Prompting with Phonemes: Enhancing LLMs' Multilinguality for non-Latin Script Languages

Hoang Nguyen[1*] **Khyati Mahajan**[2] **Vikas Yadav**[2] **Julian Salazar**[3]
Philip S. Yu[1] **Masoud Hashemi**[2] **Rishabh Maheshwary**[2]

[1]University of Illinois at Chicago [2]ServiceNow [3]Google DeepMind

`{hnguy7,psyu}@uic.edu` `julsal@google.com`

`{khyati.mahajan,vikas.yadav,masoud.hashemi,rishabh.maheshwary}@servicenow.com`

Abstract

Multilingual LLMs have achieved remarkable benchmark performance, but we find they continue to underperform on non-Latin script languages across contemporary LLM families. This discrepancy arises from the fact that LLMs are pretrained with orthographic scripts, which are dominated by Latin characters that obscure their shared phonology with non-Latin scripts. We propose leveraging phonemic transcriptions as complementary signals to induce script-invariant representations. Our study demonstrates that integrating phonemic signals improves performance across both non-Latin and Latin languages, with a particularly significant impact on closing the performance gap between the two. Through detailed experiments, we show that phonemic and orthographic scripts retrieve distinct examples for in-context learning (ICL). This motivates our proposed `Mixed-ICL` retrieval strategy, where further aggregation from both leads to our significant performance improvements for both Latin script languages (up to 12.6%) and non-Latin script languages (up to 15.1%) compared to randomized ICL retrieval.

1 Introduction

Large language models (LLMs) have demonstrated remarkable multilingual capabilities across various natural language processing (NLP) tasks. The increase in model parameters and rise of instruction datasets have led to the emergent capability of LLMs to perform tasks in few to zero shots (Brown et al., 2020; Xia et al., 2020; Wei et al., 2022; Nguyen et al., 2023a) or adapt efficiently to new tasks through in-context learning (ICL) during inference (Zoph et al., 2022). However, these capabilities remain disparate across languages (Lai et al., 2023a), with one particular axis being along non-Latin versus Latin script languages (Bang et al., 2023; Ahuja et al., 2023; Shliazhko et al., 2024).

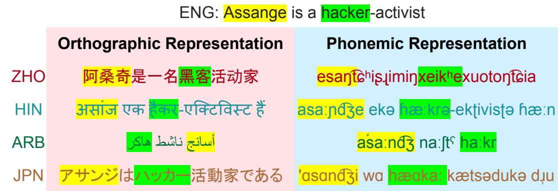

Figure 1: Orthographic and phonemic transcriptions (via the International Phonetic Alphabet; IPA) of the same sentence. Matching colors denote semantically similar words across different languages.

To mitigate this disparity, we are motivated by the crucial role of phonemic awareness in human language acquisition and processing, facilitating skills like cross-lingual transfer and reading development (Durgunoĝlu et al., 1993; Spencer and Hanley, 2003), in part due to cognates, borrowed words, and shared phonology between language families. We hypothesize that integrating phonemic information could also enable LLMs' robustness to the various choices of linguistic written script by capturing such alignments. For instance, in Figure 1, the Japanese orthographic representation[1] for *hacker* (ハッカー) is significantly different as compared to English. However, when observing the phonemic transcriptions–specifically, International Phonetic Alphabet (IPA) transcriptions at the level of phoneme discrimination– one could easily recognize the semantically similar words (hæɑkaː) highlighted in green in Figure 1.

On the other hand, the current prompting and ICL schemes rely solely on orthographic text input, overlooking potentially valuable linguistic information encoded in the phonemic structure of language. While continual pretraining with phonemic data could enhance LLMs' multilingual capabilities, it faces several challenges. One significant limitation is the scarcity of large-scale multilingual datasets that align orthographic and phonemic tran-

*Work done during an internship at ServiceNow.

[1]*Orthographic representation* and *textual / written scripts* are used interchangeably throughout this work.

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 11975–11994
April 29 - May 4, 2025

scriptions across diverse languages, especially for less-resourced languages. This lack of aligned data restricts the potential for fine-tuning models on phonemic information at scale. Furthermore, pre-training with phonemic data demands substantial computational resources due to the increased data size and complexity, requiring extensive training time on high-performance infrastructure.

Hence, we propose the phonemic integration in LLMs via prompting and ICL, as these promise a more flexible and resource-efficient approach to realize the integration's benefits. We hypothesize that augmenting with phonemic information could improve both demonstration retrieval and LLM reasoning, by explicitly surfacing fundamental cross-linguistic information (Anderson, 2018) that textual scripts might not capture or may not be readily accessible in the LLMs' internal representations. Our contributions include:

- Evaluating multilingual performance across contemporary ($\geq$7B-parameter) LLM *families* and diverse sets of tasks, with specific focus on Latin vs. non-Latin scripts, revealing a significant gap on evaluation metrics (up to 29 absolute points). We then focus our work on notable performance disparities in key generative tasks such as text generation (AYA-WIKI), machine translation (FLORES) and question-answering (AYA-MLQA).

- Investigating the integration of IPA with LLMs via (1) direct prompting (zero- and few-shot) and (2) in retrieval-based ICL augmentation. In particular, we find that aggregation of simple lexical retrieval on the text and phonemes (namely `Mixed-ICL`) gives performance gains of up to 15.1% relative on generative tasks, together with gains on Latin languages at inference time. Qualitative analyses examine retrieved cases and validate the observed empirical performance gains.

- Performing in-depth analyses of the components involved in phonemic integration, offering insights and guidance for future work aiming to improve LLMs through phonemic integration beyond inference time.

2 Background and Related Work

Phonemes are considered the smallest units of speech distinguishing one word (or word element) from another.[2] Since LLMs are mostly trained on text, they might implicitly encode some information from the phonemes present in the text; however, investigative research on the phonemic awareness in language modeling is very limited. We discuss related work in phonemic awareness in NLP and other approaches to mitigating the performance divide of languages from different written scripts, motivating the need for our in-depth exploration of phonemic integration with LLMs.

Leveraging Phonemes in Text NLP. Training text-based neural networks on both phonemic and orthographic information has given downstream task performance improvements in same- and cross-language NLP and speech tasks (Chen et al., 2014; Bharadwaj et al., 2016; Liu et al., 2024). However, such work was mostly limited to smaller LMs or task-specific models (<1B parameters) where parameter constraints remain major potential obstacles for effective integration (Wang et al., 2020). On the other hand, as LLMs scale up to many billions of parameters, training-based schemes face limited orthographic-phonemic data and high computational costs; hence, motivating our work on phonemes integration with LLMs via prompting and enhanced ICL.

Performance Gaps in Multilinguality. It was noted during the rise of early LLMs (<1B parameters) such as mBERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020) that performance varied widely across languages (Wu and Dredze, 2020), an observation that has persisted to modern LMs (see references in Section 1). Some other script-related gaps include the greater difficulty of adapting to languages with non-Latin scripts (Muller et al., 2021; Pfeiffer et al., 2021).

Non-phonemic approaches to bridging multilingual gaps via finetuning involved training schemes, translation, and contrastive presentations (Zheng et al., 2021; Kumar et al., 2022; Yang et al., 2022). With regards to script, while existing works explored improving transfer capability (Fujinuma et al., 2022; Nguyen et al., 2023c; Liu et al., 2024), they also remained limited to small LMs (<2B parameters). Bridging the gap in prompting and ICL performance gaps involved cross-lingual chain of thought and demonstrations via translation (Qin et al., 2023; Ranaldi et al., 2024), but have not to date involved phonology.

[2] https://www.britannica.com/topic/phoneme

Alternate Transcriptions in LLM Prompts. Integrating other linguistic knowledge beyond the textual scripts could lead to more robust and generalizable language models (Linzen, 2020). Even in early multilingual models, using orthographic text as input for adaptation to unseen settings has been shown to provide little gains for non-Latin script languages. Muller et al. (2021) proposed using transliteration during finetuning as a scheme to pass from non-Latin to Latin tokens that were at least phonetically similar. More recently, prior works have proposed utilizing romanization as an augmentation scheme for orthographic text inputs (Jaavid et al., 2024); their motivation does not mention phonology but rather script and token overlap with common Latin-script languages. However, romanizations might not exist for all languages, limiting the potential adaptation towards truly low-resource languages (Cenoz and Gorter, 2017). While prompting is considered, retrieval is not. Additionally, Hermjakob et al. (2018) argues that the existence of two or more parallel inconsistent romanization systems for certain languages might hamper the ability of LLMs to pick up the inter-linguistic connections.

In contrast, in this work, we focus on IPA transcriptions, which provide a universally applicable transcription system capturing the variability of sounds across languages (Mortensen et al., 2016; Bharadwaj et al., 2016). Transliteration and romanization which were motivated by gains from passing to Latin script tokens might limit benefits to certain groups of languages. On the other hand, our use of IPA instead relies on less-frequent characters that LLMs have seen primarily in phonological contexts and thus may have learned related representations; beneficial for in-context reasoning with less spurious connotations. Therefore, we use IPA, explicitly motivated by having a *phonemic* transcription, and investigate how its integration in LLM prompting might help improve the downstream task inference performance across non-Latin scripts. Though we conduct preliminary studies on benefits gained from IPA and Romanization (Section 6), we leave more in-depth comparative investigations on these matters for future works.

3 Is the Written Script Sufficient for Multilinguality?

While the multilingual discrepancy performance when comparing non-Latin and Latin languages have been studied for smaller Transformer-based LMs (<1B parameters), these studies may not fully apply to the recent advancements in contemporary LLMs ($\geq$ 7B parameters), thus requiring additional in-depth investigations.

To empirically measure the gap in performance on Latin versus non-Latin scripts as a baseline for our work, we conduct an extensive and explicit (i.e., not post-hoc) study across 4 non-Latin script languages - Hindi (hin), Arabic (arb), Chinese (zho), Japanese (jpn)[3] - and 6 Latin script languages - German (deu), French (fra), Dutch (nld), Italian (ita), Portuguese (por), Spanish (spa). A comprehensive set of tasks, grouped by family, as presented in Table 1, is evaluated, ranging from natural language understanding (NLU), natural language generation (NLG), machine translation (MT), and question-answering (QA). It is crucial to note that for each task, the nearly similar set of aforementioned languages are available for evaluation, enabling the fair comparisons across tasks and language categories. For this initial suite of evaluations, we evaluate 4 different LLMs at the frontier of their weight class: Mistral-7B (Jiang et al., 2023), Llama3-8B (Dubey et al., 2024), Gemma-7B (Team et al., 2024a) and Qwen2-7B (Yang et al., 2024). Additional details regarding the coverage of tasks, evaluation metrics, language, LLMs, and experimental setup are provided in Appendix A.1.

Task	Metrics	Dataset Name
NLU	Accuracy	PAWS-X (Yang et al., 2019) XNLI (Conneau et al., 2018)
NLG	BLEU (Papineni et al., 2002)	AYA-WIKI (Merity et al., 2022) AYA-CNN (See et al., 2017; Singh et al., 2024)
MT	chrF (Popović, 2015)	FLORES (Team et al., 2024b)
QA	F1	AYA-MLQA (Lewis et al., 2020; Singh et al., 2024)
	Accuracy	Okapi-MMLU (Hendrycks et al., 2021; Lai et al., 2023b) Okapi-HellaSwag (Zellers et al., 2019; Lai et al., 2023b) Okapi-ARC (Clark et al., 2018; Lai et al., 2023b)

Table 1: Task coverage and evaluation metrics for our baselines, including Natural Language Understanding (NLU), Natural Language Generation (NLG), Machine Translation (MT) and Question Answering (QA) tasks.

We present results in Figure 2, finding that the performance of non-Latin script languages remains inferior to Latin script languages across families in this weight class. More importantly, while the performance gap is minor on NLU tasks such as PAWS-X and XNLI across the majority of LLMs,

[3]Following Singh et al. (2024), we adopt the ISO-639-3 language code abbreviation for conciseness.

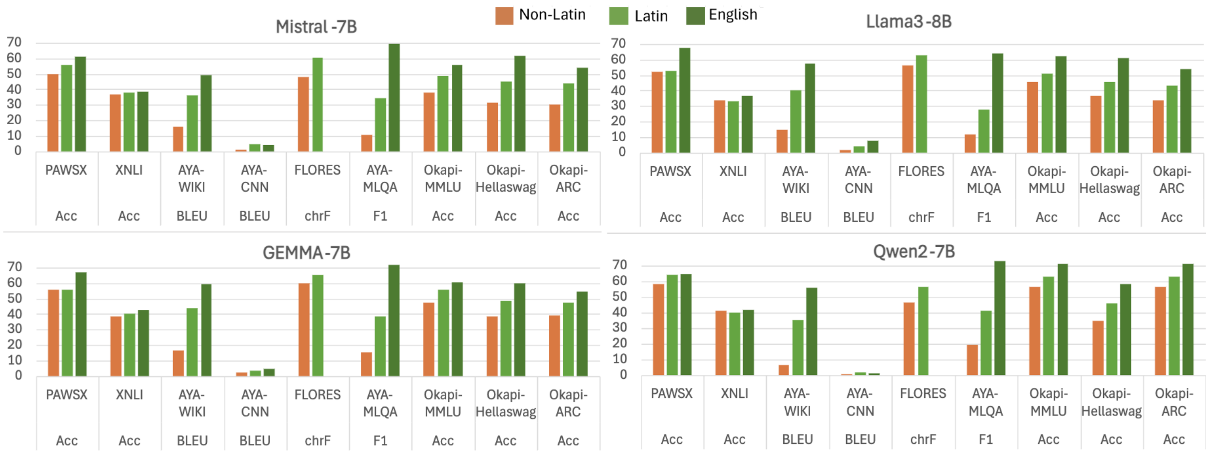

Figure 2: Performance across LLMs, grouped by languages with non-Latin scripts, Latin scripts, and English.

the performance of non-Latin script languages on NLG tasks such as AYA-WIKI is significantly worse compared to performance on Latin script languages, especially English. For instance, the performance gaps between non-Latin and Latin languages are approximately 27.24, 23.08, 5.36 points on AYA-WIKI, AYA-MLQA and FLORES with Gemma-7B. With Qwen2-7B, the gap on AYA-WIKI becomes even larger (approximately 28.59 points).We also observe the similar performance gap on AYA-CNN task. However, as compared to AYA-WIKI, the performance across LLMs is significantly worse as the dataset contains more noise and long-context samples quickly going beyond the evaluated LLMs' context windows under few-shot settings. Overall, these findings highlight that the non-Latin vs Latin language performance gap persists across LLM families and to this day.

To reduce this performance gap, we suggest incorporating phonemic information in model prompting, building on the insights from Ziegler and Goswami (2005). Their "psycholinguistic grain size theory" explains that learning to read depends on phonological awareness (the ability to recognize and work with sounds in language), and the complexity of a language's writing system determines how we process it—whether by focusing on letters, syllables, or whole words. Based on this, we believe that adding phonemic information can help LLMs better process non-Latin script languages, just as phonological awareness helps language learning for humans.

4 Prompting with Phonemes

Motivated by insights from our pilot study (Section 3), we now conduct experiments on representative task categories where different LLM families struggle to achieve similar performance between Latin and non-Latin languages, including NLG (AYA-WIKI), MT (FLORES), and question answering (AYA-MLQA[4]) and compare on the same metrics. Due to the lack of publicly available multilingual corpora with orthographic-phonemic alignments, we adopt the approach of Bharadwaj et al. (2016); Nguyen et al. (2023c) to construct our own aligned dataset for evaluation. Specifically, we use Epitran (Mortensen et al., 2018), a linguistically-crafted tool, to generate IPA transcriptions for the orthographic-only multilingual datasets, setting up the foundation for our phonemic integration explorations with LLMs.

In this work, we explore a series of experiments on phonemic integration with text-based LLMs and aim at improving LLMs' inference performance without the need for pretraining or fine-tuning. In our study, we focus on covering two contemporary mid-sized multilingual LLMs, including Llama3-8B-Instruct (Dubey et al., 2024) and Qwen2-7B-Instruct (Yang et al., 2024), chosen for their strong performance in our pilot study (Figure 2). Additional studies on the other two LLM variants are provided in Appendix B.1. More specifically, we explore two prompting approaches: direct prompting (Section 4.1) and ICL retrieval-based prompting (Section 4.2). Each approach is systematically applied to assess how well these strategies integrate phonemic information into the models and enhance LLMs performance. We then present results on Latin and non-Latin languages, showing

[4]AYA-MLQA is distinguished from the original MLQA (*ORI_MLQA*). Further discussions on the differences are provided in Appendix D.

Llama3-8B-Instruct	AYA-WIKI - BLEU (↑)					FLORES - chrF (↑)					AYA-MLQA - F1 (↑)				
	0-shot	Random	BM25			0-shot	Random	BM25			0-shot	Random	BM25		
			Script	IPA	Mixed			Script	IPA	Mixed			Script	IPA	Mixed
HIN	3.90	37.93	39.55	40.56	40.42	52.90	58.46	58.78	58.98	58.75	35.67	47.48	47.85	47.82	49.30
ARB	4.78	26.02	28.21	28.01	27.96	48.15	59.26	59.00	59.88	60.04	19.41	32.34	31.93	30.30	30.49
ZHO	1.50	3.79	6.52	6.51	6.84	48.71	56.10	56.01	56.00	56.32	11.17	12.11	13.38	13.70	14.38
JPN	1.26	1.26	4.27	3.89	4.18	48.90	53.67	54.22	53.89	54.66	17.10	21.33	24.34	23.09	28.82
Average	2.86	17.25	19.64	19.74	**19.85**	49.67	56.87	57.00	57.19	**57.44**	20.84	28.32	29.38	28.73	**30.75**

Qwen2-7B-Instruct	AYA-WIKI - BLEU (↑)					FLORES - chrF (↑)					AYA-MLQA - F1 (↑)				
	0-shot	Random	BM25			0-shot	Random	BM25			0-shot	Random	BM25		
			Script	IPA	Mixed			Script	IPA	Mixed			Script	IPA	Mixed
HIN	20.74	32.05	34.67	34.07	34.79	57.56	56.99	57.39	57.05	57.60	28.04	46.46	47.16	46.48	46.79
ARB	9.59	11.34	11.64	11.60	13.58	61.04	61.45	61.50	61.77	62.07	18.09	33.11	34.43	33.42	34.75
ZHO	2.26	1.41	1.83	2.24	1.91	58.31	58.53	58.01	58.42	58.91	7.49	8.30	9.91	8.86	10.03
JPN	2.05	2.13	2.53	2.43	2.84	55.80	55.94	56.07	55.87	56.93	14.12	22.35	22.33	22.28	22.57
Average	8.66	11.73	12.67	12.59	**13.28**	58.18	58.23	58.24	58.28	**58.88**	16.93	27.55	28.46	27.76	**28.54**

Table 2: LlaMa-3-8B and Qwen2-7B Instruct ICL results using the BM25 retrieval method using Script vs IPA vs Mixed strategy for retrieval - our proposed Mixed retrieval strategy outperforms all other methods across all tasks.

that phonemic integration indeed helps reduce the performance gap between them.

4.1 Phonemic Integration via Direct Prompting

Task	Prompt Variation	Llama3-8B-Instruct		Qwen2-7B-Instruct	
		0-shot	3-shot	0-shot	3-shot
AYA-WIKI (BLEU)	Script	2.45	17.25	4.73	8.3
	Script + IPA	**2.86**	**17.76**	**8.66**	**11.73**
FLORES (chrF)	Script	38.17	**57.06**	57.96	**58.45**
	Script + IPA	**49.67**	56.87	**58.18**	58.23
AYA-MLQA (F1)	Script	17.31	20.84	15.37	27.32
	Script + IPA	**20.84**	**28.32**	**16.93**	**27.55**

Table 3: Effect of prompting with orthographic and/or phonemic information on Llama3-8B-Instruct and Qwen2-7B-Instruct models.

We first study the most straightforward approach to inject phonemic information: direct prompting. More specifically, under the assumption that LLMs might be able to surface internal knowledge about correspondences between script and phonemes effectively, we append the phonemic transcription in the prompt as additional auxiliary information for the textual script inputs. We conduct experiments with 0-shot and 3-shot prompting with random sampling. Additional details of prompt templates and variations are provided in Appendix A.2.

As shown in Table 3, concatenating IPA information with orthographic inputs consistently improves performance across various tasks and evaluation settings, except in the 3-shot setting for FLORES. Given that 0-shot performance on FLORES is already high with standard prompting, we hypothesize that text-based LLMs can effectively handle FLORES without much benefit from adding IPA information through in-context examples.

4.2 Phonemic Integration via ICL Retrieval

Besides directly injecting IPA information through prompting, we explore the use of phonemic IPA information to enhance ICL retrieval. Since LLMs are highly sensitive to various prompt formats (Lu et al., 2024), based on our observations in 4.1, we maintain a consistent Script+IPA prompt format for our experiments, while leveraging different ICL retrieval strategies in a fixed 3-shot setting.

Additionally, we compare the impact of few-shot example retrievals based on phonemic and orthographic similarity, against a baseline of randomly retrieved examples (`Random`). The retrieval methods include orthographic-based matching (`Script-ICL`), IPA-based matching (`IPA-ICL`), and our proposed mixed strategy (`Mixed-ICL`). In the `Mixed-ICL` approach, matching scores are calculated separately with `Script` and `IPA`, then averaged for each sample. The top 3 samples are selected after re-ranking by the averaged scores, leveraging both orthographic and phonemic similarity information. Additional in-depth comparisons with other mixing strategies are further explored

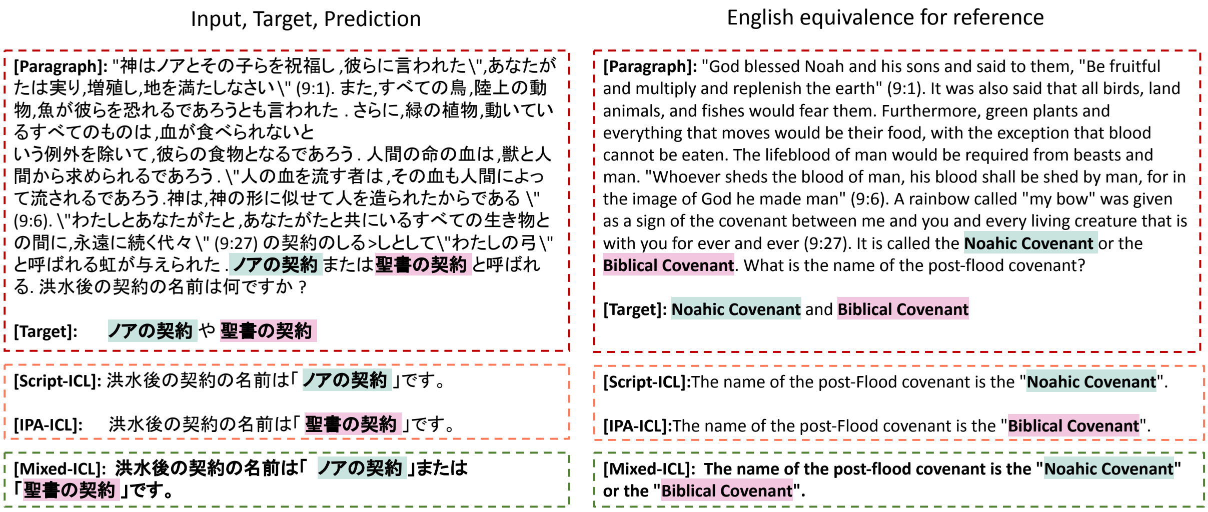

Figure 3: Example for JPN on MLQA task by probing the generated output from LLMs when prompted with different ICL retrieval methods (`Script-ICL` vs `IPA-ICL`vs `Mixed-ICL`). `Mixed-ICL` yields more comprehensive output and semantically closer to the target than `IPA-ICL` and `Script-ICL` counterparts.

in Appendix B.4. Due to a lack of phonemic transcription embedders, we focus our studies on using lexical retrieval (namely BM25 sparse retrieval; Trotman et al., 2014; Luo et al., 2023) to compute matching scores across all of our ICL variants unless stated otherwise. Similar studies on a dense retriever variant (Appendix B.3) further validate the effectiveness and flexibility of our method.

We find that the ICL retrieval methods generate strong improvements over the Random baselines across both mid-sized LLMs Llama3-8B-Instruct and Qwen2-7B-Instruct (Table 2). In Llama3-8B-Instruct, we observe average absolute performance improvements of 1.06 and 0.41 points on AYA-MLQA with `Script-ICL` and `IPA-ICL`, respectively. The gain is further boosted to 2.43 points with our proposed `Mixed-ICL` strategy by combining benefits from both `Script` and `IPA`. We further analyze our observations and findings with fine-grained analysis on aspects such as the impact of Latin vs non-Latin scripts, `Script-ICL` vs `IPA-ICL` retrieval.

4.3 Reducing the Performance Gap between Latin versus non-Latin Languages

We observe that the inclusion of IPA information in ICL leads to improved performance for both Latin and non-Latin script languages, and especially contributes to better performance for non-Latin script languages (12.57%, 1% and 8.58% relative performance gain for AYA-WIKI, FLORES, and AYA-MLQA respectively), helping reduce the previously observed performance gap (Table 4).

In particular, we find that the `Mixed-ICL` strategy contributes to the most gains across all tasks, with the exception of FLORES on Latin languages, where the `IPA-ICL` strategy achieves stronger performance gain; refer to Appendix C for absolute performance details of Latin language. This reinforces our main finding: integrating phonemic information in ICL prompting leads to improved performance for not just non-Latin languages, but also Latin languages, with non-Latin languages seeing higher gains. This study also reiterates the essentials of leveraging IPA as phonemic representation for universal language support.

AYA-WIKI	`Script-ICL`	`IPA-ICL`	`Mixed-ICL`
Latin	+12.07%	+12.08%	**+12.57%**
non-Latin	+13.84%	+14.45%	**+15.07%**
FLORES	`Script-ICL`	`IPA-ICL`	`Mixed-ICL`
Latin	+0.33%	**+0.51%**	+0.37%
non-Latin	+0.23%	+0.55%	**+1.00%**
AYA-MLQA	`Script-ICL`	`IPA-ICL`	`Mixed-ICL`
Latin	+0.60%	+4.09%	**+4.58%**
non-Latin	+3.74%	+1.45%	**+8.58%**

Table 4: Relative performance improvements using different ICL retrieval methods when compared to `Random` ICL retrieval baseline with Llama3-8B-Instruct.

[INPUT]
<ENG eq>: At the age of 19, Adichie left Nigeria for the United States to study communications and political science at Drexel University in Philadelphia.
<Script>: 在19岁时, 阿迪奇离开尼日利亚前往美国, 在费城的德雷克塞尔大学学习通信与政治学.
<IPA>: t͡sai19sueiʂɻ, etit͡ɕʰilikʰainiɻilijat͡ɕʰianwaŋmeikuo, t͡saifeiʈ͡ʂʰeŋtiteleikʰeseɻtaɕyeɕyeɕitʰoŋɕinjuʈ͡ʂeŋʈ͡ʂɻɕye.

[Top IPA-ICL]
<ENG eq>: Earth's artificial intelligence system (similar to the Internet) warns church authorities and scientific envoys of the Christian-Islamic Church about a divine "invasion" and the preparation of countermeasures.
<Script>: 地球的人工智能系 统 (类似于互联网) 警告基督教 - 伊斯兰教教会的教会权威和科学使节关于神圣的"入侵"和反制措施的准 备.
<IPA>: tit͡ɕʰioutiɻenkoŋʈ͡ʂɻneŋɕitʰoŋ (leisijuxulianwaŋ) t͡ɕiŋkaot͡ɕitut͡ɕiao – isilant͡ɕiaot͡ɕiaoxueitit͡ɕiaoxueit͡ɕʰyanweixekʰeɕyeʂɻt͡ɕiekuanjuʂenʂeŋti "ɻut͡ɕʰin" xefanʈ͡ʂɻt͡sʰuoʂɻtiʈ͡ʂunpei.

[Top Script-ICL]
<EN eq>: A Jewish American born in Philadelphia, Pennsylvania, Isaiah Edwin Leopold.
<Script>: 一个犹太裔美国人出生在费城,宾夕法尼亚州的艾萨亚·埃德温·利奥波德.
<IPA>: ikejoutʰaiimeikuoɻenʈ͡ʂʰuʂeŋt͡saifeiʈ͡ʂʰeŋ,pinɕifanijaʈ͡ʂoutiisaja·aitewen·liaopuote.

[Top Mixed-ICL]
<ENG eq>: A 2003 graduate of Texas A&M University in College Station, Landgraf received his Juris Degree in 2008 from Saint Mary's University School of Law in San Antonio.
<Script>: 2003年毕业于德克萨斯A&M大学在大学站, 兰德格拉夫获得了他的法学博士学位在2008年圣玛丽大学法学院在圣安东尼奥.
<IPA>: 2003nianpijejutekʰesasia&Mtaɕyet͡saitaɕyeʈ͡ʂan, lantekelafuxuoteliaotʰatifaɕyepuoʂɻɕyeweit͡sai2008nianʂeŋmall itaɕyefaɕyejuant͡saiʂeŋantoŋniao.

Figure 4: Example of top-1-retrieved samples from different ICL retrievers for the [INPUT] query sample. <ENG eq> denotes the equivalent translation of the target language input in English for readability. <Script> and <IPA> denote the orthographic and corresponding IPA representation of the given sample. The highlights represent the aligned concepts captured across examples, including *university*, *location*, and *governance* respectively (best viewed in color).

5 How do `Script` vs `IPA` vs `Mixed` Work for ICL?

To gain a better intuitive understanding of different ICL retrievers, we conduct additional qualitative studies by (1) probing the generated output differences when promping with different ICL retriever methods, and (2) investigating the top retrieved examples from different retrievers. Observing the LLM generations when prompted with different ICL approaches (Figure 3), the `Mixed-ICL` strategy shows a more comprehensive answer than its `Script-ICL` and `IPA-ICL` counterparts, resulting in the answer closest to the ground truth target among the three compared.

Additionally, when comparing the top retrieved examples from different ICL approaches (Figure 4), we observe the top retrieved example from `Mixed-ICL` covers the most similar concepts with the original input query (i.e. *Drexel University* vs *Texas A&M University* (*university*), *Philadelphia* vs *San Antonio* (*location*), *political science* vs *law* (*governance*)). This showcases its closer semantic connections with the input query sample as compared to the other two variants, where fewer similar concepts were captured. These observations support our motivations in synthesizing the knowledge from both Script and IPA information via ICL, helping explain the performance improvements observed in Table 2.

6 Analysis

For consistency and conciseness, our analysis results are reported as averages over the aforementioned non-Latin languages for the targeted tasks unless stated otherwise.

How diverse are ICL examples retrieved with Script vs IPA? We present Table 5 to measure the overlap between retrieved samples from `Script-ICL` and `IPA-ICL`. Ranging from 7.82% to 13.45% on average, we find that leveraging phonemic information provides distinct ICL retrieval examples from when orthographic information is considered. Keeping performance in mind (Table 2), this observation points to the fact that retrieval with IPA information is a robust and effective tool.

Task	HIN	ARB	ZHO	JPN	Avg
AYA-WIKI	11.13%	8.33%	6.73%	5.07%	**7.82%**
FLORES	11.07%	8.73%	6.80%	5.60%	**8.05%**
AYA-MLQA	15.00%	8.93%	23.40%	6.47%	**13.45%**

Table 5: Percentage of overlapping retrieved ICL examples from `Script-ICL` and `IPA-ICL` Retrieval methods (`Script-ICL` ∩ `IPA-ICL`) for evaluated tasks on Llama3-8B-Instruct model.

Are both the script and IPA inputs needed for ICL prompting? Since `IPA-ICL` and `Script-ICL` only consider the corresponding relevant information for selecting ICL examples, (i.e., IPA and Script respectively), the remaining information might be deemed unimportant and provide little information for LLMs' inference. For instance, if using `Script-ICL`, it is possible that only Script information is essential while the IPA information can be left out. Therefore, we conduct additional studies on the impact of removing this possibly unused information. As observed in Table 6, including both Script and IPA information yields the best performance consistently across different tasks. This shows the importance of providing comprehensive information regarding both ICL and query samples for LLMs' predictions regardless of the types of information used for ICL retrieval. In addition, we also observe the significant performance decrease when Script information is removed from the prompt, aligning with the claims from previous works that phonemic information is considered more as an addendum, rather than a replacement, to textual scripts for enhancing downstream task performance for text-based LMs (Nguyen et al., 2024).

Task	Metric	`Script-ICL`		`IPA-ICL`	
		w/o IPA	w/ IPA	w/o Script	w/ Script
AYA-WIKI	BLEU	17.42	**19.64**	12.90	**19.74**
FLORES	chrF	**58.12**	57.00	21.13	**57.19**
AYA-MLQA	F1	28.59	**29.38**	3.70	**28.73**

Table 6: Effect of removing unused information from ICL when prompting Llama3-8B-Instruct; "w/o" means masking information with empty spaces, keeping prompt the same (`<script input>: "XYZ", <ipa input>: ""`).

Does `IPA-ICL` retrieval vary with different tokenizers? Since text-based LLM tokenizers might not have been trained with processing IPA inputs, we investigate the potential impact of different tokenizers on `IPA-ICL`. As observed in Table 7, most tokenizers show benefits of `IPA-ICL` over Random baseline. However, we observe that naïve CS and WS tokenizers typically perform worse than tokenizers from pre-trained LLMs (14.44% vs 10.75% relative performance gain on AYA-WIKI with Llama3-8B-Instruct and WS tokenizers respectively). On AYA-MLQA, WS and CS tokenizers even hurt task performance, resulting in the relative performance decrease of 0.74% and 3.35% when compared to the Baseline.

IPA versus Romanization As previously mentioned, romanization is also considered a viable phonemic signal. Despite appealing characteristic of leveraging roman characters to transliterate and/or capture pronunciation for the target languages, romanization is not standardized and heavily language-specific, resulting in various potential romanization schemes for given languages. More importantly, since romanization is only available in non-Latin languages, it cannot be utilized to capture phonemic information for Latin languages, leading to the inability to enhance LLMs multilingual capability for Latin languages.

Tokenizer type	AYA-WIKI	FLORES	AYA-MLQA
Baseline	17.25	56.87	28.32
WS	19.47 (↑12.83%)	56.58 (↓ 0.51%)	28.11 (↓0.74%)
CS	19.11 (↑10.75%)	56.78 (↓ 0.16%)	27.37 (↓3.35%)
Dense	19.19 (↑11.22%)	56.95 (↑ 0.13%)	28.62 (↑1.06%)
Qwen2-Inst	19.64 (↑13.82%)	57.12 (↑ 0.44%)	**29.32 (↑3.53%)**
Llama3-Inst	**19.74 (↑14.44%)**	**57.19 (↑0.55%)**	28.73 (↑1.45%)

Table 7: Impact of different tokenizations on `IPA-ICL` performance. WS denotes whitespace separator originally proposed by BM25 algorithms (Trotman et al., 2014) with English focus. CS denotes the tokenization in which each character (excluding white space) is treated as a single token. Baseline denotes the Random sampling approach for ICL. Average performance across all non-Latin languages is reported.

For completeness, we conduct additional investigations on prompting LLMs with romanization, as similarly done with IPA throughout our work. Following Jaavid et al. (2024) and Nguyen et al. (2024), we leverage the language-specific romanization tools to preprocess the script-romanization aligned datasets. Similar to Section 4, we explore two different directions (1) Direct prompting with romanization and (2) Enhanced Roman-ICL instead of IPA-ICL.

Task	Variation	Llama3-8B-Instruct		Qwen2-7B-Instruct	
		0-shot	3-shot	0-shot	3-shot
AYA-WIKI (BLEU)	Script	2.45	17.25	4.73	8.3
	Script + IPA	**2.86**	**17.76**	**8.66**	11.73
	Script + Roman	1.64	17.41	5.82	**13.30**
FLORES (chrF)	Script	38.17	57.06	57.96	**58.45**
	Script + IPA	49.67	56.87	**58.18**	58.23
	Script + Roman	**51.26**	**57.76**	57.23	58.41
AYA-MLQA (F1)	Script	17.31	20.84	15.37	27.32
	Script + IPA	20.84	28.32	**16.93**	**27.55**
	Script + Roman	**20.95**	**29.79**	13.20	27.00

Table 8: Effect of prompting with both IPA and Romanization as phonemic information for Llama3-8B-Instruct and Qwen2-7B-Instruct models.

Direct Prompting. As indicated in Table 8, similar to Script+IPA, Script+Roman gains improvements over the Script-only prompting approach in most cases. However, it is unclear whether IPA or romanization serves as a better phonemic signal for this approach. For instance, on AYA-WIKI task, Script+IPA tends to perform better than Script+Roman, but the opposite might exist on FLORES task. Hence, we further study ICL based approaches with Romanization as well.

ICL retrieval benefits with IPA and Romanization. We evaluate the impact of IPA-ICL retriever and Roman-ICL retriever independently from the impact of the prompting variation, we conduct evaluations on two scenarios: (1) Only Romanization is used within the prompt as phonemic signal and (2) Only IPA is used within the prompt. Table 9 reveals that our IPA Retriever is consistently more effective than the Romanization-based counterpart across all of the evaluated tasks regardless of phonemic signals used for prompting. Such a gap can be significantly obvious as 2.33 and 1.39 BLEU points on AYA-WIKI when prompting with only Romanization and only IPA respectively. With this observation, we focus our main studies on phonemic integrations using IPA as phonemic signals. We leave further in-depth comparative studies between Romanization and IPA for future works.

	Roman. Prompt Only		IPA Prompt Only	
	Roman. Retriever	IPA Retriever	Roman Retriever	IPA Retriever
AYA-WIKI	16.95	**19.28**	18.35	**19.74**
FLORES	57.79	**58.21**	56.92	**57.44**
AYA-MLQA	30.49	**30.69**	28.46	**28.73**

Table 9: Effect of romanization-ICL Retriever and IPA-ICL Retriever when different prompting is adopted. BM25 Retriever and Llama3-8B-Instruct are leveraged for consistency.

Compound benefits of `IPA-ICL` *and* `Roman-ICL`. As both romanization and IPA can be essential complementary information beyond the written scripts, we conduct additional investigations on whether aggregating both information as a comprehensive enhanced ICL approach can further facilitate the multilingual capability of LLMs. Concisely, we evaluate the `All-ICL` variant, an ICL approach leveraging **all** information including Script, IPA, and Romanization for ICL retrieval with the aforementioned `Mixed-ICL` aggregation mechanism. As observed in Table 10, our empirical studies demonstrate the complementary benefits of leveraging different additional sources of information for enhanced ICL, leading to the most significant performance improvements across all evaluated downstream tasks.

Task	Baseline	`Script-ICL`	`IPA-ICL`	`Roman-ICL`	**`All-ICL`**
AYA-WIKI	16.90	19.15	19.40	19.17	**19.67**
FLORES	56.23	56.68	56.85	56.56	**57.03**
AYA-MLQA	28.32	29.38	28.73	28.46	**30.84**

Table 10: Benefits of different enhanced ICL approaches for the evaluated tasks on Llama3-8B-Instruct model. `Baseline` denotes the Random-sampling ICL approach. `All-ICL` denotes the combination of `Script-ICL`, `IPA-ICL` and `Roman-ICL` via the introduced `Mixed-ICL` aggregation mechanisms.

Overall, our analyses confirm our findings: IPA information is an effective tool, even in different settings, towards better inference-time performance—especially for non-Latin languages.

7 Conclusion

We investigate the effect of integrating phonemic information towards improving the multilingual abilities of large-scale language models at inference time. Our pilot study demonstrates that the performance gap between Latin and non-Latin script languages remains high even on latest state-of-the-art LLMs (with an approximate average 23% difference in performance) for certain tasks, especially on generation tasks (approx. 65%).

Motivated by these observations, we propose introducing phonemic integration in prompting with LLMs. We explore two incorporation mechanisms: direct prompting and ICL retrieval. While we observe performance gains with direct prompting, our empirical study demonstrates that the ICL retrieval provides an even more effective way to improve downstream task performance. The `Mixed-ICL` retrieval strategy captures diverse and similar ICL examples beyond the textual scripts, leading to the best overall performance across multiple tasks, also evidenced in the case studies we present.

We contribute an extensive empirical study on the effect of integrating phonemic information towards improving the performance of large-scale LLMs, reducing the performance gap between Latin and non-Latin performance. We show that incorporating phonemic information as IPA with few-shot ICL retrieval prompting is an effective method to improve multilingual performance for languages with differing written scripts.

Limitations

We conduct multiple studies and analyses towards providing a comprehensive report on how phonemic integration with orthographic prompting can improve performance for non-Latin script languages, especially towards reducing the performance gap between their Latin counterparts. However, our study has its limitations.

First, we rely on the external resources for both multilingual evaluation datasets and IPA generation tools to generate the phonemic text input, and thus rely on their quality. To the best of our knowledge, we utilize the best, linguistically informed, IPA generation tool that has been widely adopted by previous works for preprocessing IPA transcriptions (Bharadwaj et al., 2016; Nguyen et al., 2023c). Regarding evaluation datasets, with the goal of future extensions towards more languages, we leverage the most comprehensive multilingual datasets available, including Aya Collections (Singh et al., 2024) and Okapi (Lai et al., 2023b). We heavily rely on their data quality control protocols for our target languages. However, since the datasets leverage different machine translation tools to generate corpora across a large number of supported languages, the data quality for our targeted non-Latin languages might not be optimal. Despite our attempts in data cleaning as mentioned in Appendix A.2, we have not performed a thorough and systematic quality evaluation of our extracted dataset from the contemporary benchmark datasets.

Second, our work is restricted to phonemic integration via prompting. Unlike previous works that explore Instruction Fine-tuning and Continual Pre-training concurrently (Jaavid et al., 2024), we seek to provide in-depth insight into the effect of phonemic integration with text-based-LLMs on downstream task performance by isolating the additional parameter updates and training objectives of fine-tuning and pretraining paradigms. The study serves as a foundational guideline for future fine-tuning approaches for better alignment between phonemic and orthographic signals.

Third, we explore a simple preliminary `Mixed-ICL` strategy to aggregate the benefits from both IPA and Scripts. The promising results not only provide insights into the early exploration of phonemic integration with text-based LLMs but also encourage future works on investigating more effective and dynamic aggregation mechanisms (Ye and Ling, 2019; Nguyen et al., 2020, 2023b) to enhance the benefits further.

Lastly, our study is limited to the 4 non-Latin and 6 Latin script languages. However, we ensure that each of the non-Latin languages chosen for the study has different written scripts, making them diverse and our task of evaluation complex as presented in Table 11. Extending the study to more languages covering more diverse linguistic groups, writing scripts, language families (Nguyen and Rohrbaugh, 2019; Singh et al., 2024) remains a future direction for our research.

Ethical Considerations and Societal Impact

Our study is mainly empirical in nature, and does not involve studies with humans. We also utilize publicly available datasets, which are reportedly non-toxic. We plan to make our unique IPA-orthographic aligned data available publicly , and hope that along with our study, it motivates further exploration and research into the potential of including prompting with phonemic information towards better performance.

We believe our study could have a positive impact by reducing the performance gap of non-Latin script languages compared to Latin script languages. We hope our work will encourage the exploration of new methods in (and beyond) phonemic integration, towards further reducing the performance gap and improving access for all.

Our work only generates the IPA transcription from the publicly available multilingual benchmark datasets. In other words, we do not generate any new contents besides the transcriptions of the given textual scripts; hence, our work and its generated data do not pose any potential risks.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual evaluation of generative AI. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267, Singapore. Association for Computational Linguistics.

C. Anderson. 2018. *Essentials of Linguistics, 2nd Edition*. Open textbook library. eCampusOntario.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–718, Nusa Dua, Bali. Association for Computational Linguistics.

Akash Bharadwaj, David Mortensen, Chris Dyer, and Jaime Carbonell. 2016. Phonologically aware neural model for named entity recognition in low resource transfer settings. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1462–1472, Austin, Texas. Association for Computational Linguistics.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, pages 1877–1901.

Jasone Cenoz and Durk Gorter. 2017. Minority languages and sustainable translanguaging: threat or opportunity? *Journal of Multilingual and Multicultural Development*, 38(10):901–912.

Dongpeng Chen, Brian Mak, Cheung-Chi Leung, and Sunil Sivadas. 2014. Joint acoustic modeling of triphones and trigraphemes by multi-task learning deep neural networks for low-resource speech recognition. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5592–5596. IEEE.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *ArXiv*, abs/1803.05457.

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

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.

Chunyuan Deng, Yilun Zhao, Xiangru Tang, Mark Gerstein, and Arman Cohan. 2024. Investigating data contamination in modern benchmarks for 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)*, pages 8698–8711.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Aydin Y Durgunoĝlu, William E Nagy, and Barbara J Hancin-Bhatt. 1993. Cross-language transfer of phonological awareness. *Journal of Educational Psychology*, 85(3):453–465.

Yoshinari Fujinuma, Jordan Boyd-Graber, and Katharina Kann. 2022. Match the script, adapt if multilingual: Analyzing the effect of multilingual pretraining on cross-lingual transferability. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1500–1512, Dublin, Ireland. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Ulf Hermjakob, Jonathan May, and Kevin Knight. 2018. Out-of-the-box universal Romanization tool uroman. In *Proceedings of ACL 2018, System Demonstrations*, pages 13–18, Melbourne, Australia. Association for Computational Linguistics.

J Jaavid, Raj Dabre, M Aswanth, Jay Gala, Thanmay Jayakumar, Ratish Puduppully, and Anoop Kunchukuttan. 2024. Romansetu: Efficiently unlocking multilingual capabilities of large language models via romanization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15593–15615.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Gokul Karthik Kumar, Abhishek Gehlot, Sahal Shaji Mullappilly, and Karthik Nandakumar. 2022. Mucot: Multilingual contrastive training for question-answering in low-resource languages. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, pages 15–24.

Viet Lai, Nghia Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Nguyen. 2023a. Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13171–13189.

Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023b. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 318–327.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020. Mlqa: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330.

Changmao Li and Jeffrey Flanigan. 2024. Task contamination: Language models may not be few-shot anymore. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):18471–18480.

Tal Linzen. 2020. How can we accelerate progress towards human-like linguistic generalization? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5210–5217.

Yihong Liu, Chunlan Ma, Haotian Ye, and Hinrich Schuetze. 2024. TransliCo: A contrastive learning framework to address the script barrier in multilingual pretrained language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2476–2499, Bangkok, Thailand. Association for Computational Linguistics.

Sheng Lu, Hendrik Schuff, and Iryna Gurevych. 2024. How are prompts different in terms of sensitivity? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5833–5856.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Man Luo, Xin Xu, Zhuyun Dai, Panupong Pasupat, Seyed Mehran Kazemi, Chitta Baral, Vaiva Imbrasaite, and Vincent Y. Zhao. 2023. Dr.icl: Demonstration-retrieved in-context learning. *CoRR*, abs/2305.14128.

Rishabh Maheshwary, Vikas Yadav, Hoang Nguyen, Khyati Mahajan, and Sathwik Tejaswi Madhusudhan. 2024. M2lingual: Enhancing multilingual, multi-turn instruction alignment in large language models. *arXiv preprint arXiv:2406.16783*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2022. Pointer sentinel mixture models. In *International Conference on Learning Representations*.

David R. Mortensen, Siddharth Dalmia, and Patrick Littell. 2018. Epitran: Precision G2P for many languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).

David R. Mortensen, Patrick Littell, Akash Bharadwaj, Kartik Goyal, Chris Dyer, and Lori Levin. 2016. PanPhon: A resource for mapping IPA segments to articulatory feature vectors. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3475–3484, Osaka, Japan. The COLING 2016 Organizing Committee.

Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. When being unseen from mbert is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462.

Hoang Nguyen, Ye Liu, Chenwei Zhang, Tao Zhang, and S Yu Philip. 2023a. Cof-cot: Enhancing large language models with coarse-to-fine chain-of-thought prompting for multi-domain nlu tasks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12109–12119.

Hoang Nguyen and Gene Rohrbaugh. 2019. Cross-lingual genre classification using linguistic groupings. *Journal of Computing Sciences in Colleges*, 34(3):91–96.

Hoang Nguyen, Chenwei Zhang, Ye Liu, Natalie Parde, Eugene Rohrbaugh, and S Yu Philip. 2024. Cori: Cjkv benchmark with romanization integration-a step towards cross-lingual transfer beyond textual scripts. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4008–4020.

Hoang Nguyen, Chenwei Zhang, Ye Liu, and S Yu Philip. 2023b. Slot induction via pre-trained language model probing and multi-level contrastive learning. In *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 470–481.

Hoang Nguyen, Chenwei Zhang, Congying Xia, and S Yu Philip. 2020. Dynamic semantic matching and aggregation network for few-shot intent detection. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1209–1218.

Hoang Nguyen, Chenwei Zhang, Tao Zhang, Eugene Rohrbaugh, and S Yu Philip. 2023c. Enhancing cross-lingual transfer via phonemic transcription integration. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9163–9175.

Marta R. Costa-jussà NLLB Team, James Cross, Onur Çelebi, Kenneth Heafield Maha Elbayad, 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.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2021. Unks everywhere: Adapting multilingual language models to new scripts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10186–10203.

Maja Popović. 2015. chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the tenth workshop on statistical machine translation*, pages 392–395.

Libo Qin, Qiguang Chen, Fuxuan Wei, Shijue Huang, and Wanxiang Che. 2023. Cross-lingual prompting: Improving zero-shot chain-of-thought reasoning across languages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2695–2709.

Leonardo Ranaldi, Giulia Pucci, and Andre Freitas. 2024. Empowering cross-lingual abilities of instruction-tuned large language models by translation-following demonstrations. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 7961–7973, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. 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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Oleh Shliazhko, Alena Fenogenova, Maria Tikhonova, Anastasia Kozlova, Vladislav Mikhailov, and Tatiana Shavrina. 2024. mGPT: Few-shot learners go multilingual. *Transactions of the Association for Computational Linguistics*, 12:58–79.

Shivalika Singh, Freddie Vargus, Daniel D'souza, Börje Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura O'Mahony, Mike Zhang, Ramith Hettiarachchi, Joseph Wilson, Marina Machado, Luisa Moura, Dominik Krzemiński, Hakimeh Fadaei, Irem Ergun, Ifeoma Okoh, Aisha Alaagib, Oshan Mudannayake, Zaid Alyafeai, Vu Chien, Sebastian Ruder, Surya Guthikonda, Emad Alghamdi, Sebastian Gehrmann, Niklas Muennighoff, Max Bartolo, Julia Kreutzer, Ahmet Üstün, Marzieh Fadaee, and Sara Hooker. 2024. Aya dataset: An open-access collection for multilingual instruction tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11521–11567, Bangkok, Thailand. Association for Computational Linguistics.

Llinos H Spencer and J Richard Hanley. 2003. Effects of orthographic transparency on reading and phoneme awareness in children learning to read in wales. *British Journal of Psychology*, 94(1):1–28.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024a. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

NLLB Team et al. 2024b. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841.

Andrew Trotman, Antti Puurula, and Blake Burgess. 2014. Improvements to bm25 and language models examined. In *Proceedings of the 19th Australasian Document Computing Symposium*, pages 58–65.

Zirui Wang, Zachary C Lipton, and Yulia Tsvetkov. 2020. On negative interference in multilingual models: Findings and a meta-learning treatment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4438–4450.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners. In *ICLR*. OpenReview.net.

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.

Congying Xia, Chenwei Zhang, Hoang Nguyen, Jiawei Zhang, and Philip Yu. 2020. Cg-bert: Conditional text generation with bert for generalized few-shot intent detection. *arXiv preprint arXiv:2004.01881*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Huiyun Yang, Huadong Chen, Hao Zhou, and Lei Li. 2022. Enhancing cross-lingual transfer by manifold mixup. *arXiv preprint arXiv:2205.04182*.

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

Zhi-Xiu Ye and Zhen-Hua Ling. 2019. Multi-level matching and aggregation network for few-shot relation classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2872–2881, Florence, Italy. Association for Computational Linguistics.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Bo Zheng, Li Dong, Shaohan Huang, Wenhui Wang, Zewen Chi, Saksham Singhal, Wanxiang Che, Ting Liu, Xia Song, and Furu Wei. 2021. Consistency regularization for cross-lingual fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3403–3417.

Johannes C Ziegler and Usha Goswami. 2005. Reading acquisition, developmental dyslexia, and skilled reading across languages: A psycholinguistic grain size theory. *Psychological Bulletin*, 131(1):3–29.

Barret Zoph, Colin Raffel, Dale Schuurmans, Dani Yogatama, Denny Zhou, Don Metzler, Ed H. Chi, Jason Wei, Jeff Dean, Liam B. Fedus, Maarten Paul Bosma, Oriol Vinyals, Percy Liang, Sebastian Borgeaud, Tatsunori B. Hashimoto, and Yi Tay. 2022. Emergent abilities of large language models. *TMLR*.

A Additional Setup Details

A.1 Pilot Study Details

In this section, we provide details regarding our Pilot Study setup, separated into 3 main sections: (1) Task Coverage, (2) Language Coverage, (3) Experimental Setup

Language Type	Language Name	ISO Language	Writing Script
non-Latin	Arabic	ARB	Arabic
	Hindi	HIN	Devanagari
	Simplified Chinese	ZHO	Hanzi
	Japanese	JPN	Katakana, Hiragana, Kanji
Latin	German	DEU	Latin
	French	FRA	Latin
	Italian	ITA	Latin
	Dutch	NLD	Latin
	Portuguese	POR	Latin
	Spanish	SPA	Latin
Latin	English	ENG	Latin

Table 11: Details of language coverage in the Pilot Study

Language & Task Coverage As mentioned in the 3, the objective of our Pilot Study is to compare and contrast performance of contemporary LLMs ($\geq$7B parameters) on non-Latin and Latin languages. English (ENG) performance is considered the upper bound performance. Details of covered languages together their corresponding written scripts are presented in Table 11.

With the goal of conducting a thorough evaluation of contemporary LLMs, we cover a wide range of tasks where all of the considered non-Latin and Latin languages are avaiable. We separate the tasks into 4 major categories: Natural Language Understanding (NLU), Natural Language Generation (NLG), Machine Translation (MT) and Question-Answering (QA). Following Singh et al. (2024); Jaavid et al. (2024), we leverage standard evaluation metrics for each task type as denoted in Table 1. For NLU, as PAWS-X (Yang et al., 2019) and XNLI (Conneau et al., 2018) datasets have all the Latin and non-Latin languages, we leverage the original one for our pilot study. For FLORES dataset where high-quality translations are available across 200 languages (Team et al., 2024b), we focus on evaluating LLMs' translation capability of LLMs from target language to Engish (target $\rightarrow$ ENG). For Multiple Choice Question Answering (QA), we extract the multilingual version of MMLU, HellaSwag and ARC datasets accumulated from Okapi dataset collection Lai et al. (2023b).

For NLG and Extractive QA tasks, we leverage Aya Collection dataset (Singh et al., 2024) where these tasks are available in 114 languages, including the considered non-Latin and Latin languages. Unlike the original multilingual MLQA (Lewis et al., 2020) datasets, Aya Collections leverage the English (ENG) splits and generate the corresponding equivalence for other languages via NLLB translation tools (NLLB Team et al., 2022). As NLLB might contain degrading performance on certain low-resource languages, including our non-Latin languages, the dataset quality might suffer. Therefore, direct comparison between previous SOTA baselines and our empirical study might be considered inappropriate. Additional empirical studies validating our statement are provided in Section D.

Experimental Setup We conduct our empirical study via the EleutherAI evaluation framework[5] where default settings for each task are leveraged. The results are reported in 3-shot settings so that LLMs can follow the output format from the in-context examples and generate appropriate output responses for the given tasks. Random sampling is leveraged to extract 3-shot examples for each given query sample.

A.2 Details of main experimental setup

Prompt templates by task. In our experiments we observe that slight prompt variability can drastically affect task performance (Lu et al., 2024). Additionally, certain challenging tasks such as AYA-MLQA rely on sufficient contexts and prompt templates to execute the given tasks effectively. Therefore, we outline the specific prompts leveraged in our study in Table 12. Consistent prompting eliminates the influence of prompt variations on the observed differences in our empirical studies. Instead, the changes introduced by our individual study can be directly measured through the performance evaluation. The optimal prompt selection and tuning for the best downstream task performance is beyond the scope of our study.

[5]https://github.com/EleutherAI/lm-evaluation-harness

Task	Prompt Template
AYA-WIKI	You are an expert in Text Simplification task. Given the <script input>, generate a more complex version of the given input. You are also given the IPA phonemic transcription as supplementary information. The format will be as follows: ## Input: {{input}} ## Input (in IPA): {{input(in IPA)}} ## Answer: {{answer}} The Answer must be a complete sentence with correct grammatical structure of the target language. The Answer must be in the target language script. Follow the examples if given.
FLORES	Given the <script input> and IPA <ipa input>, translate the <script input> to English. Return one single answer. Do not provide explanations. The format is as follows: <script input>: {{input}} <ipa input>: {{input_ipa}} <script output>:{{script_output}} The Answer must be in the target language script. Follow the examples if given.
AYA-MLQA	You are an expert in extractive question answering. You will be given Context and a Question (Context + Question) and you must generate at most one Answer, based only on the information in the Context + Question. You are also given the IPA phonemic transcriptions of the equivalent Context + Question (in IPA) as supplementary information. The overall format will be: ## Context + Question:{{context+question}} ## Context + Question(in IPA):{{context+question (in IPA)}} ## Answer: {{answer}} The Answer must appear verbatim in the Context + Question. The answer must be in the target language script. If the Question cannot be answered based on the Context, you will output "unanswerable". Follow the examples if given.

Table 12: Prompt templates by task. $\{\{\cdot\}\}$ denotes the corresponding information for individual samples.

Dataset Statistics. Considering the future extensions of our work towards more languages, we purposely conduct our evaluations on the multilingual datasets covering a wide range of languages. For Machine Translation task, we leverage FLORES dataset (Team et al., 2024b). For AYA-WIKI and AYA-MLQA, we extract from Aya Collection Dataset (Singh et al., 2024) covering 114 languages. However, as Aya leveraged NLLB translation tool (NLLB Team et al., 2022) to generate translations, the translation data might be noisy due to the potential low-quality generations such as <unk> tokens, etc. Therefore, we conduct further quality control steps to filter out the noisy samples. For each task, we conduct evaluations on 500 randomly sampled examples to form the testing set (D_{test}). We also construct an example pool of 10,000 samples from the train_set (D_{pool}) for ICL retrieval where $D_{pool} \cap D_{test} = \varnothing$. The sole major exception is 1012 samples for D_{pool} on FLORES due to the data availability (Team et al., 2024b). Additionally, after quality control filtering for Aya Collection dataset, for MLQA task, $D_{pool} = 5192$ samples for HIN. However since $1012 \gg k$ and $5192 \gg k$ where k=3, performance of ICL retrieval is not heavily affected. ICL retriever extracts top-k examples from D_{pool} to prompt with LLMs where k is the number of in-context examples for the given query sample. In most of our experiments, we experiment with k=3 unless stated otherwise.

We adhere to Apache 2.0 License from Aya Datasets (Singh et al., 2024) and MIT license from Epitran package (Mortensen et al., 2016) when constructing the orthographic-phonemic aligned datasets and conducting evaluations for the aforementioned tasks.

Implementation and Hyperparameters. Similar to our pilot study presented in Section A.1, for mid-sized LLMs, we conduct our empirical study via Ethereum AI evaluation framework. The hyperparameters are set similarly to the default configuration for each individual task. We conduct evaluation on Llama3-8B-Instruct and Qwen2-7B-Instruct models on 2xA100 GPUs 80GB. Each suite of experiments across all evaluated tasks will consume approximately 2 hours, resulting in total of 10 hours for one LLM type.

For Supervised Fine-tuning (SFT) presented in Section 6, following Maheshwary et al. (2024), we conduct full FT on M2Lingual dataset. However, for direct evaluation on the impact of in-language SFT, we extract and FT Llama3-8B-Instruct model only on samples of the targeted non-Latin languages. Our SFT training takes around 4 hours on 5xA100 (80GB) GPUs.

B Additional Experimental Explorations

Llama-3-8B Instruct	AYA-WIKI			FLORES			AYA-MLQA		
	Script	IPA	Mixed	Script	IPA	Mixed	Script	IPA	Mixed
HIN	39.86	39.88	39.44	58.59	58.63	59.16	48.47	49.41	49.25
ARB	26.51	26.90	26.83	59.79	59.40	59.78	32.49	33.46	33.29
ZHO	5.09	4.35	5.04	56.40	55.87	56.46	12.81	12.61	13.00
JPN	2.71	2.64	2.92	53.65	54.30	53.69	27.88	26.59	28.28
Average	18.54	18.44	**18.56**	57.11	57.05	**57.27**	30.41	30.52	**30.96**

Qwen2-7B Instruct	AYA-WIKI			FLORES			AYA-MLQA		
	Script	IPA	Mixed	Script	IPA	Mixed	Script	IPA	Mixed
HIN	32.43	33.64	34.28	57.42	57.29	57.81	46.56	46.98	47.59
ARB	11.47	11.49	13.04	61.94	62.01	61.79	33.30	33.24	33.92
ZHO	1.23	1.43	1.88	58.52	58.42	58.66	8.90	8.06	9.65
JPN	1.91	2.15	2.6	55.82	56.00	56.22	22.38	22.40	23.10
Average	11.76	12.18	**12.95**	58.43	58.43	**58.62**	27.78	27.67	**28.57**

Table 13: Llama3-8B-Instruct and Qwen2-7B-Instruct ICL results using the Dense Retrieval method using Script vs IPA vs Mixed strategy for retrieval - our proposed Mixed retrieval strategy outperforms all other methods across all tasks.

B.1 Experiments with mid-sized 7B/8B LLMs

Beyond Table 2, we conduct further evaluation of our enhanced ICL on other mid-sized LLMs, including: Gemma-7B-Instruct (Team et al., 2024a), Mistral-7B-Instruct (Jiang et al., 2023). The empirical results demonstrate the consistent benefits of our proposed enhanced ICL across the mid-sized (7B/8B) LLMs as observed in Table 14.

B.2 Experiments with GPT-4 and Mixtral

We study proprietary, large-scale production LLMs like GPT-4 (Achiam et al., 2023) and Mixtral-8x22B-Instruct [6], selected for their widespread use in industry, towards understanding whether IPA integration helps. As seen in Table 15 on 100 samples, the performance with these proprietary and production models is not consistent. While further investigation is necessary, it is difficult to ascertain data contamination with these models for all the tasks we study (Deng et al., 2024; Li and Flanigan, 2024), and the performance changes indicate that these models might have data contamination for the tasks we experiment on. We also find some

[6] https://mistral.ai/news/mixtral-8x22b/

Gemma-7B-Instruct	AYA-WIKI - BLEU (↑)					FLORES - chrF (↑)					AYA-MLQA - F1 (↑)				
	0-shot	Random	BM25			0-shot	Random	BM25			0-shot	Random	BM25		
			Script	IPA	Mixed			Script	IPA	Mixed			Script	IPA	Mixed
HIN	11.52	40.92	42.25	42.28	42.83	24.68	63.59	63.61	63.64	63.89	32.65	46.57	47.12	46.53	46.22
ARB	2.52	25.92	27.41	26.12	28.03	20.7	64.07	64.59	64.7	64.91	20.99	29.98	30.22	30.81	32.35
ZHO	1.29	6.49	8.42	9.43	9.28	11.78	58.14	58.31	58.02	58.37	10.28	12.76	13.68	12.64	13.17
JPN	1.84	9.49	12.72	12.04	13.12	17.82	54.95	56	56.81	56.56	21.46	24.20	23.24	24.34	25.15
Average	4.29	20.71	22.70	22.47	**23.32**	18.75	60.19	60.63	60.79	**60.93**	21.35	28.38	28.57	28.58	**29.22**

Mistral-7B-Instruct	AYA-WIKI - BLEU (↑)					FLORES - chrF (↑)					AYA-MLQA - F1 (↑)				
	0-shot	Random	BM25			0-shot	Random	BM25			0-shot	Random	BM25		
			Script	IPA	Mixed			Script	IPA	Mixed			Script	IPA	Mixed
HIN	4.63	40.65	42.25	42.12	42.77	44.28	43.66	43.81	43.77	44.24	16.46	36.74	37.29	36.51	39.46
ARB	1.47	25.24	26.24	24.91	27	50.01	50.02	51.02	50.51	51.08	16.09	28.66	29.81	29.15	30.42
ZHO	1.15	6.97	8.7	9.52	9.93	51.53	53.01	53.52	53.12	53.54	5.11	11.15	13.58	12.06	13.05
JPN	1.73	9.61	11.44	11.55	12.18	49.1	49.91	49.97	50.92	50.27	10.33	20.87	21.78	22.75	21.97
Average	2.25	20.62	22.16	22.03	**22.97**	48.73	49.15	49.58	49.58	**49.78**	12.00	24.36	25.62	25.12	**26.23**

Table 14: Gemma-7B and Mistral-7B Instruct ICL results using the BM25 retrieval method using Script vs IPA vs Mixed strategy for retrieval - our proposed Mixed retrieval strategy outperforms all other methods across all tasks.

GPT-4	AYA-WIKI - BLEU (↑)					FLORES - chrF (↑)					AYA-MLQA - F1 (↑)				
	0-shot	Random	BM25			0-shot	Random	BM25			0-shot	Random	BM25		
			Script	IPA	Mixed			Script	IPA	Mixed			Script	IPA	Mixed
HIN	12.32	22.14	47.04	38.38	30.53	77.16	72.53	76.09	73.49	97.30	65.77	59.15	59.15	61.11	58.16
ARB	20.91	28.11	25.96	8.85	27.16	85.22	70.58	69.49	81.41	72.78	51.85	49.62	50.75	48.48	50.75
ZHO	3.75	10.68	13.13	6.27	10.68	64.72	68.50	64.60	71.42	66.78	30.51	27.59	36.07	36.07	38.71
JPN	27.09	10.68	23.00	19.81	25.45	67.44	62.34	69.86	66.47	64.92	48.48	55.07	50.75	54.01	56.12
Average	16.02	17.90	**27.28**	18.33	23.46	73.64	68.49	70.01	73.20	**75.45**	49.15	47.86	49.18	49.92	**50.93**

Mixtral 8x22B	AYA-WIKI - BLEU (↑)					FLORES - chrF (↑)					AYA-MLQA - F1 (↑)				
	0-shot	Random	BM25			0-shot	Random	BM25			0-shot	Random	BM25		
			Script	IPA	Mixed			Script	IPA	Mixed			Script	IPA	Mixed
HIN	1.85	49.06	10.49	47.06	15.78	64.55	58.45	60.48	64.31	60.76	44.96	49.62	51.85	46.15	43.75
ARB	0.72	1.32	45.46	24.00	22.74	75.57	62.34	70.51	75.50	79.36	30.51	40.00	38.71	24.56	29.06
ZHO	4.46	0.81	**4.37**	2.86	9.86	**0.00**	58.68	64.25	62.24	73.59	29.06	26.09	31.93	26.09	31.93
JPN	1.65	5.16	23.00	1.21	2.73	62.68	29.23	60.03	62.65	64.47	29.06	46.15	44.96	37.40	41.27
Average	2.17	14.09	**20.83**	18.78	12.78	50.70	52.18	63.82	66.17	**69.55**	33.40	40.47	**41.86**	33.55	36.50

Table 15: GPT-4 and Mixtral-8x22B-Instruct ICL results with various retrieval methods using Script vs IPA vs Mixed for retrieval (except 0-shot, all other columns represent results with 3-shot prompting).

samples moderated by the API, however upon further manual examination, the 6 total samples (out of the 400) focus on factual news topics such as a prison catching on fire, or quotes from politicians on nuclear power. Since there are only a few, we expect the performance impact to be minimal. We reserve further examination and exploration for future work.

B.3 Dense ICL retrieval also benefits with IPA

Due to space constraints, without the loss of generality, we focus our studies on BM25 retrieval methods in the main paper, and report dense retrieval results here. For dense ICL retrieval, we select *paraphrase-xlmr-multilingual-v1* (Reimers and Gurevych, 2019) as the Encoder for input query and individual samples in the ICL pools. The max_context_length is set to 512. The sentence representation is leveraged for cosine similarity computation between query and all pooling examples to select the final top-k ICL examples. Results are reported in Table 13.

GPT4	AYA-WIKI			FLORES			AYA-MLQA		
	Script	IPA	Mixed	Script	IPA	Mixed	Script	IPA	Mixed
HIN	30.21	33.66	35.57	73.97	73.61	89.91	62.07	63.95	57.14
ARB	25.98	20.49	26.99	87.05	77.17	78.52	49.62	52.94	48.48
ZHO	8.91	9.43	12.19	65.50	72.82	68.50	30.51	36.07	37.40
JPN	26.99	10.68	12.70	68.69	73.34	68.83	51.85	51.85	54.01
Average	**23.02**	18.57	21.86	73.80	74.24	**76.44**	48.51	**51.20**	49.26

Mixtral 8x22B	AYA-WIKI			FLORES			AYA-MLQA		
	Script	IPA	Mixed	Script	IPA	Mixed	Script	IPA	Mixed
HIN	37.21	41.17	9.60	60.29	65.12	78.12	42.52	49.62	42.52
ARB	29.17	6.96	39.13	76.91	66.67	76.20	42.52	38.71	38.71
ZHO	1.30	0.84	15.14	58.05	52.79	68.44	36.07	30.51	31.93
JPN	2.57	13.60	2.77	60.73	60.82	64.52	51.85	43.75	46.15
Average	**17.56**	15.64	16.66	64.00	61.35	**71.82**	**43.24**	40.65	39.83

Table 16: GPT4 and Mixtral-8x22B ICL results using the Dense Retrieval method using Script vs IPA vs Mixed strategy for retrieval.

Consistent with the sparse BM25 Retrieval results reported in Section 4.2, we observe that our `Mixed` strategy outperforms `Script` and `IPA` based ones on all downstream tasks. Similar to observed performance with open-source LLMs, GPT4 and Mixtral 8x22B also show a trend where the `Mixed` strategy outperforms most others. These observations imply that ICL retrieval benefits from looking at both orthographic and phonemic information for better example selection, guiding the LLMs towards desired generation output for downstream tasks.

	Divide-Conquer	Harmonic	Split-Half			Concat	Mix (Ours)
			IPA+Script	Script+IPA	Shuffle		
AYA-WIKI	21.34	21.61	22.15	22.14	22.02	21.62	**22.28**
FLORES	57.52	57.33	57.13	57.53	57.50	57.62	**57.84**
AYA-MLQA	28.94	28.87	28.48	28.17	28.55	28.72	**30.97**

Table 17: Comparing 4 different mixing strategies under 6-shot settings. Even number of shot is required for comparison with `Split-Half` approach. Llama3-8B-Instruct is leveraged as the base LLM.

B.4 Impact of different mixing strategies on ICL

Besides our `Mixed` strategy, there exist different approaches towards aggregating information from Script and IPA ICL retrieval. We specifically consider 4 different approaches: (1) using the `Harmonic Mean` to calculate the mixed score for each pool example (2) `Split-Half`: We retrieve top-(k//2) examples from Script and IPA separately, then aggregate them together to form the final top-k ICL samples. Within this approach, we evaluate 3 different potential ordering to aggregate ICL examples from different sources, including (a) Script+IPA, (b) IPA+Script, (c) Random Shuffle. This approach requires even k-shot samples, (3) `Divide-Conquer`: After sorting and retrieving top-k samples for both IPA and Script, we concatenate the corresponding scores to form top-2k samples. These samples are then ranked and filtered down to the top-k samples again based on their corresponding BM25 scores, (4) `Append`: We simply concatenate the scores from the two approaches and retrieve the top-k highest score as the final selected examples. Unlike previous approaches, this approach can possibly result in similar ICL examples being selected twice in the top-k samples.

Based on our empirical study in Table 17, our `Mixed` strategy yields the best performance across our evaluated tasks. `Split-Half` aggregation is limited to even-shot ICL samples and can potentially suffer from ordering sensitivity, leading to variable evaluation performance (Lu et al., 2022).

B.5 Possible Future Studies with SFT and ICL

For better understanding of the enhanced ICL as compared to continual pre-training or Supervised Fine-tuning (SFT), we conduct additional studies in which we fine-tune Llama3-8B-Instruct model with additional multilingual M2Lingual dataset (Maheshwary et al., 2024). Further implementation details are provided in Appendix A.2. As indi-

Task	0-shot	M2Lingual SFT		ICL		
		Script data	Script + IPA data	3-shot	6-shot	10-shot
AYA-WIKI	2.86	8.86	6.46	19.85	22.28	**23.16**
FLORES	49.67	18.15	19.74	57.44	57.84	**58.03**
AYA-MLQA	20.84	23.60	24.74	30.75	30.97	**31.40**

Table 18: Comparison between BM25 `Mixed-ICL` and two variations of SFT models on M2Lingual dataset: (1) Script-only data and (2) Script+IPA data. ICL approach is evaluated under 3-shot, 6-shot and 10-shot settings and SFT methods are under 0-shot evaluation. Llama3-8B-Instruct is leveraged as the base LLM.

cated in Table 18, without additional multilingual training data for the targeted languages, our BM25-Mixed ICL outperforms attempts in FT LLMs with additional multilingual dataset when the number of ICL examples reach 10 shots. This study reveals two essential implications: (1) High-quality ICL selection not only saves the computational cost of additional training but also quickly integrates rare knowledge of IPA with LLMs, (2) Naively SFT LLMs with phonemic-orthographic data might not be sufficient to extract the alignment between IPAs and scripts, emphasizing the goal of our work in gaining deeper understanding of IPAs integration with LLMs via prompting before FT is executed.

C Detailed Results on Latin languages

For further clarity of Section 6, we provide additional details of the performance across the evaluated tasks on Latin languages, including: German (deu), French (fra), Spanish (spa) and Portuguese (por) in Table 20. These results are leveraged to calculate the relative performance gain as demonstrated in Table 4.

D AYA-MLQA Performance Analysis

As observed in Table 2, our empirical studies yield less competitive performance than the originally reported performance of fine-tuned Pre-trained Language Models (PLMs) on MLQA dataset (Lewis et al., 2020). We hypothesize the discrepancy is mostly caused by the issue of dataset quality differences between the original MLQA (*ORI_MLQA*) and MLQA from Aya Collection (*AYA_MLQA*).

Data Quality Difference Our major objective in leveraging AYA dataset (Singh et al., 2024) is the broad language coverage up to 102 languages, allowing for the further investigation beyond our 4 targeted languages. However, multilingual versions of AYA datasets across tasks are generated via off-the-shelf NLLB machine translation (NLLB Team et al., 2022), which can be prone to errors. Therefore, the dataset quality between *AYA_MLQA* and *ORI_MLQA* might be different, leading to incomparable performance between ours and previously reported PLMs performance. We further validate our hypothesis via empirical evaluation of our enhanced ICL on *ORI_MLQA* and *AYA_MLQA* as demonstrated in Table 19. More specifically, we observe approximately 22.71 absolute F1 points between *ORI_MLQA* and *AYA_MLQA*. Additionally, our observed performance is on par with the originally reported mBERT FT.

	AYA_MLQA	*ORI_MLQA*		
	Ours	Ours	XLM FT	mBERT FT
HIN	49.30	63.46	34.40	50.20
ARB	30.49	55.74	42.50	52.30
ZHO	14.38	43.10	40.50	59.60
Average	**31.39**	**54.10**	**39.13**	**54.03**

Table 19: Performance evaluation comparison between *AYA_MLQA* and *ORI_MLQA* with Llama3-8B-Instruct and reported baseline of XLM and mBERT PLM.

Llama3-8B-Instruct	AYA-WIKI- BLEU (↑)					FLORES- chrF (↑)					AYA-MLQA- F1 (↑)				
	0-shot	Random	BM25			0-shot	Random	BM25			0-shot	Random	BM25		
			Script	IPA	Mixed			Script	IPA	Mixed			Script	IPA	Mixed
DEU	11.5	28.04	31.53	31.38	31.82	62.02	66.16	66.43	66.54	66.54	40.67	47.63	45.91	44.52	48.50
FRA	15.84	35.93	41.18	40.37	40.86	57.47	67.05	67.24	67.35	67.32	12.26	12.54	13.11	12.69	14.09
SPA	12.08	39.23	42.33	42.4	42.38	51.57	60.10	60.38	60.43	60.42	24.70	27.28	27.45	27.99	27.33
POR	18.25	30.61	34.92	35.83	35.57	60.45	70.18	70.30	70.52	70.66	40.52	44.96	46.74	52.62	48.56
Average	**14.42**	**33.45**	**37.49**	**37.50**	**37.66**	**57.88**	**65.87**	**66.09**	**66.21**	**66.24**	**29.54**	**33.10**	**33.30**	**34.46**	**34.62**

Table 20: Llama3-8B-Instruct ICL results using the BM25 retrieval method using `Script` vs `IPA` vs `Mixed` retrieval strategy for Latin-based languages (DEU, FRA, SPA, POR).