

LANGUAGES ARE MODALITIES

CROSS-LINGUAL ALIGNMENT VIA ENCODER INJECTION

Rajan Agarwal*

University of Waterloo

r34agarw@uwaterloo.ca

Aarush Gupta*

Independent

hiaarushgupta@gmail.com

ABSTRACT

Instruction-tuned Large Language Models (LLMs) underperform on low-resource, non-Latin scripts due to tokenizer fragmentation and weak cross-lingual coupling. We present LLINK (Latent Language Injection for Non-English Knowledge), a compute-efficient language-as-modality method that conditions an instruction-tuned decoder without changing the tokenizer or retraining the decoder. First, we align sentence embeddings from a frozen multilingual encoder to the decoder’s latent embedding space at a reserved position via a lightweight contrastive projector. Second, the vector is expanded into K soft slots and trained with minimal adapters so the frozen decoder consumes the signal. LLINK substantially improves bilingual retrieval and achieves 81.3% preference over the base model and 63.6% over direct finetuning in LLM-judged Q&A evaluations. We further find that improvements can be attributed to reduced tokenization inflation and a stronger cross-lingual alignment, despite the model having residual weaknesses in numeric fidelity. Treating low-resource languages as a modality offers a practical path to stronger cross-lingual alignment in lightweight LLMs.

1 Introduction

Natural languages serve as humanity’s primary interface, each encoding unique pragmatics, scripts, and writing systems. A central challenge in machine learning is enabling models to understand, generate, and translate across linguistic variations, to make language models accessible to everyone. However, frontier LLMs today, predominantly trained on English data, demonstrate significant performance degradation on tasks involving low-resource languages, specifically those with non-Latin scripts [Petrov et al., 2023, Limisiewicz et al., 2023].

To mitigate this, current approaches involve in-context learning or moderate finetuning. However, these introduce tokenizer fragmentation, which inflates non-English text into substantially longer sequences, and weak cross-lingual coupling within model representations [Petrov et al., 2023, Limisiewicz et al., 2023, Ahia and Kumar, 2023, Qin et al., 2025]. Existing solutions to these derivative issues include multilingual pretraining [Conneau et al., 2020, Xue et al., 2021, Le Scao and et al., 2022] and tokenizer-free byte-level models [Xue and et al., 2022] and character-level encoders [Clark et al., 2022, Tay et al., 2022], but carry significant computational and data requirements. Even recent multilingual instruction models still rely on large-scale training and careful tokenizer design [Cohere for AI, 2024, Yang and Team, 2025]. On the other hand, parameter-efficient finetuning (PEFT) strategies, such as LoRA, IA³, and BitFit [Hu et al., 2022, Liu et al., 2022, Ben-Zaken et al., 2022], reduce the adaptation cost but still commonly rely on substantial multilingual supervision.

Many languages have sparse web footprints and uneven curation, which makes full multilingual pretraining of LLMs expensive. By contrast, masked-LM encoders trained over many languages handle scarcity relatively well, absorbing monolingual text and share subword/byte structure across scripts, providing strong sequence features. However, these strengths have not translated cleanly to instruction-tuned LLMs. Adding sparse low-resource corpora into continued pretraining or SFT often has minimal effect, causes regressions in English-based evaluation performance, requires more expensive tokenizers, or leads to

*Denotes equal contribution. Work completed under Cohere Labs Community.

longer training runs to achieve parity. Therefore, this gap suggests a retrofit path that uses strong external multilingual encoders, rather than trying to make a one-size-fits-all LLM.

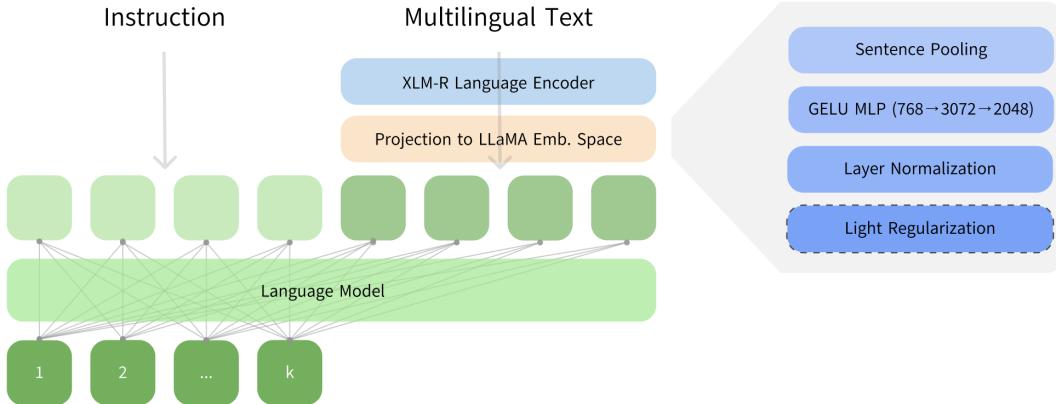


Figure 1: Illustration of LLINK Architecture, passing Multilingual text through a projection model to match LLaMA’s embedding space, then to the LLM to produce an output using the translated tokens. Dotted lines represent train-time only.

This work treats low-resource source languages as an auxiliary modality for language models. Our paper makes the following contributions:

- *Language-as-Modality.* We frame low-resource languages as a modality and inject them into decoder-only LLMs via a compact set of soft slots, bypassing decoder tokenization for non-Latin scripts. This shifts cost from fragmented decoder tokens to a small encoder plus K slots, yielding up to $\sim 3 \times$ fewer decoder tokens at prompt time on Khmer while improving cross-lingual quality.
- *Contextual teacher alignment.* Between the frozen multilingual encoder and LLM’s own hidden state, we apply a contextual teacher alignment at a reserved position, rather than to static token embeddings, providing a stable, context-aware target that strengthens cross-lingual coupling without modifying the tokenizer or decoder weights.
- *Usage-enforcing slot objective.* We add a usage-enforcing objective that penalizes the model if replacing injected slots with base embeddings does not worsen loss, making reliance on the foreign signal measurable and trainable.

We train and empirically validate LLINK on Khmer-to-English translation and Q&A tasks using the ParaCrawl En-Km v2 dataset [Bañón et al., 2020, par, 2020]. Our method achieves substantial improvements in bilingual retrieval, which we treat as a proxy for evaluating cross-lingual alignment, over direct finetuning baselines. Through LLM-as-Judge pairwise evaluation [Zheng et al., 2023, Liu et al., 2023b], LLINK-enhanced output are preferred to both the original base model and directly finetuned variants, especially when introduced to tokens out of distribution from the SFT.

2 Related Work

2.1 Tokenizer fragmentation and multilingual inequity.

Large cross-language disparities arise at the tokenization layer. Petrov et al. [2023] quantify length inflation up to $15 \times$ across languages and show the effect persists for multilingual and byte/character tokenizers. Limisiewicz et al. [2023] analyze vocabulary allocation and overlap, relating them to downstream performance. Tokenizer-free models like ByT5 reduce subword dependence by operating on bytes [Xue and et al., 2022], and character-level encoders like CANINE and Charformer avoid subword tokenization entirely [Clark et al., 2022, Tay et al., 2022], but these approaches induce longer sequences and higher training cost. A complementary line of work adapts vocabularies to reduce cross-lingual inflation without fully retraining the model [Yamaguchi et al., 2024]. These remedies require retraining with new tokenization or accept efficiency penalties; neither retrofits an existing decoder-only LLM’s tokenizer at inference time.

2.2 LangBridge & Multilingual Bridge

LangBridge [Yoon et al., 2024] introduces a lightweight bridge that maps a multilingual encoder’s hidden states (e.g., mT5) into a small sequence of soft-prompt vectors in a frozen decoder-only LM’s input-embedding space. The bridge is trained on English instruction data with a language-modeling objective, so that at inference time non-English inputs are routed through the encoder and injected as continuous prompts, yielding strong zero-shot multilingual reasoning despite English-only supervision. In contrast, LLINK aligns encoder outputs to a reserved decoder hidden state via a two-layer MLP (rather than the input embedding stream) and adds an explicit usage-enforcement objective to ensure the injected slots are consumed during generation.

2.3 Multimodal Encoder Bridges to LLMs.

Multimodal stacks such as BLIP-2 [Li et al., 2023] and LLaVA [Liu et al., 2023a] show that small cross-modal encoder injection can provide slot embeddings that the LLM consumes alongside text. BLIP-2 does this with a Q-Former that learns a small set of queries and lets them cross-attend to the encoder features through several transformer layers, adding both parameters and a inference/training cost that grows with the number of queries. LLaVA demonstrates that a simple linear projection trained on instruction-following data suffices to align vision encoder outputs with the LLM’s token space, making it much lighter at inference. Speech-to-text and speech-to-LLM systems such as Seamless similarly project non-text modalities into an LLM-consumable representation, reinforcing the view of “modality as just another encoder” [Seamless Communication Team, 2023]. Prior multilingual bridges typically inject at the embedding stream [Yoon et al., 2024]; our method follows this lightweight style but aligns at a reserved decoder hidden state and uses a two-layer MLP plus an explicit usage-enforcement objective, so LLINK keeps a runtime profile closer to LLaVA than to BLIP-2 while improving cross-lingual coupling.

3 Background

Modern large language models process text through subword tokenization, typically using Byte-Pair Encoding (BPE) or related algorithms [Sennrich et al., 2016, Kudo and Richardson, 2018] trained on predominantly English corpora. This creates severe inefficiencies for non-Latin scripts, with inflation ratios reaching $15\times$ for some languages [Petrov et al., 2023, Lotz et al., 2025], and similar challenges documented for Southeast Asian scripts [Ahia and Kumar, 2023]. To quantify this effect for Khmer, we measure tokenization on Khmer-English sentence pairs from ParaCrawl [Bañón et al., 2020] using LLaMA-style BPE vocabulary and observe substantial fragmentation.

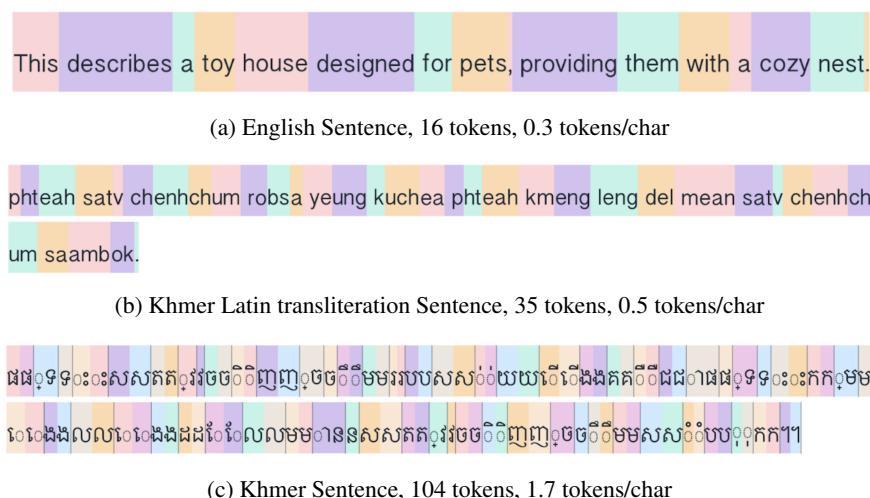


Figure 2: Tokenization of the same sentence with the LLaMA-3.2-1B tokenizer — English: 16 tokens (0.3 tok/char); Khmer translit: 35 (0.5); Khmer: 104 (1.7). Dividers on Khmer show duplicate tokens mapping to the same character.

We examine that processing a Khmer sentence yields approximately $6\times$ more tokens than its English equivalent, and transformer attention compute scales quadratically with sequence length [Vaswani et al., 2017]. A 200-character Khmer sentence might consume 130 tokens, leaving substantially less room for task instructions, few-shot examples, or generated outputs compared to English. The model must learn cross-lingual representations across fragmented tokens, making alignment optimization more difficult.

In Figure 2, an English sentence tokenizing to 16 tokens expands to 35 tokens when written in Latin transliteration, and further explodes to 104 tokens in native Khmer script using the LLaMA tokenizer. This near-order-of-magnitude difference persists across the distribution. Standard parameter-efficient adaptation methods like LoRA operate on these fragmented token sequences, only inheriting the computational and context-level effects. Even with perfect fine-tuning, the model processes more tokens per forward pass for Khmer inputs compared to English. Our approach sidesteps tokenization at the decoder by treating Khmer as an auxiliary modality, by encoding Khmer text and aligning it to the LLM’s latent space through a lightweight projector, *shifting* tokenization overhead to a small, fixed-cost encoder and a few soft slots to reduce decoder-side compute.

4 Methodology

We use a two-stage bridge that treats low-resource text as an auxiliary modality. Stage A learns a small connector that maps a frozen multilingual encoder’s sentence representation into the LLM’s latent space at a reserved position—no tokenizer changes, no heavy retraining. Stage B then exposes this signal to the decoder via a few soft slots and lightly tunes small modules so the model actually relies on it during generation. This approach circumvents tokenization inflation while preserving the base model’s weights. We provide the architectural and training details in the following sections; related connector-style approaches are discussed in Section 5.

4.1 Stage A: Contrastive Alignment

We first build a single, deterministic “foreign representation” at a reserved slot the decoder can read. A frozen XLM-R encodes a Khmer sentence and we mask-mean pool the token states to a sentence vector $\mathbf{z}_F \in \mathbb{R}^{768}$. On the English side, we append a reserved token at the end of the user instruction and take the final hidden state at that position as the teacher target (h_E). We use the following prompt template (Stage A): User:<instruction><foreign_emb> Assistant:.... . This fixes (h_E) to a known context position and makes the target prompt-dependent but decoder-stable. As a result, final latent state at that position is the teacher target $\mathbf{h}_E \in \mathbb{R}^{2048}$. The input-embedding row for <foreign_emb> is zero-initialized; under RMSNorm (no bias) [Zhang and Sennrich, 2019] a zero vector keeps the slot neutral before alignment as residual self-attention moves the state only via context.

A lightweight projector MLP g maps encoder representations into the decoder space: $g : 768 \rightarrow 3072 \rightarrow 2048$ with Linear+GELU [Hendrycks and Gimpel, 2016], dropout 0.10, and a final LayerNorm [Ba et al., 2016]. We set $\mathbf{p}_F = g(\mathbf{z}_F)$ and update *only* g , ensuring the multilingual encoder and decoder stay frozen. The training objective is

$$\mathcal{L} = \frac{1}{2} [\text{NCE}(\mathbf{p} \rightarrow \mathbf{h}) + \text{NCE}(\mathbf{h} \rightarrow \mathbf{p})] + \lambda_{\text{dir}} \mathcal{L}_{\text{dir}} + \lambda_{\text{norm}} \mathcal{L}_{\text{norm}}.$$

For the symmetric InfoNCE [van den Oord et al., 2018], for each parallel pair we treat $(\mathbf{p}_F, \mathbf{h}_E)$ as the positive, use in-batch negatives, and augment the denominator with hard negatives. [He et al., 2020, Radford et al., 2021]. For a strong selection of negatives, we maintain a 32768-item fp16 FIFO queue of cosine-normalized teacher vectors and, per step, select the 256 hardest by cosine similarity.

We add two light regularizers. First, a direction term $\mathcal{L}_{\text{dir}} = \|\mathbf{p}_F - \mathbf{h}_E\|^2$ with L2 regularization and vector normalization, as well as weight 0.05, acting as an anchor to push alignment beyond what InfoNCE alone enforces. Second, a log-norm matching term $\mathcal{L}_{\text{norm}} = (\log \|\mathbf{p}_F\| - \log \|\mathbf{h}_E\|)^2$ with weight 0.02, which keeps magnitudes comparable and prevents exploding or vanishing norms. The result is a compact vector at a known position that reliably summarizes the foreign sentence in the decoder’s own space.

4.2 Stage B: Multi-Token Injection with Usage Enforcement

However, the decoder may still disregard the aligned foreign representation without explicit training signals. Stage B aims to encourage usage of the vector by expanding it into K soft tokens (akin to soft

prompt tuning [Lester et al., 2021]) and adding training signals so the model learns to rely on them during generation.

In LLINK, we expand the Stage A vector into K slot embeddings and replace the single `<foreign_emb>` with reserved tokens `<f0>–<fk-1>`. These slots live in the context like ordinary tokens and can be attended at every layer, giving enough capacity to carry longer or denser content than a single 2048-d point. To expand the vector, we unit-normalize, apply a non-affine LayerNorm, and multiply by a learned scalar initialized to the median norm of the base embedding matrix; the $K = 8$ slots are inserted after the instruction as `<f0>...<f7>`. We inject by computing base embeddings and overwriting the rows at the reserved token positions so the decoder sees them directly.

Then, to teach the model to read the slots, we apply low-rank LoRA to attention and MLP projections and train it jointly with the scale, adapter, and expander. This keeps the Stage A projector and all other base weights remain frozen. We synthetically generate Q&A prompts that use a `<foreign_emb>` vector and reply in English. Four task templates were included in this distribution, including `bullet_pointify`, `translate_to_english`, `summarize_in_english` and `qa_about_text`.

Prompts are tokenized with `<f0>–<f7>` and the pipeline mirrors inference, first encoding Khmer with XLM-R, projecting its embeddings with scale/adapter, expand to K slots and then decode. Providing features alone does not guarantee usage, so we add a lightweight usage-contrast. Every third step we compute the supervised fine-tuning loss with injected slots, \mathcal{L}_{SFT} , and a “zeroed” loss $\mathcal{L}_{\text{zero}}$ where the reserved positions are restored to the original embeddings. We penalize cases where removal helps:

$$\mathcal{L}_{\text{contrast}} = 0.05 \max(0, \mathcal{L}_{\text{SFT}} - \mathcal{L}_{\text{zero}}).$$

Two small alignment auxiliaries anchor the slots to the Stage A target space without dominating training. A unit-vector matching term increases the cosine similarity to the teacher slot, and an InfoNCE loss with weight 0.01 against the same teacher to prevent drift. The total objective is $\mathcal{L}_{\text{SFT}} + \mathcal{L}_{\text{contrast}} + (\text{auxiliaries})$. In practice this shifts the model from paraphrasing around the slots to actually using them, while keeping changes to the base LLM minimal.

5 Experimental Setup

5.1 Data

We use ParaCrawl v2 English–Khmer [Bañón et al., 2020] for all experiments. We truncate Khmer strings to at most 256 characters, and take a 140k English-Khmer pair subset from the dataset, dividing it into 100k training pairs for Stage A and 40k holdout for retrieval evaluation.

Stage B requires instruction-following examples that use the injection pipeline. We synthesize instruction-following examples from parallel pairs using a LLaMA-3 70B instruction-tuned model [Llama Team, AI @ Meta, 2024]. For each Khmer sentence, the model is conditioned on the reference English translation so that targets are anchored to ground truth rather than model output. After filtering for non-empty inputs/targets, presence of the reserved token `<foreign_emb>` in the prompt, and Khmer length between 12-256 characters, we select the Stage B set with 40k training and 2k validation examples.

5.2 Baselines and Evaluation

The base model is LLaMA-3.2-1B-Instruct with no adaptation, processing Khmer directly through its BPE tokenizer and suffering from the measured $6.5\times$ fragmentation. Direct fine-tuning applies LoRA (rank 16, alpha 16, same configuration as LLINK’s LoRA component) to the base model on instruction data where Khmer is tokenized normally, representing standard parameter-efficient adaptation.

For evaluation, we first use bilingual retrieval to represent cross-lingual alignment quality, following recent CLIR-style evaluations for multilingual LLMs [Goworek et al., 2025]. Given N English-Khmer pairs, we encode all Khmer sentences through our pipeline and all English sentences through the teacher position (append `<foreign_emb>`, extract hidden state). We compute cosine similarity and report Recall at ranks 1, 5, and 10, plus Mean Reciprocal Rank (MRR) and Mean Rank. This tests whether aligned representations enable correct matching, which correlates with translation quality. To test generations, we use a LLaMA-3 70B instruction model as a judge [Llama Team, AI @ Meta, 2024], following LLM-as-Judge methodology [Zheng et al., 2023, Liu et al., 2023b], with anonymized pairwise comparisons. For each of 500 test examples, we generate outputs from two systems and record full win/loss/tie breakdowns.

Method	R@1	R@5	R@10	MRR	Mean Rank
Direct fine-tune	0.104	0.248	0.352	0.160	24.7
LLINK (Stage A)	0.430	0.706	0.819	0.642	3.8
LLINK (Full)	0.450	0.724	0.835	0.660	3.4

Table 1: Bilingual retrieval (R@k, MRR, mean rank) on n = 1,024 held-out Khmer–English pairs.

Bucket	Comparison	Wins %	Losses %	Ties %	Preference
Content Understanding (n=500)	LLINK vs. Base	69	11	20	86.3%
	LLINK vs. Fine-tune	45	23	32	66.2%
Q&A (n=500)	LLINK vs. Base	48	15	36	76.2%
	LLINK vs. Fine-tune	39	25	36	60.9%

Table 2: LLM-as-judge evaluation with selected permutations (judge sees the human reference; preference excludes ties). Judge: LLaMA 3.1 70B Instruct.

6 Results

6.1 Retrieval alignment

We evaluate Khmer to English alignment on a held-out set of 1,024 parallel pairs. For each Khmer sentence, we compare its normalized LLINK projection to the normalized teacher vectors extracted at `<foreign_emb>` for all English sentences and rank the gold target among 1,024 candidates. We report Recall@k (R@k), mean reciprocal rank (MRR), and mean rank. Table 1 shows large gains over a direct fine-tune, with R@1 improving from 0.104 to 0.450 ($\sim 4.3 \times$), and a sharp drop in mean rank. Stage A provides the primary improvement by bypassing tokenization inflation and directly aligning to the decoder’s representation space, reducing false matches.

We measure end-to-end quality with anonymized A/B comparisons and an LLM judge (LLaMA 3.1 70B Instruct). For each prompt, the judge sees two model outputs in random order and the human reference translation (for verification), then returns {win, loss, tie}. Preference is wins/(wins+losses). We bucket prompts into two task types: (i) Q&A about the foreign content, and (ii) content understanding (translation, summary, paraphrase, title).

Gains are largest on Q&A, where the slots act like a grounded summary the decoder can copy facts from. On content-understanding tasks, LLINK improves precision (fewer mixed-script or off-topic outputs) but will paraphrase rather than translate literally. This behavior is expected, as the encoding process, project it to another space and use those vectors, which would preserve meaning but not specific words or numbers. Stage B’s usage-contrast helps, but lexical exactness can still lag when the underlying reference contains uncommon terms. These show in many forms, a few notable examples being unit slips (kW vs MW), category substitution (“games” vs “instruments”), and occasional over-summarization.

7 Analysis

7.1 Understanding LLINK’s Effectiveness

The dramatic performance gap between Stage A alone (R@1: 0.104 to 0.430) and the full model (to 0.450) reveals that tokenization fragmentation is the dominant bottleneck for cross-lingual understanding. By replacing 104 fragmented Khmer tokens with 8 semantic slots, we inherently change how the decoder processes foreign text. Now, instead of attending over incomprehensible fragments, it sees coherent semantic units. Unlike static embedding mapping approaches, we align to hidden states at a reserved position *after* the decoder has processed the English context. This provides a richer, context-aware target that already encodes task-relevant information and expected answer formats. This method teaches the projection to produce representations that fit naturally into the decoder’s existing computational flow. The frozen decoder acts as an implicit regularizer, preventing the aligned representations from drifting into decoder-incompatible regions of the hidden space.

Positive cases (LLINK preferred)	Negative cases (baseline preferred)
Q&A <i>Question:</i> What policy is stated? <i>Translated Khmer:</i> We do not share your contact information unless you consent. <i>Base:</i> (untranslatable Khmer; unhelpful) <i>LLINK:</i> The company will not disclose personal data without consent.	Content understanding—translation <i>Task:</i> Translate to English. <i>Translated Khmer:</i> We have four perfect crafts to play with. <i>Base:</i> We have four good instruments to play with. <i>LLINK:</i> This suggests a variety of games available for players.
Q&A <i>Question:</i> When will classes resume? <i>Translated Khmer:</i> Classes resume Monday. <i>Base:</i> School will reopen soon. <i>LLINK:</i> Classes will resume on Monday.	Q&A <i>Question:</i> What does the text say about capacity? <i>Translated Khmer:</i> Capacity: 30 MW. <i>Base:</i> Capacity: 30 MW. <i>LLINK:</i> This indicates the power rating of the device, which is 1.5 kW.
Content understanding—translation <i>Task:</i> Translate to English. <i>Translated Khmer:</i> There is a specific door type. <i>Base:</i> This indicates a specific model number. <i>LLINK:</i> This indicates a specific type of door.	Content understanding—summary <i>Task:</i> Summarize in English. <i>Translated Khmer:</i> Submit the application by Friday. <i>Base:</i> Submit the application. <i>LLINK:</i> Send the form this weekend.

Table 3: Side-by-side qualitative cases used in the LLM-as-judge evaluation with judge LLaMA 3.1 70B Instruct; comparisons are blinded A/B with access to the human reference for verification. LLINK vs Base, LLINK averages to a 81.3% preference. LLINK vs Fine-Tune, 63.6% preference.

This architectural choice creates a natural trade-off. Consider LLINK as a lossy semantic compression, such that variable-length sequences become fixed K-dimensional representations. This explains both our strong performance on meaning-preservation tasks (Q&A, retrieval) and systematic failures on surface-form tasks (numeric precision, exact translation). The observed confusion between "30 MW" and "1.5 kW" reflects how multilingual encoders represent numbers on logarithmic scales where these values are semantically proximate.

The usage-enforcement objective also reveals that even well-aligned representations can be ignored without explicit training pressure. The decoder's strong English priors resist foreign signals, preferring to paraphrase around unknown content rather than utilize it directly. This resistance might explain why previous multilingual bridging attempts showed limited success without extensive adaptation.

7.2 Computational Trade-offs

LLINK achieves approximately 3x reduction in decoder tokens in our experiments, by shifting computational burden from the decoder to a one-time encoding cost. This trade-off favors scenarios where encoder cost amortizes across multiple uses (batch processing, caching, or repeated queries) but may not benefit single-pass translation.

The preference gaps in judge evaluation (81.3% vs base, 63.6% vs fine-tune) suggest the base model produces mixed-script nonsense, fine-tuning learns brittle pattern matching on fragments, while LLINK maintains semantic coherence but loses lexical precision. This creates a taxonomy, where semantic understanding tasks benefit from LLINK's approach, while applications requiring exact reproduction may need augmentation with copying mechanisms or hybrid strategies.

7.3 Future Work

While LLINK demonstrates promise for Khmer-English tasks, several directions merit exploration:

Scalability across languages and models. Testing on typologically diverse languages (Arabic RTL, Chinese logographic, Swahili agglutinative) would validate generalization. Similarly, scaling to larger decoders (7B, 13B) requires investigation. It may be hypothesized that stronger English priors in larger models will necessitate adjusted usage enforcement or increased K.

Dynamic slot allocation. Our fixed K=8 represents a compromise across tasks. Adaptive allocation based on input length, complexity, or entropy could improve efficiency. For example, simple queries

might need only $K=2\text{-}4$, while technical documents benefit from $K=12\text{-}16$. A lightweight classifier could predict optimal K at inference time.

Hybrid precision mechanisms. To address numeric and entity errors, we envision augmenting LLINK with specialized pathways: (1) a copying mechanism that preserves exact strings when detected, (2) dedicated slots for numbers that bypass semantic compression, or (3) attention supervision that encourages direct slot-to-output correspondence for critical tokens.

Many-to-many language bridging. Current work assumes English as the target. Extending to arbitrary language pairs requires either training pairwise projectors or learning a universal interlingual space. The latter is appealing but may sacrifice language-specific nuances.

8 Conclusion

LLINK frames low-resource languages as a modality for decoder-only LLMs, aligning compact foreign representations to a place the decoder already understands and then ensuring that this signal is actually used. This design circumvents tokenization inflation and delivers robust semantic coupling with modest engineering and compute. The same design also explains how compressed, slot-based injection favors meaning over surface form and can lose exact numerals and lexical detail. The analysis above identifies why this happens, when it matters, and how to mitigate it.

Treating low-resource, non-Latin scripts as a modality offers a compute-efficient path to improved cross-lingual behavior without retraining tokenizers or decoders, potentially broadening access for underserved languages. At the same time, mis-translations that alter numbers, units, or named entities can have outsized impact. With copy-aware training, mild structural capacity in the slots, diversified teacher targets, and numeracy-focused supervision, we anticipate maintaining LLINK’s efficiency while closing the gap on lexical fidelity; we intend that this work helps move closer to practical, small-model cross-lingual systems that serve languages underrepresented in current tokenizers and pre-training.

Acknowledgments

We thank Cohere Labs for their support, the Modal team for compute, the ParaCrawl project for parallel data, and the XLM-R and LLaMA teams for open-source models that made this work possible.

References

- Paracrawl project website. <https://paracrawl.eu/>, 2020. Accessed 2025-10-03.
- Oluwatobi Ahia and Sriram Kumar. Do all languages cost the same? tokenization in the era of commercial language models. In *Proceedings of EMNLP*, 2023. URL <https://aclanthology.org/2023.emnlp-main.614.pdf>.
- Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. URL <https://arxiv.org/abs/1607.06450>.
- Marta Bañón, Pinzhen Chen, Barry Haddow, Kenneth Heafield, Hieu Hoang, Miquel Esplà-Gomis, Mikel L. Forcada, Amir Kotvia, Sergio Ortiz Rojas, Ferran Pla Sempere, José A. Ramón, Josep A. Sarrís, Matúš Strelec, Brian Thompson, William Waites, Dane Sherburne Wiggins, and François Yvon. Paracrawl: Web-scale acquisition of parallel corpora. In *Proceedings of ACL*, 2020. URL <https://aclanthology.org/2020.acl-main.417.pdf>.
- Elad Ben-Zaken, Shauli Ravfogel, and Yoav Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of ACL (Short)*, 2022. URL <https://aclanthology.org/2022.acl-short.1/>.
- Jonathan H. Clark, Dan Garrette, Iulia Turc, and John Wieting. Canine: Pre-training an efficient tokenization-free encoder for language representation. *Transactions of the Association for Computational Linguistics*, 10:73–91, 2022. doi: 10.1162/tacl_a_00448. URL <https://aclanthology.org/2022.tacl-1.5>.
- Cohere for AI. Aya expanse: Scaling multilingual instruction models. *arXiv preprint arXiv:2412.09455*, 2024. Technical Report.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of ACL*, 2020. URL <https://aclanthology.org/2020.acl-main.747/>.
- Roksana Goworek, Olivia Macmillan-Scott, and Eda B. Özgün. Bridging language gaps: Advances in cross-lingual information retrieval with multilingual llms. *arXiv preprint arXiv:2510.00908*, 2025.
- Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9729–9738, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/papers/He_Momentum_Contrast_for_Unsupervised_Visual_Representation_Learning_CVPR_2020_paper.pdf.
- Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. URL <https://arxiv.org/abs/1606.08415>.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022. URL <https://arxiv.org/abs/2106.09685>.
- Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium, 2018. Association for Computational Linguistics. URL <https://aclanthology.org/D18-2012/>.
- Teven Le Scao and et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022. URL <https://arxiv.org/abs/2211.05100>.
- Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. URL <https://aclanthology.org/2021.emnlp-main.243/>.
- Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of ICML*, 2023. URL <https://arxiv.org/abs/2301.12597>.
- Tomasz Limisiewicz, Jiří Balhar, and David Mareček. Tokenization impacts multilingual language modeling: Assessing vocabulary allocation and overlap across languages. In *Findings of ACL*, 2023. URL <https://arxiv.org/abs/2305.17179>.

- Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/0cde695b83bd186c1fd456302888454c-Paper-Conference.pdf.
- Haotian Liu, Chunyuan Li, Qingsong Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023a. URL <https://arxiv.org/abs/2304.08485>.
- Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023b. URL <https://aclanthology.org/2023.emnlp-main.153/>.
- Llama Team, AI @ Meta. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL <https://arxiv.org/abs/2407.21783>. Version dated July 23, 2024.
- Jonas F. Lotz, António V. Lopes, Stephan Peitz, Hendra Setiawan, and Leonardo Emili. Beyond text compression: Evaluating tokenizers across scales. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2025.
- Aleksandar Petrov, Emanuele La Malfa, Philip H. S. Torr, and Adel Bibi. Language model tokenizers introduce unfairness between languages. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL <https://arxiv.org/abs/2305.15425>.
- Libo Qin, Qiguang Chen, Yuhang Zhou, Zhi Chen, Yinghui Li, Lizi Liao, Min Li, Wanxiang Che, and Philip S. Yu. A survey of multilingual large language models. *Patterns*, 6(1):101118, 2025. doi: 10.1016/j.patter.2024.101118.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021. URL <https://proceedings.mlr.press/v139/radford21a.html>.
- Seamless Communication Team. Seamless: Multilingual expressive and streaming speech translation. In *NeurIPS 2023 Workshop on Seamless Communication*, 2023.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL <https://aclanthology.org/P16-1162/>.
- Yi Tay, Vinh Q. Tran, Sebastian Ruder, Jai Gupta, Hyung Won Chung, Dara Bahri, Zhen Qin, Simon Baumgartner, Cong Yu, and Donald Metzler. Charformer: Fast character transformers via gradient-based subword tokenization. In *International Conference on Learning Representations*, 2022. doi: 10.48550/arXiv.2106.12672. URL <https://arxiv.org/abs/2106.12672>. ICLR 2022 camera-ready; arXiv:2106.12672.
- Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. URL <https://arxiv.org/abs/1807.03748>.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL <https://arxiv.org/abs/1706.03762>.
- Linting Xue and et al. Byt5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 2022. URL <https://aclanthology.org/2022.tacl-1.17/>.
- Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mt5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of NAACL*, 2021. URL <https://aclanthology.org/2021.naacl-main.41/>.
- Atsuki Yamaguchi, Aline Villavicencio, and Nikolaos Aletras. An empirical study on cross-lingual vocabulary adaptation for efficient language model inference. *arXiv preprint arXiv:2402.10712*, 2024. Findings of EMNLP 2024.
- An Yang and Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

- Dongkeun Yoon, Joel Jang, Sungdong Kim, Seungone Kim, Sheikh Shafayat, and Minjoon Seo. Langbridge: Multilingual reasoning without multilingual supervision. In *Proceedings of ACL*, 2024. URL <https://aclanthology.org/2024.acl-long.405/>.
- Biao Zhang and Rico Sennrich. Root mean square layer normalization. *arXiv preprint arXiv:1910.07467*, 2019. URL <https://arxiv.org/abs/1910.07467>.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*, 2023. URL <https://arxiv.org/abs/2306.05685>.

We release the training and inference code at <https://github.com/rajansagarwal/llink>.

A Tokenization Analysis

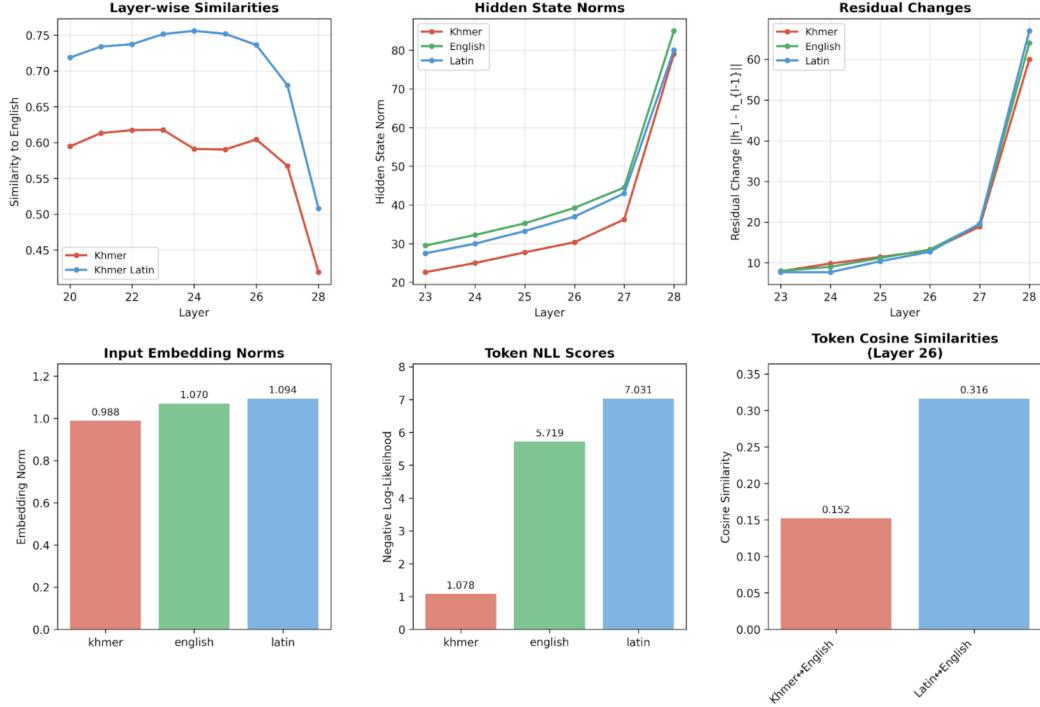


Figure 3: Analysis of fine-tuned representations with Khmer LLaMA 3.2 tokenization. The top three charts present layer-wise similarities, hidden state norms and residual changes. The bottom three charts present input embedding norms, token NLL scores and cosine similarities between Khmer, Khmer Latin transliteration and English translations of the same text.

LLaMA 3.2 shows disparate token-level NLL scores across languages: Khmer text exhibits a mean NLL of 1.078, while English and Latin transliterations achieve 5.719 and 7.031, respectively. This 5.3-fold advantage for Khmer represents an asymmetry in predictive difficulty. The substantially lower NLL for Khmer-tokenized text suggests that the model’s tokenizer, despite being primarily trained on Latin-script corpora, has developed efficient compression strategies for the Khmer script, which we note in sparser token sequences that align more naturally with the model’s learned distributional patterns.

The disparity in cosine similarity of the different scripts indicates that while Khmer tokenization produces sequences that are easier to predict, these sequences occupy more distant regions of the representation space relative to semantically equivalent English text. The Latin transliteration, despite a higher NLL, produces representations that share a greater geometric similarity with English, likely due to shared character-level and phonetic features.

Through the middle layers (20–26), we see progressive semantic alignment, where Khmer maintains cosine similarities with English between 0.59 and 0.62, while Latin achieves 0.72 to 0.76. The lower norm for Khmer embeddings suggests that Khmer tokens, on average, occupy less extreme positions in the initial embedding space, potentially reflecting the tokenizer’s treatment of the script as a more compact symbolic system.

B More LLINK Generations

Task	Content understanding — translate
Reference (En)	Good intentions for 2021?
Base	2021 (untranslatable Khmer)
Fine-tune	The 2021 Myanmar military coup.
LLINK	This suggests a positive outlook for the upcoming year.

Task	Content understanding — summarize
Reference (En)	The store sells books, magazines, and newspapers.
Base	Store sells items.
Fine-tune	The store has reading materials.
LLINK	The store offers books and publications.

Task	Q&A about foreign text (numeric weakness)
Reference (En)	The distance is 25 kilometers.
Prompt (En)	How far is it?
Base	(untranslatable Khmer; no answer)
Fine-tune	It is far away.
LLINK	The distance is approximately 10 miles.

Task	Content understanding — translate
Reference (En)	Parking is available on the second floor.
Base	(untranslatable Khmer; no answer)
Fine-tune	There is parking in the building.
LLINK	Cars go on the second floor.

Task	Content understanding — bullet pointify
Reference (En)	Requirements: valid ID, proof of address, application form.
Base	Requirements are (untranslatable Khmer)
Fine-tune	• ID • Successful • Form
LLINK	• Proper identification • Name • City

Task	Content understanding — summarize
Reference (En)	The course includes lectures, group work, and a final exam.
Base	(untranslatable Khmer)
Fine-tune	The course will test you on content.
LLINK	Students attend lectures and do projects and tests.

Task	Q&A about foreign text (numeric error)
Reference (En)	The building was constructed in 1985.
Prompt (En)	When was the building built?
Base	1985
Fine-tune	1985
LLINK	The building was built in 1980s

Task	Q&A about foreign text
Reference (En)	The fee is \$50 per person.
Prompt (En)	How much does it cost?
Base	(untranslatable Khmer; no answer)
Fine-tune	There is a fee of \$50.
LLINK	The cost is \$5 each.
