

NLP-MT_Amrita: A RL-Inspired GRPO based Agentic RAG Framework for Manipuri and Bodo

^aKuruva Divya Sree, ^aDeepa Gupta

^aDepartment of Computer Science and Engineering, Amrita School of Computing, Amrita Vishwa Vidyapeetham, Bengaluru, India.

divyasree6898@gmail.com , g_deepa@blr.amrita.edu

Abstract

Low-resource Indic machine translation (MT) remains challenging due to limited parallel corpora, linguistic diversity, and contextual ambiguities that often result in inaccurate translations and hallucinations. This study proposes AgenticRAG-GRPO, an RL-inspired retrieval-augmented translation framework that combines semantic retrieval with GRPO-inspired reward-guided candidate selection for context-aware English↔Indic MT. The framework dynamically retrieves relevant translation contexts and ranks candidate translations using a reward function based on SacreBLEU and BERT score. Experiments were conducted on the WMT 2026 Low-Resource IndicMT Shared Task for English↔Manipuri (Mni) and English↔Bodo (Brx) using IndicTrans2, NLLB-200 Distilled 600M, and Sarvam-Translate. The proposed study achieved s performance, obtaining BLEU scores up to 32.36, METEOR up to 67.01, BERTScore up to 95.04, and COMET up to 80.51 across the languages, while improving semantic faithfulness and reducing hallucinations through retrieval-guided generation. The framework also maintained inference efficiency without requiring task-specific model fine-tuning. These results demonstrate that AgenticRAG-GRPO provides an effective and scalable solution for low-resource English-Indic MT.

1 Introduction

This Machine Translation (MT) has witnessed remarkable progress with the advancement of multilingual neural machine translation (MNMT) and large language models (LLMs). However, achieving reliable translation quality for low-resource Indic languages remains a significant challenge due to the limited availability of parallel

corpora, rich morphological structures, diverse scripts, and domain-specific linguistic variations. These challenges often lead to inadequate contextual understanding, reduced translation quality, and hallucinated outputs, particularly when translating between English and under-resourced Indian languages such as Manipuri and Bodo [1, 2].

Recent studies have explored several strategies to improve translation quality for low-resource languages. Preference optimization techniques have been shown to improve translation generation by aligning model outputs with quality-aware reward functions [3], while retrieval-based example selection has demonstrated that incorporating semantically similar translation examples during inference can substantially improve contextual accuracy without requiring additional model training [4]. More recently, retrieval-augmented generation (RAG) frameworks have been extended to MT, where hierarchical and syntax-aware retrieval mechanisms further improve translation consistency by providing richer contextual information to the translation model [5]. These advances indicate that combining retrieval with translation models can effectively address limitations associated with conventional zero-shot and fine-tuning approaches.

Despite these advances, existing retrieval-based translation systems generally rely on fixed retrieval parameters and static decoding strategies, which may not be suitable for sentences with varying linguistic complexity. Furthermore, most approaches do not explicitly optimize the retrieved translation candidates before final generation, limiting their ability to consistently select the most semantically faithful translation.

To address these limitations, this paper proposes an RL-inspired Agentic Retrieval-Augmented Generation (AgenticRAG-GRPO) framework for

low-resource English↔Indic MT. The main contributions of this work are as follows:

- A novel RL-inspired AgenticRAG-GRPO framework is proposed for low-resource English↔Indic MT, integrating dynamic retrieval, adaptive decoding, and a GRPO-based reward function for translation candidate selection.
- The proposed framework is evaluated on the WMT 2026 Low-Resource IndicMT shared task for English↔Manipuri and English↔Bodo using IndicTrans2, NLLB-200 Distilled 600M, and Sarvam-Translate. Sentence embeddings are generated using the multilingual E5-Large model, while Milvus-Lite is employed as the vector database for efficient semantic retrieval.
- Extensive experiments demonstrate that the proposed retrieval-guided framework improves translation quality while reducing hallucinations through context-aware retrieval and reward-guided candidate selection, achieving competitive performance across multiple evaluation metrics on low-resource Indic language pairs.

The rest of the paper is organized as follows. Section 2 reviews the related work, Section 3 presents the proposed methodology and experimental setup, Section 4 discusses the experimental results, and Section 5 concludes the paper.

2 Literature Survey

Recent advances in multilingual neural machine translation (MNMT) have significantly improved translation quality for many languages; however, low-resource Indic languages continue to face challenges due to limited parallel corpora, rich morphological characteristics, and script diversity. A comprehensive survey by Sheshadri et al. [1] highlighted these challenges and summarized the evolution of neural machine translation for Indic languages. To promote standardized evaluation, the WMT 2025 and WMT 2026 Low-Resource Indic Machine Translation Shared Tasks introduced benchmark datasets and evaluation protocols, encouraging the development of robust translation systems for under-resourced Indian languages [6, 2]. Large multilingual translation models such as NLLB [7] and IndicTrans2 [8] have substantially improved multilingual translation quality through large-scale pre-training, while continual

knowledge transfer [9], language-specific fine-tuning using LoRA [10], and transformer-based transfer learning [11] have further enhanced adaptation for resource-constrained languages. Several studies have also focused on specific low-resource language pairs, including English-Nyishi [12], Manipuri and Sindhi [13], and Bodo [14], demonstrating the growing interest in improving translation performance for under-represented Indic languages.

More recently, retrieval-based approaches have emerged as an effective alternative to conventional fine-tuning by incorporating relevant contextual examples during inference. Zebaze et al. [4] demonstrated that similarity-based in-context example selection improves translation quality for low-resource languages, while SH-RAG introduced syntax-aware hierarchical retrieval to better handle complex sentence structures during translation [5]. The adoption of efficient vector databases has further enabled scalable semantic retrieval, making retrieval-augmented translation systems computationally practical [15]. Building upon these developments, retrieval-augmented frameworks have also been explored for domain-specific English-Hindi translation [16] and agentic retrieval for English-to-Indic MT [17], illustrating the effectiveness of retrieval-guided generation in improving contextual translation quality.

In parallel, preference optimization has gained increasing attention for aligning translation outputs with quality-aware reward functions. Nair et al. [3] proposed a hypergeometric-gamma reward mechanism for improving low-resource Indic language translation, whereas Direct Preference Optimization (DPO) guided by COMET quality estimation has demonstrated improved translation quality without explicit reinforcement learning [18].

Overall, recent advances in multilingual translation, RAG, and preference optimization have improved the quality of low-resource MT. However, challenges such as hallucinations, limited parallel data, and fixed retrieval strategies still affect translation performance. These limitations motivate the proposed RL-inspired AgenticRAG-GRPO framework, which combines adaptive retrieval and reward-guided candidate selection to generate more accurate and contextually relevant translations for low-resource Indic languages.

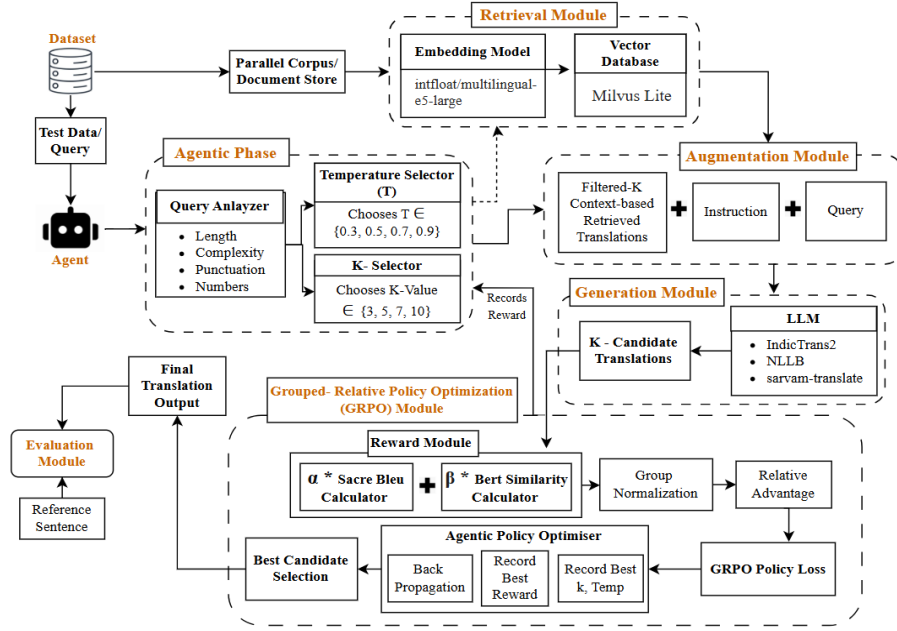


Figure 1. Architecture for the proposed RL inspired Agentic RAG based MT with GRPO for low resource languages.

3 Methodology

The proposed RL-inspired AgenticRAG-GRPO framework integrates semantic retrieval, dynamic parameter selection, multilingual translation, and GRPO-based reward-guided candidate selection for low-resource English↔Indic MT. The overall workflow of the proposed framework is illustrated in Figure 1, and the complete procedure is presented in Algorithm Table 3.

3.1 Document store and Test set

The proposed framework is evaluated using the resources provided for the WMT26 Indic Machine Translation Shared Task. Experiments focus on two low-resource Indic languages, Manipuri (Mni) in Bengali script (mni_Beng) and Bodo (Brx) in Devanagari script (brx_Deva) with English (En) in Latin (eng_Latn) script, covering both English-to-Indic and Indic-to-English translation directions. The document store is constructed from the official parallel corpora released for the shared task. Specifically, the Bodo corpus contains 15,215 parallel sentence pairs, while the Manipuri corpus comprises 23,687 parallel sentence pairs. These sentence pairs serve as the knowledge source for the proposed translation framework.

For evaluation, the official IndicMT 2026 test sets are employed. Each translation direction contains 1,000 Source (Src) sentences with their corresponding Target (Tgt) translations, enabling a

Dataset	Language Pair	#Sent Pairs	Avg. Src Length	Avg. Tgt Length
Document Store	En ↔ Mni	23,687	17.84	15.09
	En ↔ Brx	15,214	15.00	13.47
Test Set	En ↔ Mni	1,000	11.70	11.95
	En ↔ Brx	1,000	11.70	11.48

Table 1: Statistics of the WMT26 IndicMT dataset.

En: English, **Mni:** Manipuri, **Brx:** Bodo.

comprehensive evaluation across both lexical and semantic translation quality metrics. Table 1 presents the corpus statistics, including the number of parallel sentence pairs and the average sentence lengths of the Src and Tgt languages.

3.2 Retrieval Module

The retrieval module is designed to provide semantically relevant translation examples from the document store for each incoming Src sentence. During the indexing stage, the Src-side sentences of the parallel corpus with sentence-level chunking, are encoded into dense semantic representations using the ‘*intfloat/multilingual-e5-large*’ embedding model. The generated embeddings, together with their corresponding Src and Tgt sentence pairs, are stored in a Milvus Lite vector database, enabling efficient similarity-based retrieval of 1024 embedding dimension.

Sentence Characteristics	Selected Retrieval Depth (K)	Decoding Temperature (T)
Very short or numerical queries	3	0.1
Moderate-length sentences	5	0.7
Long or lexically complex sentences	7	0.7
Highly expressive or punctuation-rich sentences	10	0.9
Otherwise	5	0.3

Table 2: Agent Decision Policy for Dynamic Retrieval and Generation

Milvus Lite: is a lightweight, embedded version of the open-source Milvus vector database [19] that supports efficient storage, indexing, and similarity search of high-dimensional vector embeddings, enabling fast semantic retrieval without requiring a separate server deployment.

Given an input query sentence, the same embedding model is used to generate its dense vector representation. An Inner Product (IP) similarity search is then performed over the indexed document store to retrieve the top- K semantically similar sentence pairs. These retrieved examples provide contextual information that helps guide the translation model towards more accurate and contextually appropriate translations. The value of K is not fixed throughout the framework. Instead, it is dynamically determined by the proposed AgenticRAG module according to the characteristics of the input sentence. The retrieved sentence pairs are then passed to the subsequent augmentation module, where they are incorporated into the translation prompt for candidate generation.

3.3 Agentic Query Analysis

Different input sentences require varying amounts of contextual information during retrieval and different levels of exploration during translation generation. Using fixed retrieval and decoding parameters for all queries may either introduce unnecessary context for simple sentences or provide insufficient information for more complex ones. To address this limitation, the proposed agent employs a set of lightweight heuristic rules based on readily observable sentence characteristics, including sentence length, lexical complexity, punctuation density, and the

```

You are an expert translator.
Translate the following <Source Language> sentence to <Target Language>
(written in <Target Script>, if applicable).
Output ONLY the <Target Language> translation, nothing else.

Retrieved Context:
Source: <Retrieved Source Sentence 1>
Target: <Retrieved Target Sentence 1>

...

Source: <Retrieved Source Sentence K>
Target: <Retrieved Target Sentence K>

Input:
<Query Sentence>

Translation:

```

Figure 2: Prompt Template used for the proposed framework.

presence of numerical expressions. As summarized in Table 2, these heuristics were empirically formulated through experimental observations to dynamically determine appropriate retrieval depth (K) and decoding temperature (T), enabling the framework to balance translation quality and computational efficiency across various sentence types.

3.4 Retrieval-Augmented Translation

For every translation task, the retrieved bilingual examples and the input query are combined into a unified prompt before being passed to the translation model. The prompt explicitly specifies the Src and Tgt languages, ensuring that the generated output is restricted to the desired translation direction. The prompt template is given in Figure 2. The placeholders <Source Language>, <Target Language>, and <Target Script> are dynamically instantiated according to the translation direction. This prompt structure enables the translation models to align semantically similar bilingual examples while producing only the translated sentence without additional explanations.

3.5 RL-Inspired GRPO Optimization

The proposed framework employs an RL-inspired Group Relative Policy Optimization (GRPO) strategy to identify the best translation among multiple generated candidates. For each input sentence, the agent first determines suitable retrieval and decoding parameters, following which multiple translation candidates are generated using different retrieval depths and temperature parameters. Each candidate is then evaluated using a reward function that combines SacreBLEU and BERT Similarity, providing a balanced assessment of both lexical accuracy and

Input	• D_{DS}: Parallel training corpus containing source-target sentence pairs Document Store (DS) • D_{test}: Source-language test sentences • D_{ref}: Reference target sentences for reward computation • α, β: Reward balancing coefficients ($\alpha = \beta = 0.5$ by default)
Output	• Optimized translations τ^* • Performance logs containing selected retrieval size (k^*), Temperature (T^*), and reward scores.
Stage 1	Vector Database Initialisation 1. Initialize: ◦ Embedding Encoder E ◦ Vector Database VDB ◦ Translation Model M 2. For each batch (s_i, t_i) in D_{DS}: a. Generate dense embeddings: $e_i = E(s_i)$ b. Store (e_i, s_i, t_i) in VDB where s_i is source and t_i is target. Create an index on VDB using Inner Product (IP) similarity.
Stage 2	Dynamic Parameter Selection For each source sentence $q \in D_{test}$: 1. Analyse sentence length and complexity. 2. Select initial retrieval and decoding parameters where opt is optimal: Case 1: If q is long or complex: $k_{opt} = 10, T_{opt} = 0.6$ Case 2: If q is very short: $k_{opt} = 3, T_{opt} = 2$ Case 3: Otherwise: $k_{opt} = 5, T_{opt} = 4$ Clear heuristics defined in Table 1. 3. Define exploration sets: $K_{set} = \{k_{set}, 5\}$ $T_{set} = \{T_{set}, 4\}$
Stage 3	Candidate Generation with Recovery Mechanism For each combination $k \in K_{set}$ and $T \in T_{set}$ do: 1. Retrieve the top-k nearest source sentences and their translations from VDB. 2. Generate a translation candidate: $\tau = M(q, T)$ 3. Fallback Mechanism: If τ is empty or null then: $\tau \leftarrow t_{top1}$ where t_{top1} is the target sentence associated with the highest-ranked retrieved example. 4. Store (τ, k, T) for subsequent evaluation.
Stage 4	Reward-Based Optimization (GRPO Selection) For each candidate translation τ: 1. If a reference sentence $r \in D_{ref}$ exists: Compute lexical quality using SacreBLEU: $R_{BLEU} = \text{SacreBLEU}(\tau, r)$ Compute semantic similarity using BERTScore: $R_{BERT} = \text{BERTSimilarity}(\tau, r) \times 100$ Calculate the final reward: $\text{Reward}(\tau) = \alpha R_{BLEU} + \beta R_{BERT}$ 2. Otherwise, assign a fallback reward based on translation length: $\text{Reward}(\tau) = f(\text{Length}(\tau))$

	3. Select the candidate with the highest reward: $\tau^* = \arg \max_{\tau} \text{Reward}(\tau)$ 4. Record the corresponding parameters: (k^*, T^*) 5. Save $(\tau^*, k^*, \text{Temp}^*, \text{Reward}(\tau^*))$ to the performance log.
Final Output	Return: $\{\tau_1^*, \tau_2^*, \dots, \tau_n^*\}$ along with the corresponding optimal retrieval parameters, optimal Temperature, and reward scores for all test sentences.

Table 3: Detailed Algorithm on Optimized RAG with GRPO Translation Framework

310

311 semantic preservation. The candidate with the
312 highest reward is selected as the final translation.
313 The complete end-to-end workflow of the
314 proposed framework, including vector database
315 initialization, dynamic parameter selection,
316 candidate generation, reward computation, and
317 final translation selection, is summarized in
318 Algorithm presented in Table 3.

319 3.6 Experimental Setup

320 The study is conducted on the Kaggle platform
321 using Python 3.11 with GPU acceleration. The
322 implementation utilized the PyTorch and Hugging
323 Face Transformers libraries. The experiments are
324 run on an NVIDIA Tesla P100 GPU (16 GB
325 VRAM) with 2 vCPUs and 13 GB RAM.

326 3.6.1. Models Used

327 **IndicTrans2:** Introduced by AI4Bharat in 2024
328 [8], IndicTrans2 is a 1B parameter encoder-
329 decoder transformer designed specifically for
330 Indic MT. It supports high-quality translation
331 between English and multiple Indic languages
332 through dedicated English-to-Indic
333 (*‘ai4bharat/indictrans2-en-indic-1B’*) and Indic-
334 to-English (*‘ai4bharat/indictrans2-indic-en-1B’*)
335 models.

336 **NLLB-200 Distilled 600M (NLLB-600M):**
337 Introduced by Meta AI in 2022 [7], it is a 600M
338 parameter encoder-decoder transformer
339 supporting translation across 200 languages,
340 including several low-resource Indic languages. It
341 is distilled from the 54B-parameter NLLB model,
342 providing efficient multilingual translation with
343 lower computational requirements.
344 (*‘facebook/nllb-200-distilled-600M’*)

345 **Sarvam-Translate [20]:** (*‘sarvamai/sarvam-
346 translate’*) Developed by Sarvam AI, Sarvam-
347 Translate is a multilingual encoder-decoder

348 translation model optimized for Indian languages.
349 It is designed for efficient and high-quality
350 translation across multiple Indic language pairs
351 while offering faster inference suitable for
352 practical deployment.

353 3.6.2. Evaluation Metrics

354 The translation quality is evaluated using a
355 combination of lexical, character-level, edit-
356 based, and semantic evaluation metrics. BLEU (↑)
357 is employed to measure -gram overlap between
358 the generated and reference translations, while
359 METEOR (↑) considers synonymy, stemming,
360 and word alignments to provide a more
361 linguistically informed evaluation. Translation
362 Edit Rate (TER↓) quantifies the minimum number
363 of edits required to transform the generated
364 translation into the reference, where lower values
365 indicate better translation quality. chrF++ (↑)
366 evaluates character and word-gram similarity,
367 making it particularly suitable for
368 morphologically rich languages. To assess
369 semantic preservation, BERTScore (↑) computes
370 contextual embedding similarity between
371 candidate and reference translations, whereas
372 COMET (↑) provides a neural evaluation metric
373 trained on human translation judgments, offering
374 a comprehensive assessment of overall translation
375 quality. Also, sacreBLEU (↑) has been used for
376 Reward calculation. Together, these metrics
377 provide a balanced evaluation of lexical accuracy,
378 semantic faithfulness, and translation fluency.

379 3.7 Results and Analysis

380 This section presents the experimental results and
381 analysis of the proposed AgenticRAG-GRPO
382 framework, including quantitative evaluation,
383 reward analysis, dynamic retrieval parameter
384 selection, and qualitative translation examples.

Translation Direction	Model	BLEU $\uparrow$	METEOR $\uparrow$	TER $\downarrow$	chrF++ $\uparrow$	BERT Score $\uparrow$	COMET $\uparrow$
En $\rightarrow$ Mni	IndicTrans2	10.29	17.38	88.27	38.88	84.57	68.01
	NLLB-600M	9.65	18.50	85.03	38.67	83.98	67.44
	Sarvam-Translate	21.50	0.48	57.80	45.20	83.10	0.76
Mni $\rightarrow$ En	IndicTrans2	0.28	8.80	123.41	16.21	84.31	47.85
	NLLB-600M	32.36	67.01	54.87	62.68	95.04	80.51
	Sarvam-Translate	23.61	59.02	68.42	55.59	93.39	75.02
En $\rightarrow$ Brx	IndicTrans2	14.01	34.40	76.09	48.00	84.32	61.58
	NLLB-600M	-	-	-	-	-	-
	Sarvam-Translate	19.20	0.42	62.40	41.50	80.50	0.72
Brx $\rightarrow$ En	IndicTrans2	0.45	8.45	106.92	16.15	85.37	51.28
	NLLB-600M	3.04	18.55	116.70	24.08	81.63	42.99
	Sarvam-Translate	2.10	11.11	105.66	16.71	78.50	35.83

Table 4: Translation performance of the proposed AgenticRAG-GRPO framework on the WMT26 IndicMT test sets.

3.7.1. Performance comparison of the proposed framework across different translation directions

Translation performance of the proposed AgenticRAG-GRPO framework across four translation directions as shown in Table 4. For En $\rightarrow$ Mni, Sarvam-Translate achieved the highest lexical performance with a BLEU score of 21.50 and the lowest TER (57.80), whereas IndicTrans2 obtained the highest BERTScore (84.57) and COMET (68.01), indicating stronger semantic preservation. NLLB-600M achieved the highest METEOR score (18.50). A similar trend was observed for En $\rightarrow$ Brx, where Sarvam-Translate achieved the highest BLEU (19.20) and lowest TER (62.40), while IndicTrans2 produced the highest BERTScore (84.32) and COMET (61.58), suggesting stronger semantic consistency.

For the reverse translation direction, NLLB-600M achieved the best overall performance for Mni $\rightarrow$ En, obtaining 32.36 BLEU, 67.01 METEOR, 54.87 TER, 62.68 chrF++, 95.04 BERTScore, and 80.51 COMET. For Brx $\rightarrow$ En, IndicTrans2 achieved the highest BERTScore (85.37) and COMET (51.28), while Sarvam-Translate obtained the lowest TER (105.66). Since NLLB-200 Distilled 600M does not officially support Bodo as a Tgt language, a fair comparison for the En $\rightarrow$ Brx translation is not possible across all models. Therefore, the comparative analysis focuses on the Brx $\rightarrow$ En, where all evaluated systems can generate English translations under a

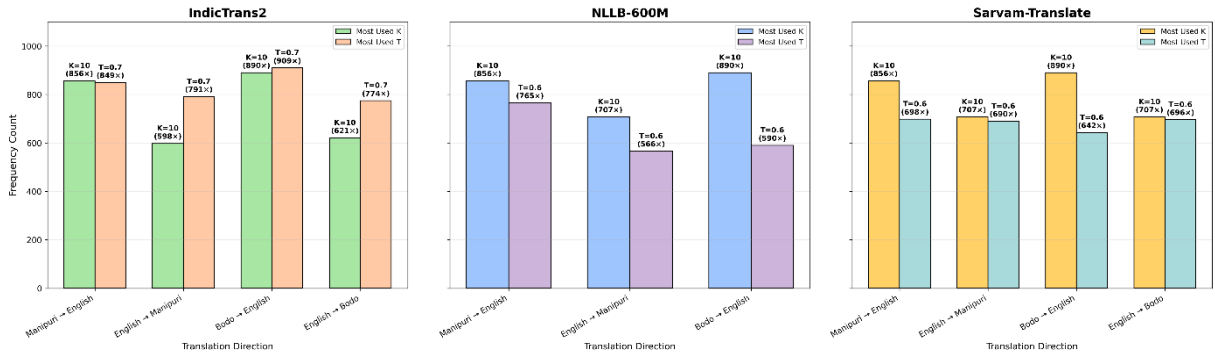
common evaluation.

Overall, Sarvam-Translate is the strongest model for En $\rightarrow$ Mni and En $\rightarrow$ Brx, NLLB-600M achieves the best performance for Mni $\rightarrow$ En, and IndicTrans2 performs best for Brx $\rightarrow$ En, while consistently demonstrating strong semantic preservation through higher BERTScore and COMET values.

Additionally, Average inference time (HH:MM:SS) analysis showed that Sarvam-Translate is the most computationally efficient model (00:45:36), followed by NLLB-600M (01:01:39) and IndicTrans2 (01:08:23) and do not include model loading, embedding generation, vector database construction, or indexing overheads as shown in Figure. These results demonstrate that retrieval-augmented inference with Milvus and multilingual-e5-large can achieve strong translation performance while avoiding the computational cost and complexity associated with conventional fine-tuning approaches.

3.7.2. Analysis of Dynamic Retrieval Parameter Selection

Following the translation performance evaluation, this section analyses the retrieval and decoding decisions made by the proposed framework during inference. Table 5 reports the Average Combined Reward obtained using the proposed GRPO-based scoring function across different translation directions. The combined



450

Figure 3: Frequency of K and T values selected by (a) IndicTrans2 (b) NLLB-600M (c) Sarvam-Translate

Translation Direction	IndicTrans2	NLLB-600M	Sarvam-Translate
En → Mni	42.2349	41.9140	16.1193
Mni → En	23.2104	23.8729	24.0193
En → Brx	40.2715	-	16.6616
Brx → En	22.6228	24.2872	21.0122

Table 5: Average Combined Reward (sacreBLEU + BERT Similarity) Across Translation Models

reward is used only for ranking translation candidates during inference and is computed from the weighted combination of sentence-level BLEU and BERT score. Consequently, it reflects the effectiveness of the candidate selection strategy rather than the overall performance of the underlying translation model. For En→Mni and En→Brx, IndicTrans2 achieved the highest rewards (42.23 and 40.27, respectively), followed closely by NLLB-600M for En→Mni (41.91). For the reverse translation directions, Sarvam-Translate obtained the highest reward for Mni→En (24.02), while NLLB-600M achieved the highest reward for Brx→En (24.29). These results indicate that the proposed reward function effectively differentiates candidate quality and selects different optimal configurations depending on both the translation model and language direction.

Figure 3 show the frequency of the selected K and T during inference. Across all three models, $K = 10$ is selected most frequently, indicating that larger retrieval contexts are generally preferred for translation generation. The preferred T , however, differs across models. IndicTrans2 most frequently selects $T = 0.7$, whereas NLLB-600M and Sarvam-Translate mostly select the $T = 6$. These results demonstrate that the proposed heuristic-based framework effectively selects K and T

parameters according to the characteristics of the translation model and input test sentence.

3.7.3. Qualitative Analysis

Representative translation examples generated by the proposed AgenticRAG-GRPO framework using the best-performing model for each translation direction are presented in Table 6. The examples show that the retrieval-augmented framework produces translations that are semantically faithful to the Src sentence while maintaining contextual relevance and fluency. By grounding generation with retrieved parallel examples, the proposed framework substantially reduces hallucinations compared with conventional zero-shot and fine-tuning approaches. The qualitative observations are consistent with the quantitative results, demonstrating the effectiveness of the proposed framework across both English-to-Indic and Indic-to-English translation tasks.

3.8 Conclusion and Future Work

This study presents AgenticRAG-GRPO, an RL-inspired retrieval-augmented framework for low-resource English-Indic machine translation. The framework integrates semantic retrieval, heuristic-based dynamic parameter selection, and a GRPO-inspired reward function to identify the most suitable translation candidate. Experimental evaluation on the WMT 2026 Low-Resource IndicMT benchmark demonstrated that Sarvam-Translate achieved the best performance for En→Mni and En→Brx, NLLB-200 Distilled 600M performed best for Mni→En, and

Translation Direction (Best Model)	Original English Sentence	Reference Sentence	GRPO-based RAG Translation
En→Mni (Sarvam-Translate)	India batter Abhishek Sharma will take no part in the T20 World Cup 2026 clash against Namibia in New Delhi on Thursday.	হোস্পিতালশিং অমসুং প্রাইমারি কেয়র প্রোভাইদরশিংগী মরক্তা কোওরদিনেসন ফগৎহমবা জরমনিনা দিজিতেল্ হেল্থ রেকোরদশিং হৌদোকথি।	ভারতকী ব্যাটার অভিষেক শর্মা থার্সডেদা নিউ দিল্লীদা নামিবিয়াগা তী-২০ বিশ্বকাপ ২০২৬-গী ম্যাচ অদুদা শরুক অমত্তা লৈরৌই।
Mni→En (NLLB-600M)	মালেমগী হকসেলগী লুচিংবশিংনা ইন্তরনেম্লে সমীৎ অমদা য়ুনিবরসেল্ হেল্থ কভরেজদা অকনবা রাফম্ থমথি।	The India women's national cricket team lifting the ICC Women's World Cup as one of the year's landmark moments.	The world's health leaders made a strong statement on Universal Health Coverage at an international summit.
En→Brx (Sarvam-Translate)	Punjab Kings defeated Mumbai Indians in Jaipur to secure a top-two league finish.	মুলুগ সাবসি আফাদা গোসোনি সাবসিখৌ মোনসে মুলুগনাং থাখি হিসাবে সানদৌ।	পঞ্জাব কিংসআ জয়পুরাব মুম্বাই ইন্ডিয়ান্সখৌ হরায়নানৈ লিগনি গোজৌনি মোননৈ জায়গায়াব থানায়খৌ পক্কা খালামদৌ।
Brx→En (Indictrans2)	ইন্টরনেসনাল ফিল্ম ফেস্টিভলআ মল্টিপল কেটিগরিসিম স্ট্রং ইন্ডিয়ন প্রিজেন্সখৌ শোকেস খালামনায় জাগাসিন।	India travelled to Colombo to play Pakistan in a group-stage match.	India travelled to Colombo to play Pakistan in a group-stage match.

Table 6: Sample Translations Generated by proposed framework with Best performing models on Test Set

IndicTrans2 produced the strongest results for Brx→En. The retrieval analysis showed that was the most frequently selected retrieval depth (K), while the optimal (T) dynamically varied between 0.6 and 0.7, depending on the translation model and input sentence characteristics. Furthermore, the inference time analysis showed that Sarvam-Translate achieved the lowest average translation time across most language directions, demonstrating that high translation quality can be achieved while maintaining computational efficiency. Future work will focus on extending the proposed framework to additional low-resource Indic languages and incorporating adaptive learning-based retrieval policies instead of heuristic parameter selection. Further investigation of advanced reward functions and quality estimation metrics is expected to improve translation robustness and generalization across diverse domains.

Acknowledgments

The authors gratefully acknowledge Sri Mata Amritanandamayi Devi (Amma), Chancellor, Amrita Vishwa Vidyapeetham, for her inspiration and or providing the computational infrastructure and research environment that supported this work.

References

- Shailashree K. Sheshadri, Deepa Gupta, and Marta R. Costa-Jussa. 2023. A voyage on neural machine translation for Indic languages. *Procedia Computer Science*, 218:2694–2712.
- Partha Pakray, Santanu Pal, Advaita 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 WMT 2026 shared task on low-resource Indic languages translation. In *Proceedings of the Eleventh Workshop on Machine Translation (WMT)*, under EMNLP 2026, Budapest, Hungary.
- Aarathi Rajagopalan Nair, Deepa Gupta, Biswajit Paul, and J. Siva Bhavani. 2025. Enhancing low-resource

- Indian language machine translation using large language models with preference optimization and hypergeometric-gamma reward. *IEEE Access*, 14:1641–1665.
- Armel Randy Zebaze, Benoît Sagot, and Rachel Bawden. 2025. In-context example selection via similarity search improves low-resource machine translation. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 1222–1252.
- Haojie Lu and Zhenjie Zhao. 2025. SH-RAG: A syntax-based hierarchical retrieval-augmented generation framework for handling syntactic complexity in literary machine translation. In *Proceedings of the 2025 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.
- Partha Pakray, Reddi Mohana Krishna, Santanu Pal, Advaita Vetagiri, Sandeep Kumar Dash, Arnab Kumar Maji, Saralin A. Lyngdoh, Lenin Laitonjam, Anupam Jamatia, Koj Sambyo, Ajit Das, and Riyanka Manna. 2025. Findings of WMT 2025 shared task on low-resource Indic languages translation. In *Proceedings of the Tenth Workshop on Machine Translation (WMT)*, pages 532–553.
- Marta R. Costa-Jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.
- Jay Gala, Pranjal A. Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Nawale, et al. 2023. IndicTrans2: Towards high-quality and accessible machine translation models for all 22 scheduled Indian languages. *arXiv preprint arXiv:2305.16307*.
- Shailashree K. Sheshadri, Deepa Gupta, Biswajit Paul, and J. Siva Bhavani. 2025. A novel approach to continual knowledge transfer in multilingual neural machine translation using autoregressive and non-autoregressive models for Indic languages. *IEEE Access*, 13:89775–89810.
- Xiao Liang, Yen-Min Jasmina Khaw, Soung-Yue Liew, Tien-Ping Tan, and Donghong Qin. 2025. Toward low-resource languages machine translation: A language-specific fine-tuning with LoRA for specialized large language models. *IEEE Access*, 13:46616–46626.
- Bhagyashree Wagh, Harish Bapat, Neha Gupta, and Saurabh Salunkhe. Transformers: Leveraging OpenNMT and transfer learning for low-resource Indian language translation.
- Nabam Kakum, Sahinur Rahman Laskar, Koj Sambyo, and Partha Pakray. 2023. Neural machine translation for limited resources English-Nyishi pair. *Sādhana*.
- Vikas Kumar, Mallireddy Rupesh, and Konjengbam Anand. 2025. Advancing neural machine translation for under-resourced Indic languages: Manipuri and Sindhi. In *Proceedings of the 2025 IEEE Conference on Engineering Informatics (ICEI)*, pages 1–9. IEEE.
- Sanjib Narzary, Maharaj Brahma, Sukumar Nandi, and Bidisha Som. 2025. BPE and morphologically segmented phrase based statistical machine translation system for Indian languages to resource constrained language Bodo. *Multimedia Tools and Applications*, 84(25):29715–29732.
- Merita Kasa Halili, Festim Halili, Sarah Kadriu, and Viola Mazlami. 2025. The role of vector databases in the era of generative AI. *Journal of Applied Sciences-SUT*, 11(21–22):174–183.
- Kuruva Divya Sree, Deepa Gupta, and Aarathi Rajagopalan Nair. 2026. Domain-specific retrieval-augmented generation for English-to-Hindi machine translation using BLOOMZ-3B. *Procedia Computer Science*, 283:202–212.
- Kuruva Divya Sree, Aarathi Rajagopalan Nair, and Deepa Gupta. 2026. A GRPO-enhanced agentic RAG framework for English-to-Indic machine translation. *Procedia Computer Science*, 282:2978–2992.
- Aarathi Rajagopalan Nair and Deepa Gupta. 2026. Direct preference optimization in LLMs for Indian language translation guided by COMET quality estimation. *Procedia Computer Science*, 283:189–201.
- Cai, Yudong, Jeremy Zhu, Xuan Yang, and Bang Fu. 2026. *The architecture handbook for Milvus vector database: Design and implement high-performance vector search systems with Milvus*. Packt Publishing Ltd.
- Sarvam AI. 2025. *Sarvam-translate*. Hugging Face model repository. <https://huggingface.co/sarvamai/sarvam-translate>