星辰之力 : An LLM-Driven Pipeline for Extremely Low-Resource English-to-Indic Translation — From Vocabulary Injection to Post-Training and Decoding

Lu Ruan Shaoying Jia Fan Ning Wei Wang
Zhengzhen Cai Fei Hu Heng Hu Chenzi Wang
China Telecom Corporation Limited Chongqing Branch, Chongqing, China
{ruanlu.cq, jiashaoying.cq, ningfan.cq, wangwei.cq, sazhengzhen.cq, Hufei4.cq, huh12.cq, wangchenzi.cq}@chinatelecom.cn

Abstract

We describe our team's submission to the WMT26 shared task on low-resource Indic language translation, covering five extremely low-resource English-to-X directions: Bodo, Karbi, Kokborok, Nagamese and Tagin. Because bilingual data for these pairs ranges from only about one thousand to fifteen thousand sentence pairs, we describe an *LLM-driven pipeline* in which large language models are used at every stage: to construct the bilingual resources and as the translation model itself. The pipeline has three components. First, a bilingual word-level vocabulary is constructed automatically, with a large language model (qwen3-max) performing the word-level alignment; the candidates are injected into the translation prompt as a "Helpful Vocabulary" block. Second, two large language models are post-trained with shared data and prompt formats: parameter-efficient LoRA fine-tuning of Qwen2.5-32B-Instruct (Qwen Team, 2025) and full-parameter fine-tuning of the translation-specialised Hunyuan-MT-7B (Tencent Hunyuan Team, 2025), each with an optional DPO stage. Third, inference-side decoding post-processing suppresses repetition and length degeneration, with a take-shorter combination of two decoding versions. On a held-out 80:20 split of the Tagin training data, vocabulary injection improves Tagin by $+7.09$ BLEU / -49.35 TER. On the WMT2025 gold test, Hunyuan full-parameter fine-tuning scores higher than Qwen LoRA on Bodo and Kokborok (Bodo ChrF 60.59 vs 48.72), though the two routes differ in base model as well as adaptation strategy; official WMT26 scores are in Table 10.

1 Introduction

Machine translation for the minority languages of North-East India—Bodo, Karbi, Kokborok, Nagamese and Tagin—is an extreme low-resource problem. In the WMT26 official training set the number of bilingual sentence pairs is tiny for every pair (Bodo 15,215; Tagin 5,343; Kokborok 2,269; Nagamese 1,998; Karbi 1,000). Orthographic conventions vary across these languages, and language-specific tokenisation resources are scarce; mainstream pretrained translation models also provide limited documented support for these pairs. Under such conditions a single modelling recipe rarely transfers across languages, and corpus-level automatic metrics are highly sensitive to a small number of pathological outputs such as runaway repetition and uncontrolled length.

Three structural properties compound the difficulty: heterogeneous target scripts (Bodo in Devanagari, the others romanised); small, stylistically narrow in-domain text, so large out-of-domain corpora can hurt; and short references (median $\approx$13 tokens on Bodo), so a single over-long or repetitive hypothesis disproportionately affects corpus-level BLEU, ChrF and TER.

We approach the problem from two complementary directions. The first fine-tunes the 32B instruction model Qwen2.5-32B-Instruct (Qwen Team, 2025) with LoRA (Hu et al., 2021), compensating for sparse language signals through parameter capacity and a custom vocabulary prompt; the second fully fine-tunes the translation-specialised model Hunyuan-MT-7B (Tencent Hunyuan Team, 2025). Although its published supported-language list does not include the five target languages, its translation-oriented training makes it a useful contrasting base to a general instruction model. Both routes share the same automatically constructed bilingual vocabulary and the same inference-side post-processing.

Our research question is which LLM-driven interventions actually move quality when only one to fifteen thousand sentence pairs exist; vocabulary injection, post-training, augmentation and decoding control are each evidence on it. The contributions are:

Proceedings of the Eleventh Conference on Machine Translation, pages 2360–2367
October 28-29, 2026 ©2026 Association for Computational Linguistics

- A dual-pipeline study contrasting parameter-efficient (Qwen-32B LoRA) and full-parameter (Hunyuan-7B) fine-tuning with shared data, prompt and evaluation settings; as the bases differ, it compares pipelines, not adaptation strategies.
- An automatically built, LLM-aligned bilingual word-level vocabulary injected into the prompt; a held-out ablation associates its use with +7.09 BLEU and −49.35 TER on Tagin (§5.2).
- A decoding-side treatment—repetition/length control and a per-sentence take-shorter combination of two decoding versions—that changes corpus-level metrics without retraining (Bodo BLEU 19.27→27.18, §5.5).
- An empirical analysis of when data augmentation helps and of the hyper-parameter sensitivity of DPO in low-resource MT, together with the official WMT26 results of the submitted systems (§5.7).

All code, training configurations, and submission scripts used in this work are publicly released.[1]

The system is an LLM-driven pipeline in which large language models both build the bilingual vocabulary and serve as the post-trained translation models, comprising three stages—vocabulary construction and prompt injection, post-training (SFT with an optional DPO stage), and inference-side post-processing—shown in Figure 1.

2 Task and Data

2.1 Task setup and official data

We participate in Category 2 (the extremely low-resource track) of the WMT26 shared task on low-resource Indic language translation (Pakray et al., 2026), the latest edition of an annual series (Pal et al., 2023; Pakray et al., 2024, 2025) that has also driven neural systems for related North-East Indian pairs (Kakum et al., 2023), and submit for the five English→X directions above. Development and analysis use the WMT2025 official gold test (Pakray et al., 2025) (2,000 sentences per pair across five domains), which is public for Bodo and Kokborok; the other pairs are compared with internal held-out splits and training logs. The vocabulary stage additionally uses the `dwyl/english-words` seed dictionary (370,100 entries) and the DashScope qwen3-max alignment API.

[1] https://github.com/opprash/wmt2026_indic_mt

Lang.	Source repository / address	Lines
Bodo	hf.co/datasets/alayaran/bodo_english_parallel	152,627
Kokborok	hf.co/datasets/{sdmy/kokborok, agnivamaiti/qa, doeltripura/english_kokborok}	12,585
Nagamese	hf.co/datasets/{agnivamaiti, GODCOMPLEX, xerces101}	16,311
Karbi	UC Berkeley eScholarship (escholarship.org); PDF → MinerU → LLM	603
Tagin	github.com/.../GinLish (public code repo)	855

Table 1: External open-source corpora used by the augmented variants, with their source repositories. HuggingFace paths are abbreviated `hf.co/datasets/`; braces group multiple repositories under one owner namespace.

Lang.	Ext.	Off.	Concat	Rem.	Final
Bodo	152,627	15,215	167,842	2,527	165,315
Kokborok	12,585	2,269	14,854	0	14,854
Nagamese	16,311	1,998	18,309	170	18,139
Karbi	603	1,000	1,603	39	1,564
Tagin	855	5,343	6,198	237	5,961

Table 2: Whole-line de-duplication for the merged (augmented) corpora.

2.2 External open-source augmentation

For the augmented variants we collect additional public parallel data (Table 1). Bodo, Kokborok and Nagamese come from multiple HuggingFace datasets (including the Google SMOL Kokborok data). The Karbi supplement (603 usable lines) is extracted by us from public Karbi literature on UC Berkeley eScholarship, converting PDFs to Markdown with MinerU and then extracting example pairs with a DashScope Qwen Flash model; the Tagin corpus comes from a public code repository. The external corpus is merged with the official corpus after whole-line de-duplication (Table 2), and the two-stage vocabulary pipeline (§3) is re-run on the merged data. Because external data is out-of-domain for some languages, augmentation helps some pairs and harms others (§5.4).

3 Custom Bilingual Vocabulary Injection

Our central low-resource enhancement is a custom bilingual word-level vocabulary, built automatically and injected into the translation prompt in two stages. Because the target languages lack reliable dictionaries and sub-word tokenizers, an in-domain word-level vocabulary supplies the lexical evidence a general model is missing, and prompt injection keeps the method identical across both pipelines.

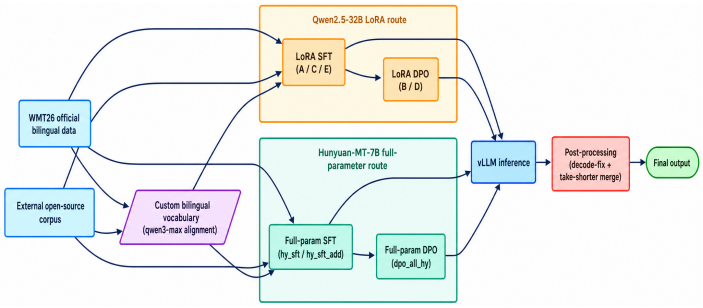

Figure 1: System overview. Official and external data feed a shared bilingual-vocabulary stage; two fine-tuning routes (Qwen-32B LoRA and Hunyuan-7B full-parameter, each with SFT/DPO variants) are decoded with vLLM and passed through decode-fix and the take-shorter merge,(A) LoRA SFT; (B) SFT → merge → DPO; (C) Augmented Data + LoRA SFT; (D) Pure LoRA DPO; (E) LoRA SFT (No Vocab Injection).

3.1 Stage 1 — sentence-level matching

Given 370,100 English seed dictionary entries, we scan every bilingual pair $(\mathrm{eng}, \mathrm{tgt})$ and, for each English token w in the dictionary, accumulate the target sentence into a map, producing a sentence-level index of every dictionary word and the pairs containing it (e.g. 13,688 hit words / 196,194 pairs on Bodo).

3.2 Stage 2 — LLM word-level alignment

For every word w with at least two collected pairs, we deterministically sample up to ten pairs and query the DashScope qwen3-max API for word-level alignment: given the English sentences and their target translations (all containing w), the model returns the corresponding target word(s) as a JSON list of candidates with descending probabilities summing to at most one. We keep candidates with probability ≥ 0.5 and show the top three at prompt time. Top-3 performed best in a preliminary Top-1/3/5 comparison and is used throughout. The resulting vocabularies contain 30,216 (Bodo), 8,255 (Nagamese), 6,503 (Kokborok), 1,811 (Karbi) and 1,697 (Tagin) aligned words under the augmented corpus. The alignment is checkpointed at the last processed word, so a run interrupted by API rate limiting or network failure resumes rather than re-querying all 13,688 Bodo words; a recovery pass re-queries words the content filter skipped.

3.3 Vocabulary injection into the input prompt

The candidates are written into the model's input text, with no decoder- or logit-level intervention. Into the official Hunyuan-MT-7B translation instruction we insert a "Helpful Vocabulary" block between the instruction line and the source sentence, one line per source word with its Top-3 candidates and probabilities (e.g. "`- announced: kepangdak (0.90) / adihin (0.60)`" for Karbi); it is omitted when no candidate is found, and Appendix A gives the exact template and a verbatim instance. The same format serves training and inference on both corpora, so the vocabulary's measured effect is not confounded by prompt differences. The candidates are soft evidence the model can override; hard lexical constraints are left to future work, as the alignments are noisy and enforcing them risks over-constraining generation here.

4 Systems and Post-Training

We run two parallel post-training pipelines, each adapting a base model by supervised fine-tuning (SFT) and, optionally, DPO, with several variants. All experiments use four H800 GPUs, bf16, a cosine schedule and cutoff length 512. The warmup ratio is 0.1 except for Hunyuan SFT, which uses 0.03. Training uses LlamaFactory v0.9.4 (Zheng et al., 2024) and inference uses vLLM (Kwon et al., 2023); vocabulary alignment uses the DashScope qwen3-max API. The two pipelines are complementary—the Qwen route adapts a 32B general model cheaply with LoRA (r=8, α=16; bs $4\times$ga 2), the Hunyuan route fully fine-tunes a smaller translation-specialised model (fa2; bs $2\times$ga 8), both under ZeRO-3—and Table 3 lists the resulting configurations.

4.1 Qwen2.5-32B LoRA SFT

Both SFT pipelines minimise the token-level negative log-likelihood of the reference translation y given the vocabulary-augmented source prompt x:

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\sum_{t} \log \pi_\theta(y_t \mid y_{<t}, x) \qquad (1)$$

Config	Data	Stage
qwen_sft	official	SFT (LoRA)
qwen_sft_add	+external	SFT (LoRA)
qwen_sft_merge_dpo	+pref.	SFT→merge→DPO
qwen_pure_dpo	+pref.	DPO only
hy_sft	official	SFT (full)
hy_sft_add	+external	SFT (full)
dpo_all_hy	+pref.	SFT→DPO (full)
take-shorter	—	post-hoc fusion

Table 3: Seven training configurations plus one post-hoc ensemble.

where x is the instruction with the injected Top-3 vocabulary block followed by the source sentence, y_t the t-th target token, and π_θ the model; the Hunyuan SFT (§4.3) uses the same objective. Qwen2.5-32B-Instruct is the general arm chosen to contrast Hunyuan-MT-7B: not translation-specialised, but supplying the multilingual pretraining and instruction following the vocabulary block needs; 32B is the largest checkpoint whose LoRA adaptation and vLLM inference fit our four-H800 budget. The smaller specialised base scoring higher hints that full fine-tuning of a translation-adapted model may suit extremely low-resource MT better than a rank-8 LoRA on a larger one, though the two factors are confounded here (§5.3). Qwen SFT uses LlamaFactory default LoRA hyper-parameters (r=8, α=16, dropout=0, lora_target=all), ZeRO-3 and three epochs. For each language the learning rate is swept over {5e-5, 1e-4, 2e-4, 5e-4} and selected by validation loss (5e-4 for Karbi/Tagin/Kokborok, 2e-4 for Nagamese/Bodo); the best adapter is reused as the DPO base.

4.2 Qwen LoRA DPO

We explore two DPO (Rafailov et al., 2023) variants: SFT→merge→DPO (the merged SFT checkpoint is the base) and pure DPO (preference training on the un-fine-tuned base). DPO uses $\beta = 0.1$, sigmoid loss, four epochs, and a sweep over {1e-4, 2e-4, 5e-4}; Qwen DPO runs under PyTorch DDP. DPO optimises the following preference objective over triples (x, y_w, y_l) drawn from a preference dataset $\mathcal{D}$ (Rafailov et al., 2023):

$$\mathcal{L}_{\text{DPO}}(\theta) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Big[\log\sigma\Big(\beta\log\tfrac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta\log\tfrac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}\Big)\Big], \quad (2)$$

where σ is the logistic function, $\beta = 0.1$ the preference temperature, y_w the chosen response, y_l the rejected response, and π_{ref} the frozen post-SFT reference policy.

4.3 Hunyuan-MT-7B full-parameter SFT

Each language is trained independently with a single learning rate (3e-5), full-parameter fine-tuning, Flash-Attention-2 (Dao et al., 2022) and DeepSpeed ZeRO-3 (Rajbhandari et al., 2020), for four epochs. The 7B full-parameter setting fits within memory on four 80GB GPUs under ZeRO-3, and four epochs suffice for the loss to converge on these corpora. The augmented variant (hy_sft_add) differs only in the training set, so differences between hy_sft and hy_sft_add are associated with the data change.

4.4 Hunyuan-MT-7B full-parameter DPO and preference data

Full-parameter DPO uses the hy_sft checkpoint as its base, with $\beta = 0.1$, sigmoid loss, learning rate 5e-6, four epochs, and ZeRO-3, which is required by our four-GPU memory budget because preference training evaluates both chosen and rejected responses and also requires reference-policy log probabilities. Preference data follows the ShareGPT-DPO schema: the chosen response is the official translation and the rejected response is produced by the post-SFT model via vLLM sampling (initial temperature 1.2, stepped up until it differs from the chosen one). Full-parameter DPO optimises the same objective (Eq. 2) with the reference policy π_{ref} set to the frozen hy_sft model and learning rate 5e-6.

4.5 Inference and decoding post-processing

Inference is multi-GPU data-parallel: each card runs an independent vLLM process and the driver partitions the input, avoiding tensor-parallel overhead; Qwen LoRA adapters are mounted directly. On the gold test about 7% of Hunyuan SFT predictions enter a repetition loop (maximum word repetition 27, maximum length 201 tokens against a reference median of 13). These long-tail samples alone drag down corpus BLEU/ChrF/TER. We apply two inference-side output fixes to anomalous outputs only, leaving clean outputs byte-identical: a post-hoc repeated-3-gram truncation and a length cap. This approximates but is not equivalent to re-decoding under such constraints. We further add a lightweight ensemble: sft and sft_test are two decoding versions of the same model, and for each sentence i we keep the shorter output, $\hat{y}_i = \arg\min_{y\in\{y_i^A, y_i^B\}} |y|$, where y_i^A, y_i^B are the two versions and $|\cdot|$ is length in tokens.

System	V	BLEU	MET.	ChrF	TER↓
sft_base	✗	5.78	0.291	26.90	144.20
sft_vocab	✓	12.87	0.352	34.61	94.85
dpo_vocab	✓	6.92	0.224	27.06	96.40
sft_dpo_vocab	✓	12.96	0.350	34.20	100.38

Table 4: Tagin 80:20 pre-experiment (Hunyuan full-parameter).V=vocabulary injected,ROUGE-L is omitted due to space constraints (Base: 39.05; Vocab: 45.18).

Direction / system	BLEU	MET.	ChrF	TER↓
Bodo Hy sft_test	24.89	0.520	60.59	76.35
Bodo Hy take-short+fix	27.18	0.516	61.50	64.13
Bodo Qwen LoRA sft	15.10	0.376	48.72	81.60
Kokb. Hy sft_add	8.74	0.235	33.53	89.39
Kokb. Qwen sft_add	5.72	0.208	31.16	94.86

Table 5: Hunyuan full-parameter vs Qwen LoRA (WMT2025 gold).

Task	Variant	BLEU	ChrF	TER↓
Bodo	sft	19.27	58.56	112.68
Bodo	sft_add	16.68	52.14	90.04
Kokb.	sft	2.07	27.60	138.07
Kokb.	sft_add	8.74	33.53	89.39

Table 6: Opposite effect of augmentation on Bodo (harmful) and Kokborok (helpful).

5 Experiments and Results

5.1 Evaluation setup

We report five official metrics: BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), ROUGE-L (Lin, 2004), ChrF (Popović, 2015) and TER (Snover et al., 2006) (lower is better). BLEU is reported using the language-specific tokenisation settings recorded by the experiment pipeline. In particular, the Bodo development tables use IndicNLP-pretokenised, lowercased sacreBLEU, whereas romanised-language runs use their recorded Latin-script evaluation setup. These values are therefore not uniformly 13a-tokenised and not comparable across tokenisation settings. The official WMT26 leaderboard additionally reports ChrF++, BERTScore (Zhang et al., 2020) and COMET (Rei et al., 2020), cited in §5.7.

5.2 Vocabulary ablation (Tagin 80:20)

To isolate the contribution of vocabulary injection we split the 5,343-line Tagin training set 80:20 and trained four Hunyuan variants (Table 4). Under this held-out setting, adding the vocabulary block is associated with +7.09 BLEU, +0.061 METEOR, +6.13 ROUGE-L, +7.71 ChrF and −49.35 TER relative to the baseline. The high TER of the base system (144.20) indicates length instability without the vocabulary; adding it reduces TER to 94.85. Pure DPO scores below the SFT systems on all metrics, and SFT→DPO is close to SFT with a small TER increase consistent with the length increase observed at scale (§5.5).

5.3 Full-parameter Hunyuan vs LoRA Qwen

On the WMT2025 gold test the full-parameter Hunyuan pipeline consistently leads the LoRA Qwen pipeline (Table 5). On Bodo, Hunyuan sft_test reaches ChrF 60.59 vs the Qwen SFT 48.72 (+11.87) and recorded BLEU 24.89 vs 15.10; on Kokborok the sft_add comparison gives ChrF 33.53 vs 31.16 and recorded BLEU 8.74 vs 5.72. Plausible contributing factors include the translation-specialised objective of the Hunyuan base, full-parameter updates versus the rank-8 LoRA adaptation; and the feasibility of full fine-tuning of a 7B model with ZeRO-3 on four 80GB GPUs. The two routes differ in base model and pretraining objective as well as adaptation strategy, so this compares complete pipelines rather than attributing the gap to the post-training method; a controlled attribution needs both strategies on one common base.

5.4 Language dependence of data augmentation

The same augmentation strategy has opposite effects on Bodo and Kokborok (Table 6). On Bodo, adding 152,627 lines of mostly out-of-domain corpus decreases BLEU and ChrF (BLEU 19.27→16.68), although TER improves. On Kokborok, whose 2,269 in-domain lines are underfit, the same augmentation increases every metric (BLEU 2.07→8.74). Augmentation is therefore not uniformly beneficial: its effect is associated with the size and domain match of the in-domain data.

5.5 Decoding post-processing and ensembling (Bodo)

Length and repetition control has a large effect on the Bodo metrics. The plain SFT output has a median length equal to the reference (13) but a mean of 20.6, while the sft_dpo run shows severe degeneration (median 89, 74.7% over-length, maximum repetition 467). Applying decode-fix to the SFT output raises recorded BLEU from 19.27 to 27.52 and drops TER from 112.68 to 71.07 with no retraining. The take-shorter ensemble (sft_test contributes 1,375 and sft 625 of 2,000 sentences) combined with decode-fix gives our best Bodo development system: BLEU 27.18, ChrF 61.50, TER 64.13 (Table 7).

Bodo system	BLEU	MET.	ChrF	TER↓	len
sft (baseline)	19.27	0.537	58.56	112.68	1.51
sft_test	24.89	0.520	60.59	76.35	1.14
sft + decode-fix	27.52	0.538	62.73	71.07	1.07
sft_test + fix	26.08	0.515	60.91	68.84	1.04
take-shorter	26.88	0.520	61.53	66.56	1.03
take-short + fix	27.18	0.516	61.50	64.13	0.98

Table 7: Decoding post-processing and take-shorter ensemble on Bodo (WMT2025 gold).

System (en→Bodo)	BLEU	MET.	ChrF	TER↓
DoDS-IITPKD (contr.)	24.97	0.519	67.81	51.50
DoDS-IITPKD (prim.)	24.45	0.513	67.71	51.84
JU-NLP (prim.)	19.71	0.455	62.47	64.97
Ours: take-short+fix	27.18	0.516	61.50	64.13
Ours: sft_test+fix	26.08	0.515	60.91	68.84

Table 8: en→Bodo: development comparison with WMT2025.

5.6 Comparison with WMT2025 baselines

As a development-time check we compare our best systems with WMT2025 same-task submissions on the identical 2025 gold test (Tables 8 and 9). Our recorded BLEU values are 27.18 on Bodo and 8.74 on Kokborok, whereas the published WMT2025 table reports 24.97 and 6.997 for its strongest systems. As the WMT2025 findings paper gives no sacreBLEU signature, this comparison is descriptive rather than proof of superiority. On Bodo the ChrF and TER of the strongest 2025 systems remain better than ours.

5.7 Official WMT26 final results

For each English→X direction, two systems were submitted. For Bodo these are the take-shorter+decode-fix combination and the SFT baseline; for Kokborok, Karbi, Nagamese and Tagin they are the sft_add system and the SFT baseline. Table 10 reports the official WMT26 final scores on the six metrics released by the organisers (Pakray et al., 2026). The 2026 test set differs from the 2025 gold test used above, and absolute scores are lower for all participating systems; for reference, the highest en→Bodo BLEU in the official results is 17.34. The two submissions diverge most on Kokborok, where the augmented contrastive1 system raises BLEU from 3.44 to 6.51 over the SFT baseline, whereas on Bodo, Nagamese and Tagin the two stay within about 0.6 BLEU. Nagamese attains the highest COMET (62.58) and METEOR (47.30), while Karbi and Tagin remain hardest in absolute terms, consistent with their smaller and more orthographically distinctive training sets.

System (en→Kokb.)	BLEU	MET.	ChrF	TER↓
ANVITA (contr.)	6.997	0.300	38.08	76.26
ANVITA (prim.)	1.756	0.107	18.58	104.04
Ours: Hy sft_add	8.74	0.235	33.53	89.39
Ours: Qwen sft_add	5.72	0.208	31.16	94.86

Table 9: en→Kokborok: development comparison with WMT2025.

Dir.	Sys.	BLEU	MET.	TER↓	ChrF++	BERT	COMET
Bodo	c1	12.12	30.74	74.20	44.98	83.15	60.29
	c2	12.46	31.06	75.49	45.12	83.15	60.27
Kokb.	c1	6.51	24.55	85.91	31.32	73.82	58.92
	c2	3.44	15.86	103.06	26.44	70.76	57.11
Karbi	c1	3.03	19.57	93.37	28.45	71.00	56.49
	c2	2.99	18.38	94.11	27.30	70.36	55.79
Naga.	c1	12.35	46.94	98.15	47.51	78.43	62.04
	c2	12.90	47.30	97.04	47.31	78.81	62.58
Tagin	c1	4.84	16.77	95.42	23.95	71.77	56.45
	c2	4.75	16.67	96.85	23.85	71.68	56.42

Table 10: Official WMT26 final test scores of the submitted systems (Category 2, English→X; c1/c2 = contrastive1/2), as released by the organisers.

6 Discussion

Three observations stand out. First, the full-parameter Hunyuan pipeline scored higher than the LoRA Qwen pipeline on Bodo and Kokborok; as the bases also differ, this is a pipeline-level, not a strategy-level, conclusion. Second, prompt-injected lexical information gave the largest observed ablation gain and reduced length instability. Third, degenerate over-long outputs were the most frequent failure mode, and inference-side control recovered part of the loss without retraining. DPO was hyper-parameter-sensitive (one Bodo run reached length ratio 9.16 and BLEU 0.81); a smaller learning rate (5e-6) and length regularisation helped, but the gain over SFT was small—plausibly because the preference triples are few and the chosen/rejected responses often similar—so DPO is not used in the submitted systems.

7 Conclusion

We presented an LLM-driven pipeline for extremely low-resource English-to-Indic translation, combining full-parameter Hunyuan fine-tuning, prompt-injected bilingual vocabulary, and inference-side length/repetition control. Official WMT26 scores appear in Table 10. Future work includes curriculum DPO, quality-weighted augmentation, and vocabulary-constrained decoding.

8 Acknowledgments

We thank the WMT26 Indic MT organizers for the data and evaluation framework, and China Telecom Corporation Limited Chongqing Branch for computing resources.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for MT*.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*.

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

Nabam Kakum, Sahinur Rahman Laskar, Koj Sambyo, and Partha Pakray. 2023. Neural machine translation for limited resources English–Nyishi pair. *Sādhanā*, Springer.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of SOSP*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out, ACL Workshop*.

Partha Pakray, Santanu Pal, Advaitha Vetagiri, Reddi Krishna, Arnab Kumar Maji, Sandeep Dash, Lenin Laitonjam, Lyngdoh Sarah, and Riyanka Manna. 2024. Findings of the WMT 2024 shared task on low-resource Indic languages translation. In *Proceedings of the Ninth Conference on Machine Translation*, pages 654–668.

Partha Pakray, Reddi Mohana Krishna, Santanu Pal, Advaitha Vetagiri, Sandeep Kumar Dash, Arnab Kumar Maji, Saralin A. Lyngdoh, Lenin Laitonjam, Anupam Jamatia, Koj Sambyo, Ajit Das, and Riyanka Manna. 2025. Findings of the WMT 2025 shared task on low-resource Indic languages translation. In *Proceedings of the Tenth Workshop on Machine Translation (WMT)*, pages 532–553.

Partha Pakray, Santanu Pal, Advaitha Vetagiri, Kshetrimayum Boynao Singh, Sandeep Kumar Dash, Arnab Kumar Maji, Saralin A. Lyngdoh, Lenin Laitonjam, Anupam Jamatia, Ajit Das, Sunita Warjri, Uzzal Sharma, Koj Sambyo, and Riyanka Manna. 2026. Findings of the WMT 2026 shared task on low-resource Indic languages translation. In *Proceedings of the Eleventh Workshop on Machine Translation (WMT), EMNLP 2026*, Budapest, Hungary.

Santanu Pal, Partha Pakray, Sahinur Rahman Laskar, Lenin Laitonjam, Vanlalmuansangi Khenglawt, Sunita Warjri, Pankaj Kundan Dadure, and Sandeep Kumar Dash. 2023. Findings of the WMT 2023 shared task on low-resource Indic language translation. In *Proceedings of the Eighth Conference on Machine Translation (WMT)*, pages 682–694.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Maja Popović. 2015. chrF: Character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation (WMT)*.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation (WMT)*.

Qwen Team. 2025. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. ZeRO: Memory optimizations toward training trillion parameter models. In *Proceedings of SC*.

Ricardo Rei, Craig Stewart, Ana C. Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of EMNLP*.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of AMTA*.

Tencent Hunyuan Team. 2025. Hunyuan-MT technical report.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *Proceedings of ICLR*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, and Yongqiang Ma. 2024. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of ACL, System Demonstrations*.

A Vocabulary-injection prompt

Training and inference use the same instruction, stored in the LlamaFactory Alpaca format with an empty `input` field. The injected vocabulary block sits between the instruction line and the source sentence, and is omitted when no candidate passes threshold. The box below is one training instance from the Tagin set, copied unmodified, with the injected block in bold; the reference translation

(the `output` field) is `no achvlv bv ajin kama do`. `Targin` is the language identifier string emitted by our data-generation script.

```
Translate the following segment into Targin,
without additional explanation.

Helpful Targin vocabulary for words in this
sentence:
- very: achvlv (0.85) / achlvv (0.65)
- lonely: ajin kama (0.90) / ajen kama (0.85)
/ ajin kaama (0.80)

You're very lonely
```