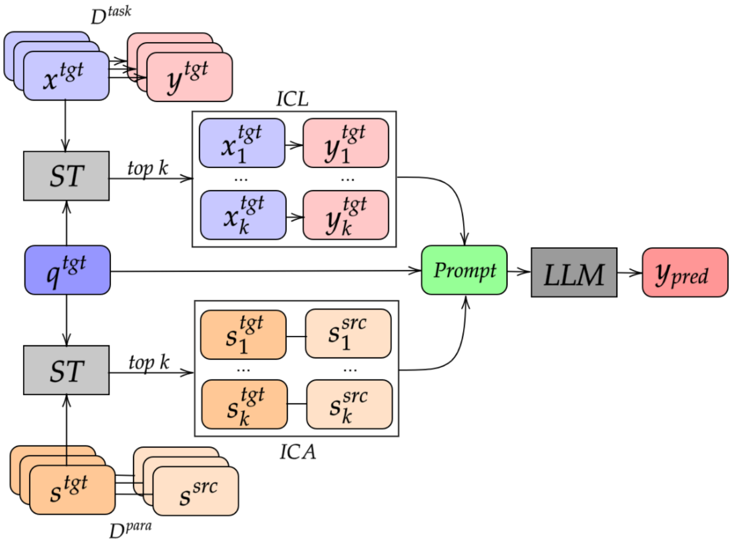

Fig. 1: Diagram for in-context learning with in-context alignment (ICL-ICA) methods.

These languages also belong to seven different families: Indo-European is the most represented with ten languages, followed by Austronesian and Atlantic-Congo (each with three), and one each from Afro-Asiatic, Sino-Tibetan, Japonic, and Koreanic. Geographically, our selection spans all major global regions except for America and Oceania, including three from Europe 1, three from Europe 2, one from Asia 1, four from Asia 2, six from Asia 3, and three from Africa. This wide regional distribution ensures our sample isn't skewed towards any single continent. In terms of resource quality, our custom metric results in a balanced mix: seven high-resource, six medium-resource, and seven low-resource languages. This means our dataset effectively combines languages that are both well-documented and widely available online with those that are either scarcely documented or barely appear in web-scale crawls.

B. Datasets

Our evaluation focused on two distinct downstream tasks: topic classification using the SIB-200 dataset [49] and machine reading comprehension with the Belebele dataset [50]. For SIB-200, we evaluated performance on its test set, while its training and development sets served as sources for In-Context Learning (ICL) examples. With Belebele, we sampled 200 questions for our evaluation and used the remainder for ICL example retrieval. Additionally, we leveraged the FLORES-200 parallel corpora [51] to generate our alignment examples, specifically utilizing the 20 languages detailed earlier.

C. Models

We selected Aya Expanse (8B) [44], which supports 23 languages, and Qwen2.5 Instruct (7B) [43], supporting 29 languages, as our primary Large Language Models (LLMs). These models were chosen for their robust multilingual capabilities despite having a relatively small number of parameters. All models were run in 16-bit precision, and classification decisions were made based on the highest logit probability for each potential label. We use stsb-xlm-r [52] for retrieving examples for both ICL and ICA. The number of retrieved examples (k) was set to 7 for ICL on SIB-200, 5 for ICL on Belebele, and 3 for all in-context alignment scenarios. For translation tasks within the translate-test zero-shot and translate-test ICL settings, we employed nllb-200-distilled-1.3B [51].

The LLMs are loaded using *AutoTokenizer* and *AutoModelForCausalLM* from transformers library. All variables set to default, except using *torch_dtype = torch.float16* to load the model in 16-bit precision. We apply chat-template to the prompt using built-in *apply_chat_template* function from AutoTokenizer. We use *max_length = 8192* and *truncation = True* for the tokenizer. Finally, the logits probability will be calculated from the output of *model(**inputs, labels=input_ids)*. We don’t use *model.generate()* because we need the raw output for calculating the probability. We use *sentence_transformers* library for the semantic textual similarity model. We

Table 3: Experiment results of Aya Expanse (8B) on Belebele dataset

method	avg	rus	zho	deu	jpn	spa	fra	arb	ind	kor	ukr	hin	ben	urd	mkd	jav	snd	sun	yor	ibo	fuv
TT-ZS	61.875	72.5	61	76.5	66.5	73.5	79	67	73	66.5	70.5	61	60	57.5	61	56	63.5	55	42	47	28.5
TT-ICL	**66.975**	86	63.5	84.5	61.5	76	81	66	80	74	72.5	71.5	**60.5**	**70**	**73.5**	56	**67**	**56.5**	**57.5**	**48**	**34**
ZS	57.85	80	85	73.5	**84.5**	66	62.5	69.5	65	77.5	75	64	50	49.5	55	46.5	37.5	36	25.5	27	27.5
ZS-ICA	45.225	45.5	80	57.5	72	54	44	45.5	55	58.5	50	41.5	40	31	42	43	31.5	33	28	28	24.5
ICL	64.65	86.5	87	84	81.5	**86**	84	81	**80.5**	81.5	82	75.5	49	56	59.5	48.5	40.5	44.5	27	27	31.5
ICL-ICA	66.625	89	86	85	82.5	84	**85.5**	**81.5**	79.5	82	82.5	**77.5**	53	59.5	69	54	46	48	33.5	27	27.5
X-ICL	65.525	**91.5**	**87.5**	**88.5**	81	84	83.5	80.5	80	**83.5**	81	75	52	64	68	45.5	39	38	33	27	28
X-ICL-ICA	66.275	88.5	84.5	**88.5**	81.5	82.5	83	81	**80.5**	81.5	**83.5**	**77.5**	53.5	62	69.5	**58**	38.5	45	33.5	29.5	23.5

Table 4: Experiment results of Qwen2.5 Instruct (7B) on Belebele dataset

method	avg	rus	zho	deu	jpn	spa	fra	arb	ind	kor	ukr	hin	ben	urd	mkd	jav	snd	sun	yor	ibo	fuv
TT-ZS	64.75	75.5	65	80.5	74	77	82.5	68	79.5	62	74.5	67.5	**70**	67	67.5	**57.5**	64	**58.5**	37.5	48.5	18.5
TT-ICL	68.15	80	68.5	87	76.5	78	86	69.5	82.5	73.5	76.5	73	67	71.5	68.5	56.5	**66**	58	**47.5**	**51**	26
ZS	65.375	86	89	84.5	83.5	86	87.5	81.5	82	85	79	65.5	64.5	59.5	73	48	40	42	28	20.5	22.5
ZS-ICA	65.625	88.5	85	86	80	84.5	83	83	74.5	83.5	79.5	70	64.5	64	74	48.5	44	41.5	29	25	24.5
ICL	67.8	88.5	**91.5**	88	85.5	**90**	**89.5**	83	84.5	85.5	81.5	63.5	65	65	75	53.5	41	49.5	25	25	26
ICL-ICA	68.475	**91.5**	89	87	**87**	87	87.5	**85.5**	**86**	86.5	84.5	69	69	69.5	74.5	52.5	46	47.5	23.5	19.5	**27**
X-ICL	68.8	90.5	89.5	88.5	84.5	**90**	88	83.5	85	**88**	81.5	70.5	65.5	68	76	52	45	49	31	23.5	26.5
X-ICL-ICA	**68.825**	89.5	87.5	**91**	84	88	88	81.5	84	**88**	**85**	**73.5**	67	**72**	**77.5**	51	45	47.5	30.5	19	**27**

convert the ICL examples into embedding space using built-in *encode* function from *sentence_transformers* with *batch_size=128* and other variables set to default. We search top-k similar sentence by converting the query into embedding space using the same method, then search for the highest dot product. We use *AutoTokenizer* and *AutoModelForSeq2SeqLM* for the machine translation model. Both of these variables then used as input for transformers *translation pipeline* and the *max_length* is set to 600, as the query is not long.

D. In-Context Methods

Our study compared eight distinct methods for classification using LLMs. We established translate-test zero-shot (TT-ZS) and translate-test in-context learning (TT-ICL) as baselines for scenarios where MT model is available. Beyond these, we explored common strategies including zero-shot (ZS), in-context learning (ICL), and cross-lingual ICL (X-ICL). Furthermore, we investigated the impact of in-context alignment (ICA) by providing semantically similar sentence translations alongside these primary methods (ZS-ICA, ICL-ICA, X-ICL-ICA).

There are 3 models used in our methods, which is Sentence Transformer (ST) for calculating the semantic textual similarity, Machine Translation to translate query from target language to source language, and Large Language Model (LLM) to do classification on downstream task. Fig. 1 shows ICL-ICA methods which is one of the methods used in our experiment. This method chooses ICL example semantically similar to q^{tgt} from D^{task} and ICA sentence semantically similar to q^{tgt} from D^{para}. These examples will be used to generate prompt for LLM to run inference. In zero-shot settings, we remove the ICL component. In ZS, ICL, X-ICL settings, we remove the ICA component. In X-ICL settings we change the ICL example from $\left(x^{tgt}, y^{tgt}\right)$ to $\left(x^{src}, y^{src}\right)$. In TT-ZS and TT-ICL settings, we add machine translation component to translate q^{tgt} before the whole process.

4. Results and Discussion

In this section, we analyze the experimental results for the cross-lingual tasks on the SIB-200 and Belebele datasets. The initial experiments on Belebele are presented in Tables 3 and 4, while those on SIB-200 appear in Tables 5 and 6. We ran experiments on 20 selected languages spanning various resource levels and language families,

Table 5: Experiment results of Aya Expanse (8B) on SIB-200 dataset

method	avg	rus	zho	deu	jpn	spa	fra	arb	ind	kor	ukr	hin	ben	urd	mkd	jav	snd	sun	yor	ibo	fuv
TT-ZS	75.02	75.02	78.9	78.3	74.22	78.6	76.16	77.58	79.78	73.49	75.01	80.72	76.53	79.1	75.62	72.83	76.76	74.28	66.48	73.87	57.12
TT-ICL	**87.14**	**90.3**	85.79	89.62	86.58	89.09	90.13	**89.82**	89.64	**88.74**	88.73	**89.91**	**88.68**	**85.38**	**87.74**	**86.25**	**89.15**	**86.16**	**81.62**	**87.55**	**71.99**
ZS	37.66	35.92	41.62	52.04	50.39	45.48	54.23	33.22	33.16	32.33	23.98	42.32	31.05	25.07	36.65	39.71	35.88	41.70	32.99	29.52	36.01
ZS-ICA	8.10	6.04	7.43	10.31	8.73	18.21	10.43	0.61	1.09	3.58	4.56	2.59	15.41	6.12	7.33	6.08	6.90	7.90	12.87	14.94	10.79
ICL	80.11	86	**89**	**90.3**	**89.4**	**89.96**	**90.18**	88.12	**90.18**	88.63	87.78	**89.91**	79.33	82.11	85.2	78.83	72.30	80.03	43.91	45.54	55.55
ICL-ICA	77.62	82.28	87.49	85.84	87.37	86.28	85.64	83.23	87.15	82.1	85.18	88.81	78.27	79.67	82.11	79.67	72.81	79.09	43.44	43.65	52.36
X-ICL	78.53	85.56	87.35	86.25	86.74	89.8	89.32	86.38	87.67	87.62	**89.21**	87	78.13	83.12	83.12	75.54	76.25	79.97	39.11	44.92	51.62
X-ICL-ICA	76.25	83.67	85.98	83.58	86.12	84.1	83.02	83.17	86.19	84.13	87.49	85.02	76.73	79.69	79.69	74.88	68.28	80.12	39.17	44.47	49.44

Table 6: Experiment results of Qwen2.5 Instruct (7B) on SIB-200 dataset

method	avg	rus	zho	deu	jpn	spa	fra	arb	ind	kor	ukr	hin	ben	urd	mkd	jav	snd	sun	yor	ibo	fuv
TT-ZS	73.56	74.41	75.8	76.32	72.43	74.61	75.87	76.74	74.06	73.29	75.22	77.56	73.77	75.25	76.01	74.88	75.39	74.27	64.29	74.55	56.57
TT-ICL	**86.96**	87.21	88.45	88.11	86.71	89.39	86.5	88.04	**89.03**	88.94	**91.18**	**89.02**	**89.41**	**90.41**	85.81	**89.66**	**89.38**	**88.51**	**81.22**	**85.03**	**67.11**
ZS	60.86	76.8	73.09	72.96	68.92	73.17	72.94	70.41	77.92	71.48	71.59	69.66	65.93	58.80	73.05	59.06	44.38	52.03	24.51	17.93	22.55
ZS-ICA	62.43	68.81	71.34	75.07	67.50	70.80	68.03	61.23	63.85	69.11	75.22	72.61	73.28	75.55	70.60	58.34	58.03	59.91	31.20	30.46	27.56
ICL	80.92	**90.32**	90.16	**91.45**	**89.75**	**90.22**	**87.42**	**89.30**	86.87	89.34	90.63	86.96	86.39	84.38	86.94	83.43	77.23	81.59	43.24	44.30	48.50
ICL-ICA	78.56	88.44	87.20	87.79	86.06	85.85	85.31	89.17	86.92	85.61	86.40	81.57	84.34	82.46	83.06	75.85	75.23	79.15	42.14	42.07	52.57
X-ICL	79.04	88.60	**90.76**	88.37	85.03	89.55	86.06	89.11	86.93	**89.76**	86.90	83.98	85.79	84.29	**88.96**	75.14	70.70	78.23	41.04	45.41	46.19
X-ICL-ICA	77.08	87.74	85.24	86.55	84.20	88.47	82.55	86.23	84.56	86.56	85.18	86.37	80.93	81.56	82.31	72.76	70.50	76.98	40.33	41.52	50.96

using the LT prompt format. The zero-shot and in-context learning methods with a translation model served as our baselines. In addition, we evaluated six other approaches zero-shot (ZS), in-context learning (ICL), and cross-lingual in-context learning (X-ICL), each with an added in-context alignment (ICA) step. Bolded figures in each row denote the best result for that language, and the second-best are underlined.

Overall, it is clear that high-resource languages such as Spanish, French, German, Arabic, Japanese, and Mandarin consistently achieve superior performance compared to their low-resource counterparts. On average, these well-represented languages yield f1-scores above 80% with relatively little fluctuation from one dataset to another. In contrast, truly low-resource languages like Yoruba, Igbo, and Nigerian Fulfulde struggle to exceed the 40–50% range, especially on the Belebele benchmark, which demands deep text understanding and nuanced reasoning to answer open-ended questions. This stark disparity highlights that both LLMs remain heavily dependent on the quantity and quality of available pretraining data. When a language is under-represented in the training corpus, model comprehension and answer accuracy drop significantly.

From a methodological standpoint, both Aya and Qwen exhibit similar trends. The translate-test zero-shot (TT-ZS) baseline delivers a stable performance across languages, generally outperforming plain zero-shot (ZS) and even pure in-context learning (ICL) in some cases. Adding a handful of translated, in-context examples (TT-ICL) further elevates accuracy, making TT-ICL the top or runner-up method for most languages on both benchmarks. Pure ZS remains the weakest approach, especially for low-resource languages. ICL and X-ICL can close the gap, but they too tend to fall short of TT-ICL's gains underscoring the power of machine translation model.

However, the absence of robust translation models for extremely low-resource languages means that TT-based methods are not always viable. In these cases, approaches that do not rely on machine translation such as ICL and X-ICL using related languages become essential. Early results suggest these alternative strategies can still provide meaningful improvements over zero-shot, although they rarely match the heights of TT-ICL on high-resource languages. Moving forward, a deeper investigation into translation-free methods, alongside targeted data augmentation for low-resource languages, will be critical to closing the performance gap and ensuring more equitable comprehension capabilities across languages.

Both models exhibit distinct sensitivities to the various prompting and alignment methods. For Aya, zero-shot approach delivers notably lower performance than any other methods, often trailing by 10–20 points compared to its own few-shot or translation-based baselines. When you add in-context alignment (ZS-ICA) on top of ZS, Aya's accuracy actually drops further, suggesting that alignment worsens its already low zero-shot outputs. Qwen, by contrast, shows a milder performance gap under ZS, although it still lags behind ICL and TT-ICL results, the drop is only around 5–10 points. Moreover, Qwen can utilize ICA effectively. ZS-ICA boosts its zero-shot scores by 3–5 points. This divergence implies that the two models have learned different internal representations of text. Aya's representations struggle to benefit from the loose guidance of alignment when no examples are provided, whereas Qwen's are sufficiently robust to utilize alignment step even in the absence of demonstrations.

Overall, these findings reaffirm that the translate-test baseline remains the most stable and generally effective approach for cross-lingual classification across languages of varying resource levels. In both Aya and Qwen, TT-ICL provide the most consistent uplift, making them the best choice for scenarios where translation models exist. However, because many low-resource languages lack reliable translation systems, it is crucial to continue exploring alternative methods such as ICL, X-ICL, and their aligned variants to ensure reasonable performance without translation. Each of these alternatives can offer efficient solutions with surprisingly strong results, especially when tuned to the specific strengths of the model and the characteristics of the dataset. In practice, model developers should weigh the availability of translation, the alignment behavior of their chosen LLMs, and the nature of the task before selecting a final classification strategy.

4.1. Chat Template Boosts In-Context Methods Performance

Our comprehensive investigation delved into the influence of prompt formatting across eight distinct in-context methods, two LLMs, and two diverse datasets. To ensure a broad and representative analysis, we carefully selected five languages German (deu), Indonesian (ind), Hindi (hin), Sundanese (sun), and Nigerian Fulfulde (fuv) each possessing varying resource levels and originating from different geographical regions, as detailed in Table 7 and 8. We meticulously categorized the prompts utilized in our experiments into three main types: template-based prompts (including FT, LT, and ST formats), transition prompts (referred to as TC), and traditional prompts (referred to as Tr). Our observations, graphically represented in Fig. 2, reveal that transition prompts exhibited the highest degree of instability, proving particularly detrimental to performance when applied to the SIB-200 dataset with the Aya model. This finding underscore how even subtle alterations in prompt structure can lead to significant performance degradation, likely because such less conventional formats are largely absent from the vast instruction-tuning datasets that shape these models, unlike more common prompt structures. In contrast, prompts constructed using consistent templates generally demonstrated a slight superiority over traditional prompts across both ICL and TT-ICL scenarios. This suggests that the inherent consistency in template-based formatting aids LLMs in achieving more effective task execution. However, results for X-ICL were notably more variable, perhaps indicating that X-ICL specific instruction-tuning is less pervasive, leading to only marginal and inconsistent gains. While the performance disparities between template-based and traditional prompts were minor in ZS and TT-ZS settings, template-based prompts did slightly outperform traditional ones specifically on the Belebele dataset. Conversely, traditional prompts delivered substantially better results on the SIB-200 dataset, demonstrating that prompt efficacy is significantly influenced by the characteristics of the specific task.

Our analysis of ICA within the framework of template-based prompts revealed largely consistent performance across all three tested prompts, with a marginally higher score observed for the FT prompt on the SIB-200 dataset. Interestingly, variations in the placement of the alignment content within the prompt yielded only negligible improvements in overall performance. However, a more complex picture emerged when examining ZS-ICA. The Aya model experienced a pronounced decrease in performance when utilizing template-based prompts in a ZS-ICA setting. Conversely, the Qwen model exhibited a similarly significant reduction in ZS-ICA performance but specifically when employing traditional prompts. This divergent behavior across models suggests that a higher

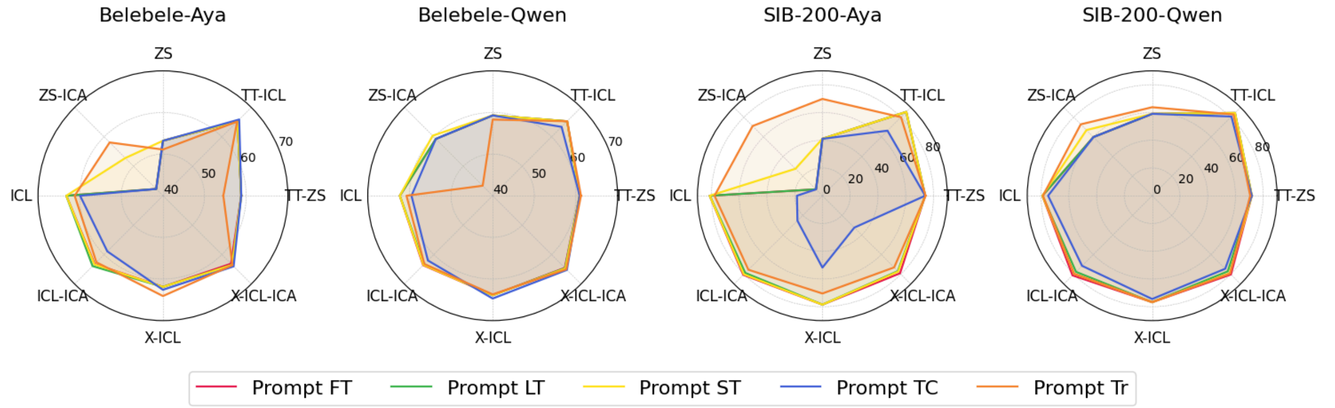

Fig. 2: Performance comparisons of 5 prompt formattings aggregated for 5 sample languages on (1) Belebele with Aya, (2) Belebele with Qwen, (3) SIB-200 with Aya, (4) SIB-200 with Qwen.

Table 7: Experiment results of Belebele for five languages on two models (Aya Expanse (8B) and Qwen2.5 Instruct (7B)) under different methods and prompt-type configurations.

Method	**Prompt**	**Aya Expanse (8B)**					**Qwen2.5 (7B)**				
		deu	**ind**	**hin**	**sun**	**fuv**	**deu**	**ind**	**hin**	**sun**	**fuv**
TT-ZS	FT,LT,ST,TC	**76.5**	**73**	**61**	55	**28.5**	80.5	**79.5**	67.5	**58.5**	18.5
	Tr	68.5	63.5	59	**55.5**	26	80.5	79	**68**	56.5	**22**
TT-ICL	FT,LT,ST	84.5	80	71.5	56.5	**34**	87	**82.5**	**73**	**58**	26
	TC	85	**82**	**73**	**59.5**	30	86.5	81.5	71	56	22
	Tr	**86**	79	72	**59.5**	29.5	**88.5**	**82.5**	71	57.5	**27**
ZS	FT,LT,ST,TC	**73.5**	65	**64**	**36**	**27.5**	84.5	82	**65.5**	**42**	**22.5**
	Tr	73	**67**	62	31.5	22	**85.5**	**83**	64	36.5	**22.5**
ZS-ICA	FT,LT,TC	57.5	55	41.5	33	24.5	86	74.5	**70**	41.5	24.5
	ST	83	53.5	62	36	**29.5**	**87.5**	**77.5**	68.5	**42**	**26.5**
	Tr	**87**	**72**	**66.5**	**36.5**	28.5	61.5	49.5	51.5	30.5	24
ICL	FT,LT,ST	**84**	**80.5**	75.5	**44.5**	**31.5**	**88**	**84.5**	63.5	**49.5**	26
	TC	81.5	76	74.5	41.5	26	86	81.5	**64.5**	40.5	25
	Tr	83	79	**76**	43	25	87	81.5	63.5	44	**27.5**
ICL-ICA	FT	85	**84**	75.5	45	26.5	87.5	**87**	67	**48.5**	28
	LT	**87**	79.5	**77.5**	**48**	**27.5**	87	86	**69**	47.5	27
	ST	86	82.5	75.5	45.5	27	88.5	**87**	66	47.5	28.5
	TC	80.5	73	75.5	41	24.5	88	83.5	65	44	**29.5**
	Tr	86.5	82	74	44.5	26	**89**	85	66.5	47	28.5
X-ICL	FT,LT,ST	88.5	80	75	38	28	88.5	**85**	70.5	**49**	26
	TC	**92.5**	83.5	74	35.5	27.5	**90.5**	83.5	**73**	43.5	**33**
	Tr	90	**84**	**77.5**	**40.5**	**28.5**	90.5	84	70	44	30
X-ICL-ICA	FT	88.5	80	76	41.5	**29.5**	90	**87**	70.5	**47.5**	27.5
	LT	91	80.5	77.5	**45**	23.5	91	84	**73.5**	**47.5**	27
	ST	88	80	74.5	42	33	90.5	**87**	70	46.5	29.5
	TC	**91.5**	82	**78**	41	27.5	91	82	72	44.5	**36.5**
	Tr	87	**82.5**	76.5	43	29	**92.5**	85.5	72.5	46.5	28

proportion of alignment content relative to the core instruction and query might, in certain contexts, confuse the model, leading to suboptimal results. Nonetheless, it's crucial to acknowledge that these outcomes are highly

Table 8: Experiment results of SIB-200 for five languages on two models (Aya Expanse (8B) and Qwen2.5 Instruct (7B)) under different methods and prompt-type configurations.

Method	**Prompt**	**Aya Expanse (8B)**					**Qwen2.5 (7B)**				
		deu	**ind**	**hin**	**sun**	**fuv**	**deu**	**ind**	**hin**	**sun**	**fuv**
TT-ZS	FT,LT,ST,TC	78.3	**79.78**	**80.72**	74.28	57.12	**76.32**	74.06	**77.56**	**74.27**	**56.57**
	Tr	**79.72**	78.48	80.07	**74.42**	**58.97**	75.76	**74.65**	77	71.73	54.13
TT-ICL	FT,LT,ST	**89.62**	**89.64**	**89.91**	**86.16**	**71.99**	88.11	**89.03**	**89.02**	**88.51**	**67.11**
	TC	70.87	71.72	67.24	66.99	54.69	87.10	83.36	81.96	84.27	67.04
	Tr	82.16	85.03	84.53	82.06	66.46	**89.38**	86.54	86.76	84.67	67.01
ZS	FT,LT,ST,TC	52.04	33.16	42.32	41.70	36.01	**72.96**	**77.92**	69.66	52.03	22.55
	Tr	**74.01**	**78.54**	**78.44**	**72.72**	**45.12**	71.42	74.71	**73.23**	**64.15**	**35.48**
ZS-ICA	FT,LT,TC	10.31	1.09	2.59	7.90	10.79	75.07	63.85	72.61	59.91	27.56
	ST	32.60	24.37	31.71	29.25	19.45	75.45	75.34	75.26	70.11	38.47
	Tr	**78.04**	**79.83**	**74.91**	**76.41**	**46.27**	**81.72**	**81.67**	**78.63**	**79.74**	**42.09**
ICL	FT,LT,ST	**90.30**	**90.18**	**89.91**	**80.03**	**55.55**	**91.45**	**86.87**	86.96	81.59	48.50
	TC	24.51	16.51	22.52	6.41	21.78	85.52	85.86	82.22	78.62	42.78
	Tr	85.86	87.69	88.11	75.08	51.92	86.97	83.76	**88.28**	**81.63**	**52.71**
ICL-ICA	FT	**87.45**	**90.44**	90.89	78.63	**55.80**	**91.17**	**87.08**	**88.19**	**84.36**	54.94
	LT	85.84	87.15	88.81	**79.09**	52.36	87.79	86.92	81.57	79.15	52.57
	ST	87.07	91.78	**91.46**	77.86	52.43	90.50	85.41	87.28	81.88	52.91
	TC	39.34	23.67	19.49	23.67	21.07	82.69	80.65	81.32	71.47	41.06
	Tr	83.53	85.42	85.06	74.94	48.27	87.28	84.69	87.91	82.83	**55.12**
X-ICL	FT,LT,ST	**86.25**	**87.67**	**87.00**	**79.97**	**51.62**	88.37	**86.93**	83.98	78.23	46.19
	TC	56.65	55.95	50.58	54.99	40.86	**89.01**	86.33	**86.39**	72.58	37.89
	Tr	80.44	82.80	81.51	65.90	41.64	85.69	84.35	84.21	**80.14**	**49.06**
X-ICL-ICA	FT	**86.16**	**89.90**	**85.73**	78.31	**55.17**	**90.54**	**89.13**	85.76	**84.41**	51.36
	LT	83.58	86.19	85.02	**80.12**	49.44	86.55	84.56	**86.37**	76.98	50.96
	ST	84.37	89.32	83.53	77.20	51.99	89.39	86.80	85.55	82.71	50.45
	TC	36.69	38.74	30.66	30.81	25.56	83.93	81.99	84.11	77.53	44.61
	Tr	77.58	82.96	82.38	71.69	50.68	85.79	88.85	84.34	83.33	**54.54**

context-dependent and vary on a case-by-case basis. For example, Qwen's performance on the SIB-200 dataset with ZS-ICA was notably robust and stable. Furthermore, results observed in the ICL-ICA and X-ICL-ICA settings were broadly analogous across different prompt types, with the only notable exception being a modest improvement for template-based prompts when used with the Aya model on the SIB-200 dataset.

Template-based prompts shine when you can supply a handful of in-context examples, especially under the TT-ICL or pure ICL setups. In these scenarios, template-based prompts consistently push high-resource Languages into the high-80s and low-90s. For example, on Belebele under ICL, Aya scores 84% on German and 31.5% on Nigerian Fulfulde, while Qwen hits 88% on German and 26% on Nigerian Fulfulde. However, under this setting template-based prompts failed to achieve best performance on Hindi, although having only marginal differences. On SIB-200 using Aya, ICL and TT-ICL are more superior compared to other prompts. This behavior is also shown in Qwen, but with more variability in mid-resource and low-resource languages. These prompts also show remarkable consistency across languages, rarely dipping more than 5 points between FT, LT, and ST variants. In zero-shot settings on SIB-200 they lag behind the Tr prompt by a huge margin. So, if you can’t supply in-context examples, template-based prompts aren’t ideal.

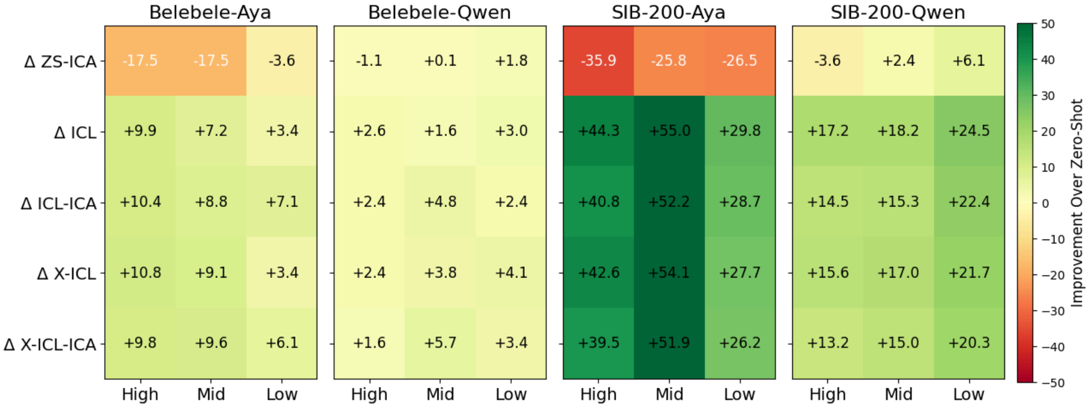

Fig. 3: Performance gain compared to zero-shot as baseline aggregated for every resource levels on (1) Belebele with Aya, (2) Belebele with Qwen, (3) SIB-200 with Aya, (4) SIB-200 with Qwen.

In general, the transition prompt is worse compared to the other two prompts because its instability. However, there are some cases where this prompt has the best performance, mostly on the Belebele benchmark. The transition prompt achieves best performance on German under X-ICL and X-ICL-ICA settings using Aya. It shows the capability of this prompt to handle relatively higher-resource language. It also shows in TT-ICL settings where this prompt achieves best performance on Indonesian, Hindi, and Sundanese, while also having good performance on German and Nigerian Fulfulde. On SIB-200 benchmark using Qwen, this prompt also has good performance in X-ICL and X-ICL-ICA with results more varying. Despite some good performances, this prompt performs poorly on other settings and SIB-200 dataset. So, we don't recommend using this prompt.

When there are no in-context examples available or zero-shot efficiency is needed, the traditional prompt is recommended. On SIB-200 benchmark using Aya, traditional prompt outperforms template-based prompts with huge margin on every language. On SIB-200 benchmark using Qwen, traditional prompt outperforms template-based prompts with lower margin in ZS-ICA setting. It also outperforms template-based prompts on lower-resource languages (Hindi, Sundanese, Nigerian Fulfulde) in ZS setting. This shows that topic classification with ZS and ZS-ICA settings don't need the formatting consistency given by template-based prompts. On the other hand, the results on Belebele benchmark don't show a clear correlation between these two prompts in ZS and ZS-ICA settings. Performance on higher-resource languages (German and Indonesian) mostly achieve better results using traditional prompt, but it doesn't generalize to all settings. Furthermore, the gain of traditional prompt in TT-ZS also remains unclear as the results vary with different models and languages.

4.2. Language Resource Level Impacts In-Context Information Effectiveness

Our evaluation involved two datasets and two distinct models, utilizing the LT prompt format consistently across all eight of our in-context methods. We observed that the translate-test method consistently proved to be one of the most stable approaches, as detailed in Tables 3, 4, 5, and 6. Remarkably, even the translate-test zero-shot method surpassed both ICL and X-ICL in performance for low-resource languages. However, this method's heavy reliance on machine translation models presents a significant limitation, as these models aren't always available for such languages. Consequently, our primary focus shifted toward methods that operate independently of machine translation models, as illustrated in Fig. 3.

We found that performance improvements on the Belebele dataset were less substantial compared to the SIB-200 dataset. This indicates that retrieving semantically similar examples remains effective for classification tasks when using human-aligned LLMs. Conversely, machine reading comprehension, like that in the Belebele dataset, is a more intricate task, demanding context-dependent reasoning for accurate answers. This suggests that while human-aligned LLMs can still reason from examples, their overall impact on such complex tasks is somewhat constrained. Further supporting this, ICL consistently performed best for SIB-200 across different models and resource levels, whereas Belebele often yielded better results in X-ICL settings, possibly because LLMs process information more effectively in English. This is also evident from the Belebele results using the Aya model,

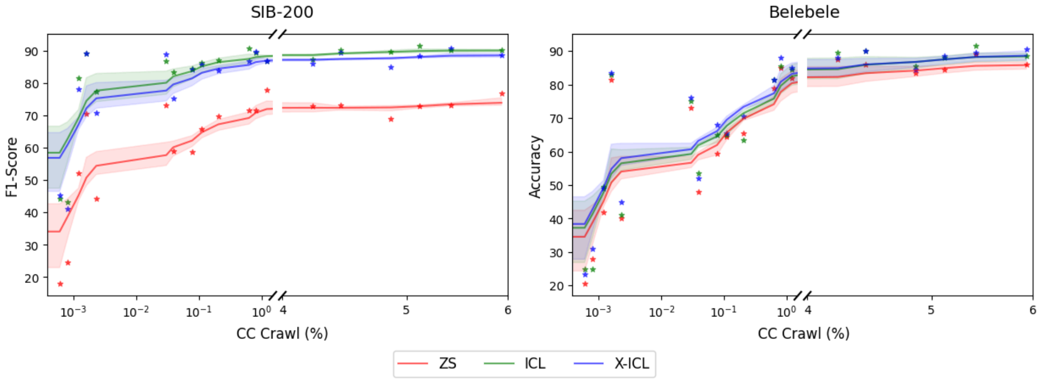

Fig. 4: Performance comparison between different language resource level on SIB-200 and Belebele Dataset Using ZS, ICL, and X-ICL

which show that LLMs perform better with high-resource languages. Conversely, the performance improvement on SIB-200 for high-resource languages was comparatively smaller, likely because LLMs already demonstrate strong zero-shot capabilities in these languages.

As illustrated in Fig. 3, it is clear that the effectiveness of ICA is not uniform but is instead highly dependent on the specific language model being used. For instance, employing ICA with the Aya model leads to a significant degradation in zero-shot performance. In contrast, the Qwen model demonstrates a beneficial impact from ICA in the same zero-shot setting, an advantage that is particularly pronounced for low-resource languages. This divergence strongly suggests that different LLMs develop their own distinct internal multilingual representations. Focusing specifically on the Belebele dataset, the benefits of ICA become even more nuanced. It enhances the Aya model's performance on ICL and X-ICL tasks for low-resource languages, while it boosts the Qwen model's performance for mid-resource languages. These findings converge on the conclusion that ICA is most advantageous when applied to languages with low to medium resource availability. For languages with extensive resources, the additional context provided by alignment may become superfluous or even counterproductive, potentially confusing the model with information it already comprehends well.

Fig. 4 illustrates a distinct and positive correlation between the volume of available data for a language and the performance of the LLMs. As the level of resources increases from low to high, there is a corresponding rise in both accuracy and f1-scores. It is particularly noteworthy that mid-resource languages frequently achieve performance levels on par with their high-resource counterparts, especially within the SIB-200 benchmark. This trend indicates that once a certain threshold of data availability is crossed, the performance benefits from additional resources begin to diminish. By closely examining the performance breakpoints in Table 3 through 6, we can pinpoint where this drop-off occurs. For the Belebele dataset (Table 3–4), performance declines sharply for languages with fewer resources than Ukrainian, whereas for the SIB-200 benchmark (Table 5–6), this significant drop only manifests for languages with fewer resources than Macedonian. This pattern strongly suggests that the latest generation of human-aligned LLMs has developed a sufficient level of robustness to process mid-resource languages with great accuracy, nearly eliminating the performance gap that once separated them from high-resource languages.

From these drop-off points we can infer a practical resource threshold for current LLMs which is language with roughly a Joshi’s Class value of 4 (which corresponds to a certain typological and corpus-size bracket) and at least 0.5 % coverage in the Common Crawl dataset. Languages meeting or exceeding these thresholds tend to yield stable, high performance, even in zero-shot runs, whereas those below them see a rapid degradation. In other words, once a language has a certain resource level for development and a minimal corpus footprint online, modern human-aligned LLMs can understand and reason over it nearly as well as high-resource languages.

Turning to the truly low-resource end, there is a striking performance gap between Asian regional languages like Javanese (jav), Sundanese (sun), and Sindhi (snd) compared to African languages such as Yoruba (yor), Igbo (ibo), and Nigerian Fulfulde (fuv). Although their Common Crawl percentages are similar, the latter group under-performs by 10–20 points, highlighting an under-representation of African language varieties in pre-training data,

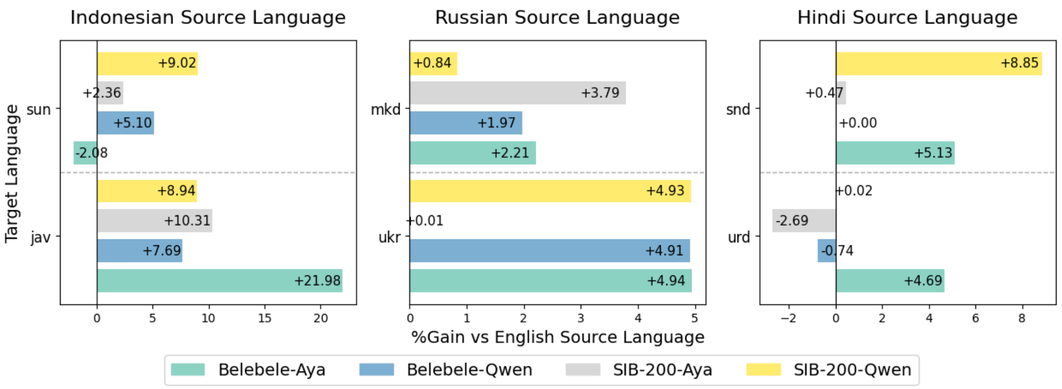

Fig. 5: Percentage gain of different source languages compared to English. Javanese and Sundanese using Indonesian as source language (Left). Ukrainian and Macedonian using Russian as source language (Center). Urdu and Sindhi using Hindi as source language (Right).

potentially compounded by dialectal diversity and fewer high-quality texts. This is shown by the steep gradient in the leftmost part in Fig. 4. This disparity underscores the need for more curated corpora and targeted data augmentation for these under-served languages. Method choices between ZS, ICL, and X-ICL have the same trend of performance with respect to language resource level. The only different is there is a larger gap between ZS on SIB-200 benchmark compared to Belebele.

4.3. Similar Source-Target Language Enhances In-Context Transferability

In an evaluation conducted across two distinct datasets and two LLMs, we investigated the effect of source language choice in X-ICL and ICA by comparing English against source languages that are more similar to the target language in terms of linguistic family and geographical region. For this experiment, we consistently employed the LT prompt format for all in-context methods. The results, as depicted in Fig. 5, generally demonstrate that using a more linguistically similar source language enhances X-ICL performance when compared to using English. The most significant improvement was observed when the source language for Javanese was switched to Indonesian, a change that yielded performance gains across all tested models and datasets. We noted only three instances of a slight performance decrease, specifically for Sundanese and Urdu. This pattern strongly suggests that Large LLMs tend to form more closely related internal representations for languages that are linguistically related. Furthermore, we sought to determine the relative importance of linguistic similarity versus script similarity. To do this, we altered the source language for Sindhi and Urdu from Hindi to Arabic, as detailed in Tables 9 and 10. The findings revealed that both factors contribute similarly, with a marginally greater improvement attributable to direct linguistic similarity over shared writing systems.

A comprehensive analysis of Table 9 and 10 confirms that selecting a source language with close linguistic ties to the target language consistently enhances the performance of ICA. However, the more variable outcomes observed on the Belebele dataset highlight how the inherent difficulty of a given task significantly influences the extent of these performance gains. Specifically, tasks with lower complexity tend to receive more substantial boosts in performance, whereas more complex ones benefit to a lesser degree. From a practical standpoint, this indicates that while ICA can effectively harness linguistic similarities to improve answer accuracy, the overall impact of this alignment is ultimately smaller than the performance gains achievable through a well-chosen source language in a X-ICL setup. Consequently, the strategic and careful selection of an optimal source language, preferably one that is better-resourced than the target language becomes a crucial consideration. This approach enables the model to maximize performance improvements without needing to increase the number of in-context examples, thereby striking an effective balance between data efficiency and overall model effectiveness.

4.4. Importance of Semantically Similar ICL Examples

We compare the effect of ICL examples retrieval techniques on TT-ICL, ICL, and X-ICL evaluated on SIB-200 using both models. We use chat-template as our prompt formatting. We sample 6 languages (hin, ben, urd, snd, yor, ibo) consisting of 3 mid-resource languages and 3 low-resource languages. We choose these languages because we hypothesize that semantically similar examples have more impact on mid- and low-resource language as LLMs

Table 9: Experiment results of Belebele for six languages on two models (Aya Expanse (8B) and Qwen2.5 Instruct (7B)) under different source languages.

Target	**Source**	**Aya Expanse (8B)**						**Qwen2.5 (7B)**					
		ZS	**ZS-ICA**	**ICL**	**ICL-ICA**	**X-ICL**	**X-ICL-ICA**	**ZS**	**ZS-ICA**	**ICL**	**ICL-ICA**	**X-ICL**	**X-ICL-ICA**
jav	eng	46.5	43	48.5	54	45.5	58	48	**48.5**	53.5	**52.5**	52	51
	ind		**46**		**56**	**55.5**	**58.5**		46.5		**52.5**	**56**	**57**
sun	eng	36	**33**	44.5	**48**	**48**	45	42	41.5	49.5	47.5	49	**47.5**
	ind		32.5		47.5	47	**50**		**42.5**		46.5	**51.5**	**48**
ukr	eng	75	50	82	**82.5**	81	**83.5**	79	79.5	81.5	**84.5**	81.5	85
	rus		**53**		**82.5**	**85**	82.5		**80.5**		82.5	**85.5**	**87**
mkd	eng	55	**42**	59.5	**69**	68	69.5	73	**74**	75	**74.5**	76	**77.5**
	rus		39.5		67.5	**69.5**	**72.5**		70.5		73.5	**77.5**	73
urd	eng	49.5	31	56	59.5	64	62	59.5	**64**	65	**69.5**	**68**	**72**
	hin		**33.5**		**60**	**67**	**65.5**		62		65	67.5	66.5
	arb		**33.5**		58	63.5	60.5		62		65	66.5	69.5
snd	eng	37.5	**31.5**	40.5	**46**	39	38.5	40	**44**	41	**46**	**45**	**45**
	hin		31		41.5	**41**	**46**		43.5		45	**45**	42.5
	arb		**31.5**		38.5	36.5	43		**44**		45	38.5	42.5

have less understanding in these settings. We only experiment with SIB-200 dataset as the gains from ICL examples are bigger compared to Belebele. We use 4 different ICL examples retrieval techniques which are static, random, top-class, and top-k. For static technique, we choose 7 examples from each class and use it as our examples for all test data. For random technique, we choose 7 random examples from each class everytime we run inference on test data. For top-class techniques, we use semantic textual similarity and choose the most similar example for each class. For top-k techniques, we use semantic textual similarity and choose 7 most similar examples, not necessarily from different classes. We always use top-k techniques for our previous experiments because it has been proven that semantically similar examples boost ICL performance [46]. However, we further ablate this technique under different language resource levels and models.

As shown in Fig. 6, the gain between changing ICL examples retrieval techniques remain consistent across different resource levels. Performance in low-resource languages remain significantly lower, but the retrieval techniques performance remain the same with top-k > top-class > random > static. The difference between each ICL retrieval techniques are around 1-3% in f1-score. Semantically similar ICL examples remain important in human-aligned LLMs. However, the use of top-k and top-class techniques need resources such as ICL examples dataset, semantic textual similarity model, and increased computation time due to semantic retrieval. When efficiency is more important than performance, random examples can be an alternative if ICL example dataset is available. However, in extreme low-resource settings where ICL examples dataset is unavailable, static examples can be an alternative with less than 10% drop in performance. Furthermore, if machine translation model is available, the effectiveness of ICL examples retrieval decrease. In this case, the use of random examples can be considered for the sake of efficiency.

As shown in Table 11, ICL example retrieval technique using top-k is still the best option in the ICL and X-ICL settings. However, the gains decreased in TT-ICL setting as the results varied more between top-k and top-class techniques with only little margins. This shows that ICL examples matters less when the query is already translated into English highlighting the robustness of TT-ICL technique. The performance gap between mid- and low-resource languages is also smaller in TT-ICL setting as Yoruba and Igbo achieve performance over 80% when using TT-ICL. The use of top-class ICL examples retrieval techniques may confuse models as they are given similar sentences which have different labels. It shows on the results for ICL and X-ICL as top-class has lower performance compared

Table 10: Experiment results of SIB-200 for six languages on two models (Aya Expanse (8B) and Qwen2.5 Instruct (7B)) under different source languages.

		Aya Expanse (8B)						Qwen2.5 (7B)					
Target	**Source**	**ZS**	**ZS-ICA**	**ICL**	**ICL-ICA**	**X-ICL**	**X-ICL-ICA**	**ZS**	**ZS-ICA**	**ICL**	**ICL-ICA**	**X-ICL**	**X-ICL-ICA**
jav	eng	39.71	**6.08**	78.83	79.67	75.54	74.88	59.06	58.34	83.43	**75.85**	75.14	72.76
	ind		5.88		**81.37**	**83.33**	**79.90**		**60.29**		75.49	**81.86**	**75.49**
sun	eng	41.70	**7.90**	80.03	79.09	79.97	80.12	52.03	**59.91**	81.59	79.15	78.23	76.98
	ind		5.39		**81.86**	**81.86**	**81.37**		56.37		**84.31**	**85.29**	**81.37**
ukr	eng	23.98	4.56	87.78	85.18	89.21	87.49	71.59	75.22	90.63	86.40	86.90	85.18
	rus		**5.39**		**89.22**	**89.22**	**87.75**		**79.41**		**87.75**	**91.18**	**88.24**
mkd	eng	36.65	**7.33**	85.2	82.11	83.12	79.69	73.05	70.60	86.94	83.06	88.96	82.31
	rus		3.92		**85.29**	**86.27**	**84.80**		**76.47**		**87.25**	**89.71**	**87.25**
urd	eng	25.07	**6.12**	82.11	79.67	**83.12**	**79.69**	58.80	**75.55**	84.38	82.46	84.29	81.56
	hin		2.94		**83.33**	80.88	79.41		75.49		**82.84**	84.31	82.35
	arb		3.43		82.35	81.37	77.94		75.00		**82.84**	**85.78**	**84.31**
snd	eng	35.88	**6.90**	72.30	72.81	72.21	68.28	44.38	58.03	77.23	75.23	70.70	70.50
	hin		3.43		**75**	72.55	**72.55**		51.96		**76.47**	**76.96**	**77.45**
	arb		3.43		74.02	**73.53**	71.57		**58.82**		75.00	73.53	75.49

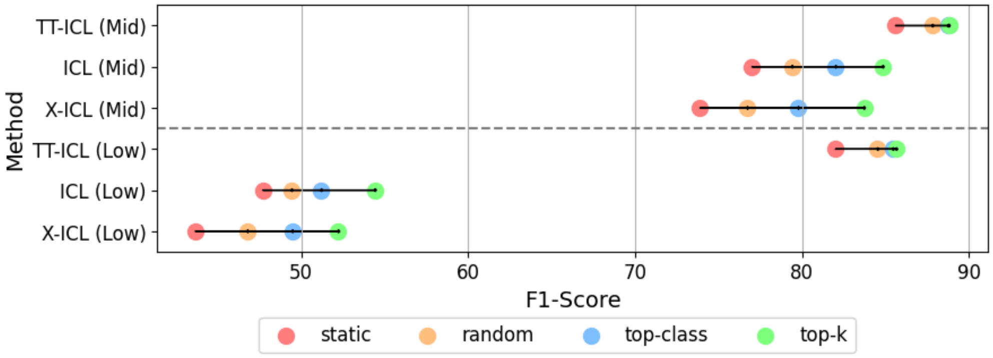

Fig. 6: Comparison of ICL examples retrieval techniques between mid- and low-resource languages

to top-k, although they both use the same semantic textual similarity model. However, this result is not seen in TT-ICL setting showing that models reason better when provided with English query. In general, static examples have the worst performance, while random examples are the second worst. These results show that semantically similar ICL examples are still relevant, especially in ICL and X-ICL settings.

5. Conclusion

Modern human-aligned multilingual LLMs continue to benefit from in-context information, though under specific conditions: modern prompt templates consistently outperform traditional formats in In-Context Learning (ICL) and Cross-Lingual In-Context Learning (X-ICL), while "transition" prompts can significantly reduce performance, highlighting LLMs' sensitivity to prompt structure, with traditional formats often faring better in zero-shot settings. In-Context Alignment (ICA) remains valuable, primarily for mid- to low-resource languages, but its effectiveness strongly depends on both the model and the prompting strategy. For instance, ICA is reducing zero-shot performance in Aya while improving it in Qwen2.5. Swapping English for a linguistically closer source language in X-ICL can improve performance and serves as an alternative when data in more similar languages is available. Furthermore, the task type significantly impacts ICL performance; while ICL generally works well for classification tasks, it offers limited benefits for reasoning-intensive benchmarks like Belebele, particularly for low-resource languages. Semantically similar ICL examples remain relevant in topic classification, especially in

Table 11: Experiment results of SIB-200 for six languages on two models (Aya Expanse (8B) and Qwen2.5 Instruct (7B)) under different ICL retrieval techniques.

Method	**ICL Retrieval**	**Aya Expanse (8B)**						**Qwen2.5 (7B)**					
		hin	**ben**	**urd**	**snd**	**yor**	**ibo**	**hin**	**ben**	**urd**	**snd**	**yor**	**ibo**
TT-ICL	static	86.23	86.11	82.79	84.13	78.92	84.89	86.23	85.74	86.51	84.97	77.76	81.34
	random	88.15	**88.94**	85.02	87.63	80.76	86.55	88.32	88.07	88.53	87.49	80.22	84.51
	top-class	89.52	88.13	**87.44**	88.23	**81.80**	**87.90**	**89.57**	88.23	89.75	87.34	**82.59**	84.87
	top-k	**89.91**	88.68	85.38	**89.15**	81.62	87.55	89.02	**89.41**	**90.41**	**89.38**	81.22	**85.03**
ICL	static	83.15	70.12	72.82	64.39	37.17	38.55	80.21	80.29	75.41	67.32	38.75	39.96
	random	86.80	70.93	75.01	65.03	41.21	41.35	83.67	80.76	79.28	69.58	40.83	38.60
	top-class	87.58	77.69	76.24	71.58	41.46	42.08	84.78	85.55	80.30	70.32	40.57	41.16
	top-k	**89.91**	**79.33**	**82.11**	**72.30**	**43.91**	**45.54**	**86.96**	**86.39**	**84.38**	**77.23**	**43.24**	**44.3**
X-ICL	static	79.8	64.33	69.42	60.09	33.9	34.18	78.55	79.73	71.53	60.49	34.55	38.61
	random	84.32	67.31	72.11	62.81	38.94	39.44	81.64	78.36	76.66	64.05	37.91	37.5
	top-class	86.34	73.52	75.35	**72.57**	**40.98**	40.14	81.11	84.81	77.56	63.32	37.4	42.38
	top-k	**87**	**78.13**	**83.12**	72.21	39.11	**44.92**	**83.98**	**85.79**	**84.29**	**70.7**	**41.04**	**45.41**

ICL and X-ICL settings, but their effectiveness drops in TT-ICL when machine translation is used. Finally, when efficiency is crucial, consider using simpler ICL example retrieval, especially for complex tasks requiring reasoning or for mid-resource languages where a machine translation model is available.

CRediT Authorship Contribution Statement

Ubaidillah Ariq Prathama: Conceptualization, Methodology. Software, Formal analysis, Writing – Original Draft, Writing – Review & Editing. **Ayu Purwarianti**: Conceptualization, Methodology, Writing – Review & Editing, Supervision. **Samuel Cahyawijaya**: Conceptualization, Methodology, Writing – Review & Editing, Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

The dataset was openly provided on https://huggingface.co/datasets/Davlan/sib200, https://huggingface.co/datasets/facebook/belebele, and https://huggingface.co/datasets/Muennighoff/flores200.

Declaration of Generative AI and AI-assisted Technologies in The Writing Process

The authors used generative AI to improve the writing clarity of this paper. They reviewed and edited the AI-assisted content and take full responsibility for the final publication.

References

[1] OpenAI *et al.*, "GPT-4 Technical Report," Mar. 2023, [Online]. Available: https://arxiv.org/abs/2303.08774

[2] OpenAI *et al.*, "OpenAI o1 System Card," 2024, [Online]. Available: https://arxiv.org/abs/2412.16720

[3] DeepSeek-AI, "DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning," 2025, [Online]. Available: https://arxiv.org/abs/2501.12948

[4] T. Cohere *et al.*, "Command A: An Enterprise-Ready Large Language Model," 2025, [Online]. Available: https://arxiv.org/abs/2504.00698

[5] A. Dubey *et al.*, "The Llama 3 Herd of Models," Jul. 2024, [Online]. Available: https://arxiv.org/abs/2407.21783

[6] G. Team *et al.*, "Gemma 3 Technical Report," 2025. [Online]. Available: https://arxiv.org/abs/2503.19786

[7] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning*, 2017, pp. 1126–1135, [Online]. Available: https://proceedings.mlr.press/v70/finn17a.html

[8] G. I. Winata *et al.*, "Learning Fast Adaptation on Cross-Accented Speech Recognition," in *Proc. Interspeech 2020*, 2020, pp. 1276–1280, doi: 10.21437/Interspeech.2020-45.

[9] K. Hämmerl, J. Libovický, and A. Fraser, "Understanding Cross-Lingual Alignment—A Survey," in *Findings of the Association for Computational Linguistics: ACL 2024*, L.-W. Ku, A. Martins, and V. Srikumar, Eds., Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 10922–10943. doi: 10.18653/v1/2024.findings-acl.649.

[10] Z. Liu *et al.*, "CrossNER: Evaluating Cross-Domain Named Entity Recognition," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 15, pp. 13452–13460, May 2021, doi: 10.1609/aaai.v35i15.17587.

[11] S. Cahyawijaya, H. Lovenia, W. Chung, R. Frieske, Z. Liu, and P. Fung, "Cross-Lingual Cross-Age Adaptation for Low-Resource Elderly Speech Emotion Recognition," in *Proc. Interspeech 2023*, 2023, pp. 3352–3356, doi: 10.21437/Interspeech.2023-327.

[12] S. Cahyawijaya, H. Lovenia, and P. Fung, "LLMs Are Few-Shot In-Context Low-Resource Language Learners," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2024, pp. 405–433. doi: 10.18653/v1/2024.naacl-long.24.

[13] B. Wilie, S. Cahyawijaya, J. He, and P. Fung, "High-Dimensional Interlingual Representations of Large Language Models," in *Proceedings of the 7th Workshop on Research in Computational Linguistic Typology and Multilingual NLP*, Aug. 2025, pp. 122–155, doi: 10.18653/v1/2025.sigtyp-1.14.

[14] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic BERT Sentence Embedding," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds., Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 878–891. doi: 10.18653/v1/2022.acl-long.62.

[15] S. Cahyawijaya, H. Lovenia, T. Yu, W. Chung, and P. Fung, "InstructAlign: High-and-Low Resource Language Alignment via Continual Crosslingual Instruction Tuning," in *Proceedings of the First Workshop in South East Asian Language Processing*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2023, pp. 55–78. doi: 10.18653/v1/2023.sealp-1.5.

[16] P. Chen, S. Yu, Z. Guo, and B. Haddow, "Is It Good Data for Multilingual Instruction Tuning or Just Bad Multilingual Evaluation for Large Language Models?," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds., Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 9706–9726. doi: 10.18653/v1/2024.emnlp-main.542.

[17] U. Shaham, J. Herzig, R. Aharoni, I. Szpektor, R. Tsarfaty, and M. Eyal, "Multilingual Instruction Tuning With Just a Pinch of Multilinguality," in *Findings of the Association for Computational Linguistics: ACL 2024*, L.-W. Ku, A. Martins, and V. Srikumar, Eds., Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 2304–2317. doi: 10.18653/v1/2024.findings-acl.136.

[18] T. Brown *et al.*, "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., Curran Associates, Inc., 2020, pp. 1877–1901, [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[19] A. Chowdhery *et al.*, "PaLM: scaling language modeling with pathways," *J. Mach. Learn. Res.*, vol. 24, no. 1, Jan. 2023, [Online]. Available: https://www.jmlr.org/papers/v24/22-1144.html

[20] R. Zhang, S. Cahyawijaya, J. C. B. Cruz, G. Winata, and A. Aji, "Multilingual Large Language Models Are Not (Yet) Code-Switchers," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2023, pp. 12567–12582. doi: 10.18653/v1/2023.emnlp-main.774.

[21] G. I. Winata, A. Madotto, Z. Lin, R. Liu, J. Yosinski, and P. Fung, "Language Models are Few-shot Multilingual Learners," in *Proceedings of the 1st Workshop on Multilingual Representation Learning*, D. Ataman, A. Birch, A. Conneau, O. Firat, S. Ruder, and G. G. Sahin, Eds., Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 1–15. doi: 10.18653/v1/2021.mrl-1.1.

[22] X. V. Lin *et al.*, "Few-shot Learning with Multilingual Generative Language Models," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2022, pp. 9019–9052. doi: 10.18653/v1/2022.emnlp-main.616.

[23] A. Asai *et al.*, "BUFFET: Benchmarking Large Language Models for Few-shot Cross-lingual Transfer," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2024, pp. 1771–1800. doi: 10.18653/v1/2024.naacl-long.100.

[24] Y. C. Bilge, N. Ikizler-Cinbis, and R. G. Cinbis, "Cross-lingual few-shot sign language recognition," *Pattern Recognition*, vol. 151, p. 110374, 2024, doi: 10.1016/j.patcog.2024.110374.

[25] E. Tanwar, S. Dutta, M. Borthakur, and T. Chakraborty, "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)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2023, pp. 6292–6307. doi: 10.18653/v1/2023.acl-long.346.

[26] B. Workshop *et al.*, "BLOOM: A 176B-Parameter Open-Access Multilingual Language Model," Nov. 2022, [Online]. Available: https://arxiv.org/abs/2211.05100

[27] R. Pan, J. Antonio García-Díaz, and R. Valencia-García, "Comparing Fine-Tuning, Zero and Few-Shot Strategies with Large Language Models in Hate Speech Detection in English," *CMES - Computer Modeling in Engineering and Sciences*, vol. 140, no. 3, pp. 2849–2868, 2024, doi: 10.32604/cmes.2024.049631.

[28] J. Wei *et al.*, "Finetuned Language Models are Zero-Shot Learners," in *International Conference on Learning Representations*, 2022, [Online]. Available: https://openreview.net/forum?id=gEZrGCozdqR

[29] V. Sanh *et al.*, "Multitask Prompted Training Enables Zero-Shot Task Generalization," in *International Conference on Learning Representations*, 2022, [Online]. Available: https://openreview.net/forum?id=9Vrb9D0WI4

[30] H. W. Chung *et al.*, "Scaling instruction-finetuned language models," *J. Mach. Learn. Res.*, vol. 25, no. 1, Jan. 2024, [Online]. Available: https://www.jmlr.org/papers/v25/23-0870.html

[31] S. Cahyawijaya *et al.*, "Cendol: Open Instruction-tuned Generative Large Language Models for Indonesian Languages," in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, L.-W. Ku, A. Martins, and V. Srikumar, Eds., Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 14899–14914. doi: 10.18653/v1/2024.acl-long.796.

[32] R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, and C. Finn, "Direct Preference Optimization: Your Language Model is Secretly a Reward Model," 2024, [Online]. Available: https://arxiv.org/abs/2305.18290

[33] D. Zhu, S. Trenous, X. Shen, D. Klakow, B. Byrne, and E. Hasler, "A Preference-driven Paradigm for Enhanced Translation with Large Language Models," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, K. Duh, H. Gomez, and S. Bethard, Eds., Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 3385–3403. doi: 10.18653/v1/2024.naacl-long.186.

[34] S. S. Ramesh *et al.*, “Group Robust Preference Optimization in Reward-free RLHF,” in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024, [Online]. Available: https://openreview.net/forum?id=PRAsjrmXXK

[35] E. Choi, A. Ahmadian, M. Geist, O. Pietquin, and M. G. Azar, “Self-Improving Robust Preference Optimization,” 2025, [Online]. Available: https://arxiv.org/abs/2406.01660

[36] Y. Bai *et al.*, “Constitutional AI: Harmlessness from AI Feedback,” 2022, [Online]. Available: https://arxiv.org/abs/2212.08073

[37] S. Yao *et al.*, “ReAct: Synergizing Reasoning and Acting in Language Models,” in *The Eleventh International Conference on Learning Representations* , 2023, [Online]. Available: https://openreview.net/forum?id=WE_vluYUL-X

[38] X. Wang, L. Caccia, O. Ostapenko, X. Yuan, W. Y. Wang, and A. Sordoni, “Guiding Language Model Reasoning with Planning Tokens,” in *First Conference on Language Modeling*, 2024, [Online]. Available: https://openreview.net/forum?id=wi9IffRhVM

[39] S. Goyal, Z. Ji, A. S. Rawat, A. K. Menon, S. Kumar, and V. Nagarajan, “Think before you speak: Training Language Models With Pause Tokens,” in *The Twelfth International Conference on Learning Representations*, 2024, [Online]. Available: https://openreview.net/forum?id=ph04CRkPdC

[40] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, “Deep reinforcement learning from human preferences,” *Adv Neural Inf Process Syst*, vol. 30, 2017, [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/d5e2c0adad503c91f91df240d0cd4e49-Paper.pdf

[41] H. Lee *et al.*, “RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback,” 2024, [Online]. Available: https://arxiv.org/abs/2309.00267.

[42] A. Ahmadian *et al.*, “Back to Basics: Revisiting REINFORCE-Style Optimization for Learning from Human Feedback in LLMs,” in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, L.-W. Ku, A. Martins, and V. Srikumar, Eds., Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 12248–12267. doi: 10.18653/v1/2024.acl-long.662.

[43] Qwen *et al.*, “Qwen2.5 Technical Report,” Dec. 2024, [Online]. Available: https://arxiv.org/abs/2412.15115

[44] J. Dang *et al.*, “Aya Expanse: Combining Research Breakthroughs for a New Multilingual Frontier,” 2024, [Online]. Available: https://arxiv.org/abs/2412.04261.

[45] J. Wei *et al.*, “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models,” in *Advances in Neural Information Processing Systems*, 2022, vol. 35, pp. 24824–24837, [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf.

[46] J. Liu, D. Shen, Y. Zhang, B. Dolan, L. Carin, and W. Chen, “What Makes Good In-Context Examples for GPT-3?,” in *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, E. Agirre, M. Apidianaki, and I. Vulić, Eds., Dublin, Ireland and Online: Association for Computational Linguistics, May 2022, pp. 100–114. doi: 10.18653/v1/2022.deelio-1.10.

[47] K. Lyu, H. Zhao, X. Gu, D. Yu, A. Goyal, and S. Arora, “Keeping LLMs Aligned After Fine- tuning: The Crucial Role of Prompt Templates,” in *Advances in Neural Information Processing Systems*, 2024, vol. 37, pp. 118603–118631, [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2024/file/d6f034bb216b472fc7d32ec7aff20342-Paper-Conference.pdf.

[48] P. Joshi, S. Santy, A. Budhiraja, K. Bali, and M. Choudhury, “The State and Fate of Linguistic Diversity and Inclusion in the NLP World,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Online: Association for Computational Linguistics, Jul. 2020, pp. 6282–6293. doi: 10.18653/v1/2020.acl-main.560.

[49] Y. Graham and M. Purver, “SIB-200: A Simple, Inclusive, and Big Evaluation Dataset for Topic Classification in 200+ Languages and Dialects,” in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Mar. 2024, pp. 226–245, doi: 10.18653/v1/2024.eacl-long.14.

[50] L. Bandarkar *et al.*, “The Belebele Benchmark: a Parallel Reading Comprehension Dataset in 122 Language Variants,” in *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2024, pp. 749–775. doi: 10.18653/v1/2024.acl-long.44.

[51] NLLB Team *et al.*, “No Language Left Behind: Scaling Human-Centered Machine Translation,” Jul. 2022, [Online]. Available: https://arxiv.org/abs/2207.04672.

[52] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 3980–3990. doi: 10.18653/v1/D19-1410.