CycleDistill: Bootstrapping Machine Translation using LLMs with Cyclical Distillation

Anonymous ACL submission

Abstract

Large language models (LLMs), despite their ability to perform few-shot machine translation (MT), often lag behind dedicated MT systems trained on parallel corpora, which are crucial for high quality machine translation (MT). However, parallel corpora are often scarce or non-existent for low-resource languages. In this paper, we propose CycleDistill, a bootstrapping approach leveraging LLMs and few-shot translation to obtain high-quality MT systems. CycleDistill involves iteratively generating synthetic parallel corpora from monolingual corpora via zero- or few-shot MT, which is then used to fine-tune the model that was used for generating said data for MT. CycleDistill does not need parallel corpora beyond 1 to 4 few-shot examples, and in our experiments focusing on three Indian languages, by relying solely on monolingual corpora, it can achieve high-quality machine translation, improving upon a few-shot baseline model by **20-30 chrF points** on average in the first iteration. We also study the effect of leveraging softmax activations during the distillation process and observe mild improvements in translation quality.

1 Introduction

Machine translation (MT) for low-resource languages poses persistent challenges due to the limited availability of bilingual corpora and the linguistic variation these languages exhibit. Although large language models (LLMs) can perform translation with minimal supervision, their effectiveness in low-resource settings is typically inferior to systems trained with substantial parallel data (Koehn et al., 2017; Gu et al., 2018).

This paper introduces *CycleDistill*, a resource-efficient framework for improving translation quality without requiring extensive parallel data. The approach begins with a small set of example translations and utilizes LLMs to generate synthetic
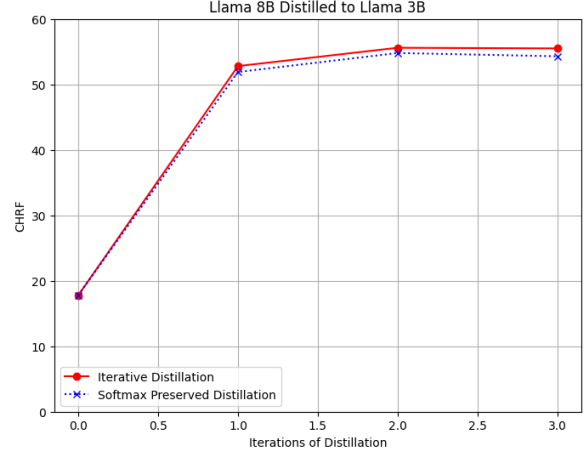

Figure 1: chrF scores over distillation cycles for LLaMA 8B → 3B using Iterative and Softmax-Preserved Distillation under a zero-shot Hindi setting. Marginal gains observed across iterations.

parallel corpora from monolingual text. These corpora are then used to iteratively fine-tune the translation model, enabling progressive performance gains with each cycle.

The framework incorporates two key techniques. First, *Iterative Synthetic Data Distillation* leverages repeated cycles of data generation and model training to enhance translation performance over time (Kim et al., 2021). Second, *Soft Distribution-Preserving Distillation* transfers detailed token-level probability distributions from teacher to student models, allowing for more comprehensive knowledge retention (Tan et al., 2019). Building on previous work in self-training (He et al., 2020), sequence-level and soft-target knowledge distillation (Kim and Rush, 2016; Hinton et al., 2015), *CycleDistill* offers a practical and scalable solution for MT in low-resource scenarios.

The main contributions of this work are:

- We present *CycleDistill*, a self-supervised MT framework that improves translation quality using only monolingual corpora and minimal

1

supervision.
- We propose a token-level soft distillation strategy to facilitate richer and more effective learning from teacher models.
- We demonstrate that our method achieves substantial improvements of 20-30 chrF points over few-shot translation baselines, with consistent chrF score gains across three Indian low-resource languages.

2 Related work

Low-resource machine translation (MT) remains a significant challenge due to the scarcity of parallel corpora and high linguistic diversity (Koehn et al., 2017; Gu et al., 2018). Knowledge distillation (KD) has become a popular approach for addressing these issues, transferring knowledge from large teacher models to smaller student models (Hinton et al., 2015). Sequence-level KD (Kim and Rush, 2016) and iterative or self-training strategies (Kim et al., 2021; Furlanello et al., 2018) have demonstrated improvements in low-resource and multilingual MT (Tan et al., 2019). Recent advances include continual KD, which sequentially distills knowledge from multiple existing models (Zhang et al., 2023), and encoder-aware KD for better transfer in compute-constrained and low-resource settings (Velayuthan et al., 2025).

Back-translation and its iterative variants are also highly effective for low-resource MT, as they leverage monolingual data to generate synthetic parallel corpora (Edunov et al., 2018; Hoang et al., 2018). These methods have shown strong gains in extremely low-resource and Indic language scenarios, especially when combined with transfer learning and data filtering (Luo et al., 2020; Tars et al., 2021; Ahmed et al., 2023; Krishnamurthy et al., 2024).

While both KD and back-translation have advanced the field, their integration and comparative effectiveness, particularly in settings with minimal parallel supervision, remain active areas of research. Our proposed **CycleDistill** framework is novel in that it bootstraps high-quality MT systems using only monolingual corpora and a handful of few-shot examples, without relying on large-scale parallel data. Unlike prior work, CycleDistill combines cyclical iterative synthetic data generation with token-level soft distribution-preserving distillation, enabling progressive model refinement and compression.
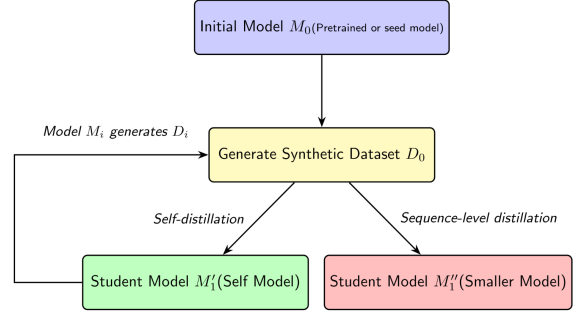

Figure 2: An overview of the CycleDistill framework, which iteratively generates synthetic parallel data from monolingual corpora and refines translation models through cyclic self distillation.

3 Methodology

This work aims to enhance low resource languages to English machine translation through the adoption of two iterative distillation strategies: cyclic synthetic data generation and an advanced distillation approach that preserves detailed token-level information, such as softmax distributions and subword structures. Our methodology is grounded in recent developments in knowledge distillation and self-training for neural machine translation (Kim and Rush, 2016; Gou et al., 2021).

3.1 Iterative Synthetic Data Distillation

Our first approach enables the base translation model to iteratively improve by generating and learning from its own synthetic data. The procedure is as follows:

- **Base Model Initialization:** The process begins with a pretrained base translation model, denoted as M_0, which is capable of translating from an Indic language to English.

- **Synthetic Data Generation:** The model M_0 is employed to generate a synthetic dataset D_0 comprising translation pairs. This step is inspired by self-training methodologies that have demonstrated efficacy in low-resource scenarios (He et al., 2020).

- **Self-Distillation:** Utilizing the generated synthetic data, knowledge distillation is performed in two ways:
  - The same model architecture is further refined, resulting in an updated model M_1.

2

- Additionally, knowledge is distilled into a smaller student model, M_1', via sequence-level knowledge distillation, whereby the student learns from the teacher's generated translations (Kim and Rush, 2016).

- **Iterative Refinement:** This procedure is repeated for three cycles. In each iteration i (where $i = 1, 2, 3$):
  - The distilled model M_i (or M_i') produces a new dataset D_i comprising additional translations.
  - Subsequently, M_i is distilled into M_{i+1} and a new student model M_{i+1}'.

The underlying principle is that, by iteratively learning from its own outputs, the model can progressively improve its performance. This iterative process benefits both the primary and the student models, enhancing their generalization capabilities and, in certain cases, enabling model size reduction with minimal compromise in performance.

3.2 Soft Distribution-Preserving Distillation

The second strategy extends the distillation process by capturing more granular information from the teacher model:

- **Enhanced Data Extraction:** During synthetic translation generation, for each token position t, we record:
  - The top-k token predictions ($\{y_1^{(t)}, \ldots, y_k^{(t)}\}$) (Fan et al., 2018)
  - The corresponding softmax probabilities ($\{p_1^{(t)}, \ldots, p_k^{(t)}\}$), where $\sum_{j=1}^{k} p_j^{(t)} \leq 1$

This comprehensive information set is motivated by the demonstrated effectiveness of soft-target distillation in capturing the teacher model's knowledge (Hinton et al., 2015).

- **Logit-Based Distillation:** The student model is trained to match not only the final output sequences but also the softmax distributions over the top-k tokens at each position. This is achieved by minimizing the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951) loss:

$$\mathcal{L}_{\text{KD}} = \sum_{t=1}^{T} \text{KL} \left(P_{\text{teacher}}^{(t)} \parallel P_{\text{student}}^{(t)} \right)$$
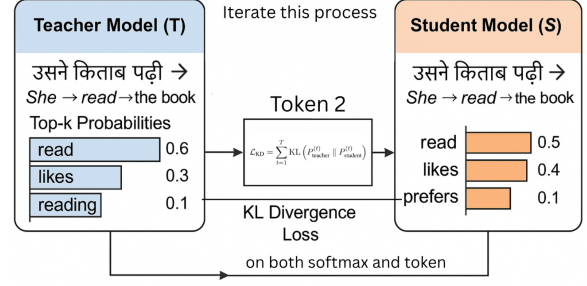

Figure 3: An Overview of the *Soft Distribution Preserving Distillation*. Unlike standard distillation, this method preserves top-k token distributions at each position. The student model learns not only from final outputs but also from the richer probability landscape, encouraging finer-grained generalization.

where T denotes the sequence length, and $P^{(t)}$ represents the softmax distributions. This approach enables the student model to more accurately approximate the teacher's behavior, as suggested in prior research (Hinton et al., 2015; Mukherjee and Khapra, 2021).

- **Iterative Distillation:** This process is also conducted over three iterations. In each cycle, the student from the previous round assumes the role of the new teacher, and a fresh synthetic dataset is generated, ensuring the transfer of rich token-level distributions.

4 Experiments

This section outlines the experimental framework designed to investigate the efficacy of iterative knowledge distillation in enhancing machine translation quality. Our approach involves distilling knowledge from larger language models into smaller counterparts, followed by comprehensive performance evaluation across multiple metrics and languages.

4.1 Models and Languages

Our study employs four language models of varying sizes from the LLaMA (Meta, 2024) and Gemma (Google, 2024) families:

- **Gemma 2 9B** (G_{9B})
- **Gemma 2 2B** (G_{2B})
- **LLaMA 3.1 8B** (L_{8B})
- **LLaMA 3.2 3B** (L_{3B})

Each larger model undergoes distillation to produce both a refined same-size model and a compressed smaller model, adhering to established

3

Sequence Distillation principles (Kim and Rush, 2016). Our evaluation encompasses three Indic languages:

- **Hindi** (HIN)
- **Bengali** (BEN)
- **Malayalam** (MAL)

4.2 Distillation Process

For a given teacher model T, distillation is performed to produce two student models:

- Same-size student ($S_{\text{same}} \leftarrow T$)
- Smaller student ($S_{\text{small}} \leftarrow T$)

The distillation relationships are formally expressed as:

$$G_{9B} \rightarrow \{G'_{9B}, G_{2B}\}, \quad L_{8B} \rightarrow \{L'_{8B}, L_{3B}\}$$

where the refined large models (G'_{9B}, L'_{8B}) are subsequently utilized for synthetic data generation. We select $k = 20$ after empirical evaluation of the teacher models' output distributions revealed that the probability mass beyond the 20 highest-scoring tokens is negligible. We perform the experiments only upto three iterations ($n = 3$). This limit was set because we observed that the performance gains stabilized after the third iteration. Further iterations yielded negligible improvements, indicating that the models were approaching a performance plateau, making additional computational cycles inefficient.

4.3 Training Data

Models are fine-tuned using the **BPCC seed corpus** (Gala et al., 2023), a parallel Indic-to-English dataset. Consistent with established practices in low-resource translation research (Kunchukuttan et al., 2023), we randomly sample 20,000 sentence pairs for training and distillation. We use a fixed prompt format for all of the language and model pair, discussed in Figure 4.

4.4 Synthetic Data Generation

Following each distillation iteration, the most recent large model generates synthetic English translations for the original 20,000 source sentences. This synthetic data generation process is repeated for three complete cycles to enable progressive model refinement.

4.5 Prompt Used

The prompt utilized for the translation task described in Section 4.3 is shown in Figure 4.

Figure 4: Example of the general prompt used for the translation task.

In 1-shot and 4-shot settings, example translation pairs are inserted into the middle section of the prompt prior to the final instruction.

4.6 Evaluation

Model performance is assessed using the **IN22 Gen corpus** (Gala et al., 2023), the standard evaluation benchmark coupled with the BPCC seed corpus. The translation quality is quantified through chrF scores (Popović, 2015). This metric provides standardized measurement of n-gram translation accuracy, aligning with current best practices in machine translation evaluation.

5 Results and Analyses

We first describe our main results on CycleDistill (iterative self distillation) and then analyze its various effects.

5.1 Main Results

Zero-Shot Setting We observe a consistent performance trend across iterations of distillation. The first iteration results in a substantial performance increase. The second and third iteration usually have similar values with the first iteration, but we notice a small increase of 1-2% of chrF with each iteration.

This pattern holds true for both *iterative distillation* and *soft distribution-preserving distillation*, with no significant differences observed between the two. However there are some notable results –

- For the Gemma 2B model with Bengali and the LLaMA 3B model with Malayalam, itera-

Model	Iter	chrF (0-shot)			chrF (1-shot)			chrF (4-shot)		
		BEN	HIN	MAL	BEN	HIN	MAL	BEN	HIN	MAL
G_{9B}	Base	41.4	47.9	39.9	42.7	49.2	38.8	24.2	44.6	14.5
	DD_1	61.1	64.4	60.2	60.8	64.2	60.0	53.1	63.8	37.0
	SD_1	60.9	64.7	60.4	60.1	64.5	57.9	49.3	63.7	18.2
	DD_2	61.4	64.5	60.7	60.5	64.6	60.2	52.4	63.7	37.2
	SD_2	60.5	64.7	60.7	64.8	64.9	59.1	49.3	64.3	32.9
	DD_3	61.0	60.4	61.1	60.6	59.0	60.4	52.8	57.7	37.8
	SD_3	61.4	64.4	61.0	60.9	63.3	58.4	45.0	64.1	48.1
L_{8B}	Base	29.2	33.6	22.8	26.6	36.0	8.5	13.5	24.1	14.0
	DD_1	44.9	29.8	42.6	39.6	26.8	17.6	16.7	18.9	17.4
	SD_1	42.1	40.3	40.6	32.0	39.6	21.2	16.7	29.3	17.4
	DD_2	48.3	50.3	46.2	42.0	55.5	26.4	16.5	51.1	17.4
	SD_2	46.2	54.1	44.5	38.3	39.4	23.5	15.1	33.4	17.4
	DD_3	38.9	37.3	17.8	30.0	27.6	15.0	18.3	21.0	17.4
	SD_3	38.9	50.8	38.0	38.7	40.7	22.3	17.0	27.3	17.4
L_{3B}	Base	24.2	14.5	2.9	18.4	17.8	5.0	13.4	14.5	14.0
	DD_1	46.0	52.7	38.9	39.3	52.8	27.4	27.0	36.3	17.4
	SD_1	49.4	53.1	33.5	37.5	51.9	18.2	17.2	34.5	17.3
	DD_2	34.3	55.0	37.5	28.0	55.6	24.5	12.8	42.7	17.3
	SD_2	52.3	54.4	29.4	39.3	54.8	17.5	16.6	44.4	17.2
	DD_3	26.1	55.1	27.1	16.4	55.5	18.7	13.4	42.6	17.4
	SD_3	45.2	53.9	25.3	37.5	54.3	17.4	13.5	42.8	17.3
G_{2B}	Base	24.6	28.8	23.8	28.7	33.4	27.8	19.0	31.2	13.4
	DD_1	50.9	58.4	48.3	50.3	58.7	46.6	27.7	54.1	25.4
	SD_1	40.1	58.3	48.2	58.3	56.9	47.1	23.8	55.5	23.0
	DD_2	50.0	58.1	48.2	50.1	58.4	47.1	29.0	53.8	25.8
	SD_2	43.0	58.4	49.0	48.8	58.1	47.4	28.6	51.2	21.4
	DD_3	49.9	57.8	47.4	49.4	57.2	46.9	34.9	54.9	25.3
	SD_3	49.1	56.8	48.5	45.4	56.8	47.0	32.8	53.3	21.0
Average		**44.4**	**51.5**	**40.9**	**39.8**	**49.6**	**31.0**	**26.8**	**42.5**	**21.6**

Table 1: chrF scores for all models and methods across three languages and shot settings, with column averages.

tive distillation outperforms soft distribution-preserving distillation.

- In contrast, for the LLaMA 8B model with Hindi and the LLaMA 3B model with Bengali, soft distribution-preserving distillation demonstrates superior performance compared to iterative distillation.

One-Shot Setting The one-shot setting yields the best overall performance, with the highest chrF scores observed exclusively in this configuration. The performance trend across iterations closely resembles that of the zero-shot setting. We observe some crossover between the two distillation methods, where one approach outperforms the other depending on the iteration count. Notable observations include:

- For the LLaMA 3B model on the Malayalam dataset, iterative distillation surpasses soft distribution-preserving distillation in performance.

- Conversely, for the LLaMA 3B model on the Bengali dataset, soft distribution-preserving distillation outperforms iterative distillation.

Four-Shot Setting Performance declines slightly in the four-shot setting compared to earlier configurations, though iteration-wise differences remain minimal. Both iterative and soft distribution-preserving distillation exhibit similar gradual improvements and overall trends. This drop is primarily attributed to reduced contextual clarity due to increased input length, the four-shot prompt is approximately 60% longer than the one-shot, placing greater demands on the model's context window. Maintaining coherence across multiple examples becomes harder as prompts grow longer. The degradation is more pronounced in linguistically complex languages, suggesting that context dilution disproportionately affects grammatically rich targets. These results highlight the need to balance shot count and context efficiency in multilingual

5

distillation, especially under limited model capacities.

5.2 Impact of Language Morphology on chrF

To further investigate the observed decline in 4-shot performance, particularly for morphologically rich languages, we visualize language-specific sensitivity to increasing shot settings. As shown in Table 1, we find a notable and steeper decline from 1-shot to 4-shot for Bengali and Malayalam, compared to Hindi, which supports the hypothesis that context dilution disproportionately impacts morphologically complex languages.

5.3 Effectiveness in Extremely Low Resource Languages

Study on Nepali To assess the robustness and generalizability of our proposed method in low-resource or moderately known language settings, we conducted experiments using Meta's LLaMA 3.1 8B and LLaMA 3.2 3B models. We selected Nepali, written in the Devanagari script, as the target language. This language has partial representation in the model's pretraining corpus, which means the models possess a basic understanding of it and are capable of generating reasonable outputs, although it is not extensively covered. Despite this limited exposure, the models were able to produce useful distillation data. When we applied our method, we observed consistent improvements over baseline methods, as shown in Table 2. These results suggest that our method remains effective even when the target language has minimal presence in the training data. This demonstrates the potential of our approach to enhance performance in low-resource and cross-lingual generalization scenarios.

Study on Manipuri The investigation included preliminary experiments on the Manipuri (Meitei script) to English translation task, utilizing several prominent large language models, specifically GPT-4, LLaMA 3.1 8B, and Gemma 2 9B. These models were evaluated for their ability to generate synthetic distillation data, which is the first step for the proposed CycleDistill framework.

Results indicated a significant limitation: none of the evaluated models were capable of producing usable distillation data for Manipuri. This suggests that the process is inherently constrained in environments where the base large language model cannot effectively perform few-shot translation for the target low-resource language. Further detailed
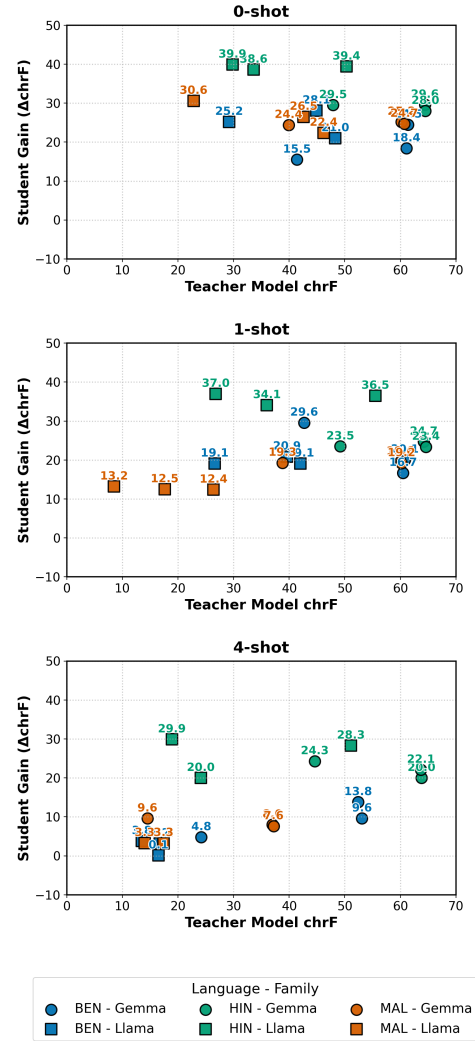

Figure 5: Scatter plot illustrating the relationship between teacher model performance and student model gain across zero-shot, one-shot, and four-shot settings in the CycleDistill framework.

experiments were conducted on Manipuri (Meitei script) using the LLaMA 3.1 8B and LLaMA 3.2 3B models within the iterative distillation framework. As presented in Table 2, these results consistently showed no improvement in chrF scores across successive iterations.

5.4 Further Analyses

Teacher Quality vs. Student Gain

To examine the correlation between teacher model performance and student gains within our CycleDistill framework, we analyzed the relevant data as depicted in Figure 5, where the x-axis indicates teacher performance (measured by the chrF score of models such as G'^*_{9B} or L'^*_{8B} when gener-

Model	Iter	Nepali (Devanagari Script)			Manipuri (Meitei Script)		
		0-shot	1-shot	4-shot	0-shot	1-shot	4-shot
L_{8B}	Base	12.47	13.95	–	16.88	17.45	17.45
	DD_1	38.59	38.08	–	18.51	17.74	17.75
	SD_1	54.44	36.19	–	16.97	17.61	17.43
	DD_2	35.23	30.45	–	18.52	17.02	17.17
	SD_2	54.31	35.19	–	18.84	17.82	18.08
	DD_3	33.24	20.38	–	17.87	15.97	15.98
	SD_3	54.74	34.35	–	18.04	16.98	16.93
L_{3B}	Base	17.16	17.15	–	17.13	17.44	17.45
	DD_1	48.55	48.75	–	18.58	16.82	17.41
	SD_1	47.31	25.51	–	18.70	16.77	16.81
	DD_2	40.48	38.23	–	17.88	14.74	14.57
	SD_2	47.31	25.67	–	17.35	15.11	14.81
	DD_3	41.15	39.34	–	17.49	15.73	15.59
	SD_3	47.08	31.11	–	17.08	13.64	13.47

Table 2: chrF scores for Nepali (Devanagari script) and Manipuri (Meitei script) over the Llama model family.

ating synthetic data), and the y-axis represents student gain (ΔchrF, denoting the improvement over the baseline, e.g., $\text{chrF}^*_{G^*_{2B}}\text{distilled} - \text{chrF}^*_{G^*_{2B}}\text{base}$).

Our analysis reveals that this relationship varies by shot setting. In zero-shot, a positive correlation holds, with higher teacher scores driving greater gains, validating distillation's reliance on data quality in example-free scenarios. In one-shot, correlation vanishes, as a single example anchors learning, making gains independent of teacher quality. In four-shot, gains are suppressed overall, due to context dilution and error propagation in longer prompts, positioning one-shot as the optimal for effective distillation.

Error Propagation and Recovery

A key limitation observed during our experiments is the susceptibility of the iterative framework to error propagation. Specifically, if an error such as the use of incorrectly generated or misaligned synthetic data is introduced at any iteration (for example, the second cycle), it can lead to a substantial degradation in performance, with declines of up to 30 to 40 chrF points observed in certain settings. These errors are compounded across subsequent iterations, as the model continues to self-distill based on flawed data, making recovery increasingly difficult. However, we also find that corrective interventions such as fine-tuning with accurately generated synthetic data can effectively mitigate such errors in subsequent iterations. This underscores the importance of early detection and correction of distillation errors, as well as the need for robust validation mechanisms during each cycle
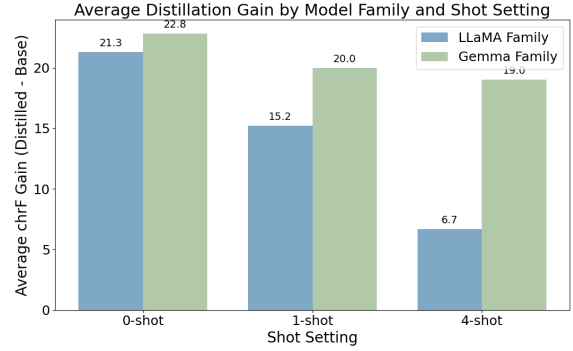

Figure 6: chrF gains for Gemma and LLaMA across shot settings.

to prevent error amplification.

Performance of CycleDistill over Model Families

A key finding is the divergence in performance between LLaMA and Gemma models under CycleDistill, as shown in Figure 6. Gemma exhibits superior, robust learning, as compared to LLaMA.

These results emphasize that the choice of base model architecture critically influences the stability and effectiveness of iterative distillation strategies.

Efficiency of Knowledge Absorption across Model Families

The analysis of knowledge absorption rates reveals that the LLaMA 3B model exhibits a significantly higher efficiency in learning from its teacher compared to the Gemma 2B model. Specifically, the average absorption rate for LLaMA 3B is 1.190, while Gemma 2B achieves 0.628. This metric is

defined as

$$\text{Absorption Rate} = \frac{\text{Student Peak Gain}}{\text{Teacher Base Score}}$$

where Student Peak Gain is the maximum chrF improvement over the student's base score across distillation iterations and Teacher Base Score is the teacher's initial chrF score, is averaged across nine evaluation conditions (three languages and three shot settings). Although the Gemma family demonstrates superior absolute chrF scores, supported by a stronger teacher (Gemma 9B), the LLaMA 3B's higher absorption rate suggests it is a more efficient learner, particularly beneficial in resource-constrained distillation scenarios.

6 Conclusion

This work presents *CycleDistill*, a structured and data-efficient framework for enhancing machine translation from low-resource languages to English. By leveraging iterative synthetic data generation and token-level soft distillation, CycleDistill improves translation performance without reliance on large-scale parallel corpora. Experimental results across multiple low-resource Indian languages confirm consistent gains in chrF scores, demonstrating the effectiveness of the approach under varying linguistic and architectural conditions.

The integration of iterative self-distillation with soft distribution-based learning reveals complementary benefits, though performance improvements taper beyond the second iteration, and translation quality remains sensitive to error accumulation, particularly in morphologically rich languages and limited supervision settings. Nevertheless, *CycleDistill* enables both model refinement and compression without relying on large-scale parallel corpora, making it an efficient and scalable solution for low-resource MT and a meaningful contribution to multilingual NLP research.

7 Limitations

Despite the effectiveness of CycleDistill in enhancing translation performance through iterative and soft distribution-preserving distillation, the approach exhibits several notable limitations. Firstly, empirical results demonstrate diminishing marginal improvements beyond the second iteration, with performance frequently plateauing or deteriorating by the third cycle. Secondly, the method relies on synthetic data generated by teacher models, which

may introduce compounding translation errors over successive iterations due to self-reinforcement effects. Thirdly, in few-shot scenarios, particularly involving morphologically rich languages such as Malayalam and Bengali, the system suffers significant performance degradation, up to 30 chrF points, largely attributable to increased prompt lengths and consequent loss of contextual coherence. Finally, the current evaluation is limited to three Indic languages and specific model families (Gemma and LLaMA), thereby restricting the generalizability of the findings to other language pairs and model architectures.

References

Mazida Akhtara Ahmed, Kishore Kashyap, Kuwali Talukdar, and Parvez Aziz Boruah. 2023. Iterative back translation revisited: An experimental investigation for low-resource english assamese neural machine translation. In *ICON*.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *EMNLP*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898.

Tommaso Furlanello, Zachary Chase Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. 2018. Born-again neural networks. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*.

Jay Gala, Pranjal A. Chitale, Raghavan AK, Varun Gumma, Sumanth Doddapaneni, Aswanth Kumar, Janki Nawale, Anupama Sujatha, Ratish Puduppully, Vivek Raghavan, Pratyush Kumar, Mitesh M. Khapra, Raj Dabre, and Anoop Kunchukuttan. 2023. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Transactions on Machine Learning Research*. Published 12/2023, reviewed on OpenReview: https://openreview.net/forum?id=vfT4YuzAYA.

Google. 2024. Gemma 2: Next-generation open models from google. https://ai.google.dev/gemma/. Accessed: 2025-05-17.

Jianping Gou, Baosheng Yu, Stephen J. Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1700–1722.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O.K. Li. 2018. Universal neural machine translation for extremely low resource languages. In *Proceedings*

8

of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT).

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Revisiting self-training for neural sequence generation. In *International Conference on Learning Representations (ICLR)*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*.

Vu Cong Duy Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. *arXiv preprint arXiv:1806.04402*.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Yunsu Kim, Jaesong Lee, Jooyoul Lee, and Hermann Ney. 2021. Improving low-resource neural machine translation with iterative back-translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*.

Parameswari Krishnamurthy, Ketaki Shetye, and Abhinav PM. 2024. MTNLP-IIITH: Machine translation for low-resource indic languages. In *WMT*.

S. Kullback and R. A. Leibler. 1951. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86.

Anoop Kunchukuttan and 1 others. 2023. The indicnlp corpus: A large-scale multilingual corpus for indic languages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Gong-Xu Luo, Ya-Ting Yang, Rui Dong, Yan-Hong Chen, and Wen-Bo Zhang. 2020. A joint back-translation and transfer learning method for low-resource neural machine translation. *Scientific Programming*, 2020.

Meta. 2024. Llama 3: Open foundation and instruction models. https://llama.meta.com/llama3. Accessed: 2025-05-17.

Subhajit Mukherjee and Mitesh M. Khapra. 2021. Distilling large-scale teacher models into compact student models for neural machine translation. *Transactions of the Association for Computational Linguistics*, 9:459–474.

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

Xinyi Tan, Longyue Wang, Tao Qin, Xiang-Yang Li, and Tie-Yan Liu. 2019. Multilingual neural machine translation with knowledge distillation. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*.

Maali Tars, Andre Tättar, and Mark Fišel. 2021. Extremely low-resource machine translation for closely related languages. In *EAMT*.

Menan Velayuthan, Nisansa De Silva, and Surangika Ranathunga. 2025. Encoder-aware sequence-level knowledge distillation for low-resource neural machine translation. In *Proceedings of the Eighth Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2025)*, pages 161–170, Albuquerque, New Mexico, U.S.A. Association for Computational Linguistics.

Yuanchi Zhang, Peng Li, Maosong Sun, and Yang Liu. 2023. Continual knowledge distillation for neural machine translation. In *ACL*.

A   Appendix A : Visualization of Effects of our Methods over Shots

This appendix provides a set of visualizations that illustrate the impact of the proposed methods under varying shot settings. Figures 7-11 demonstrate how performance characteristics evolve as the number of shots increases, thereby offering a more detailed understanding of the underlying behavior and effectiveness of our approach.
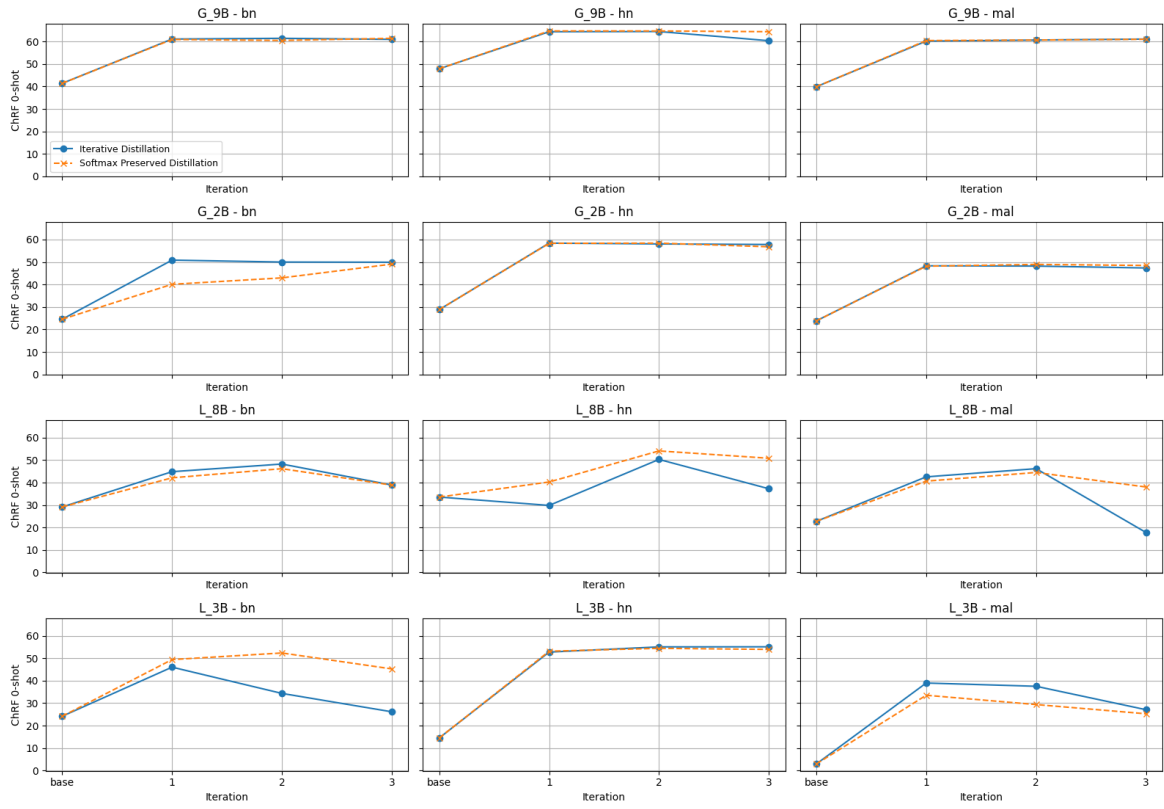
Figure 7: Comparison of the methods at 0-shot setting
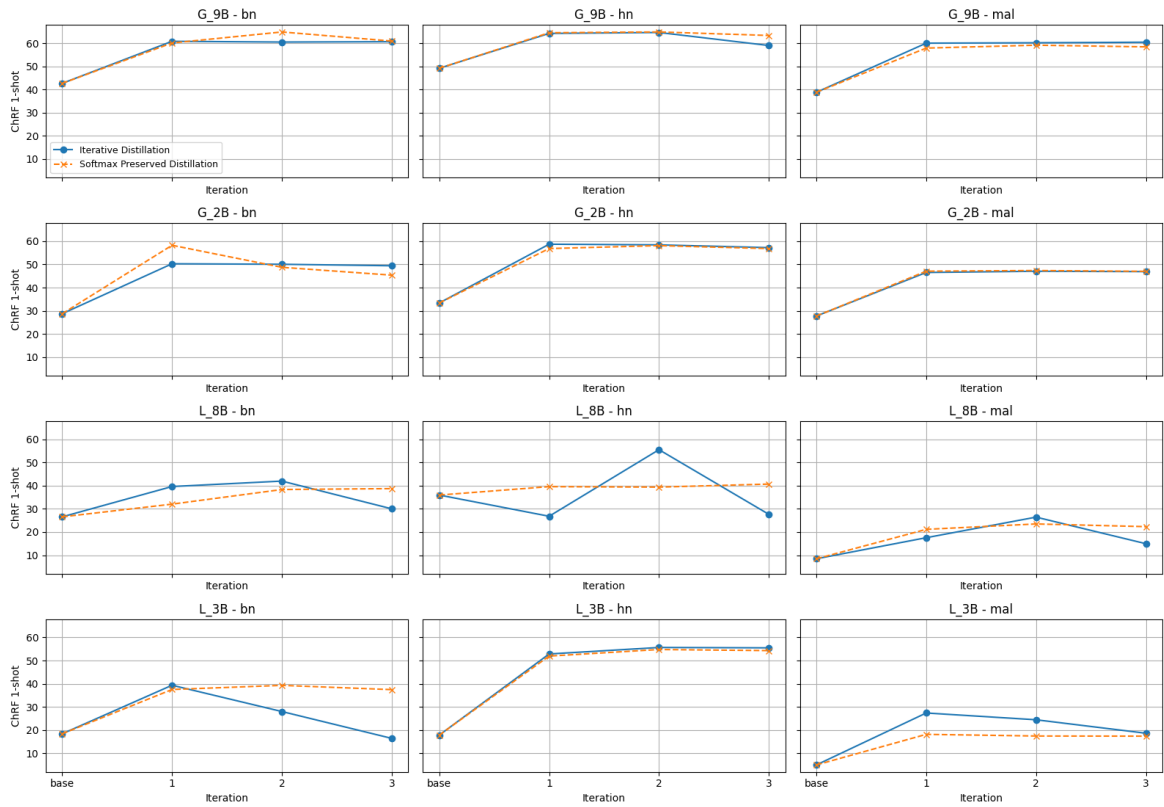
Figure 8: Comparison of the methods at 1-shot setting
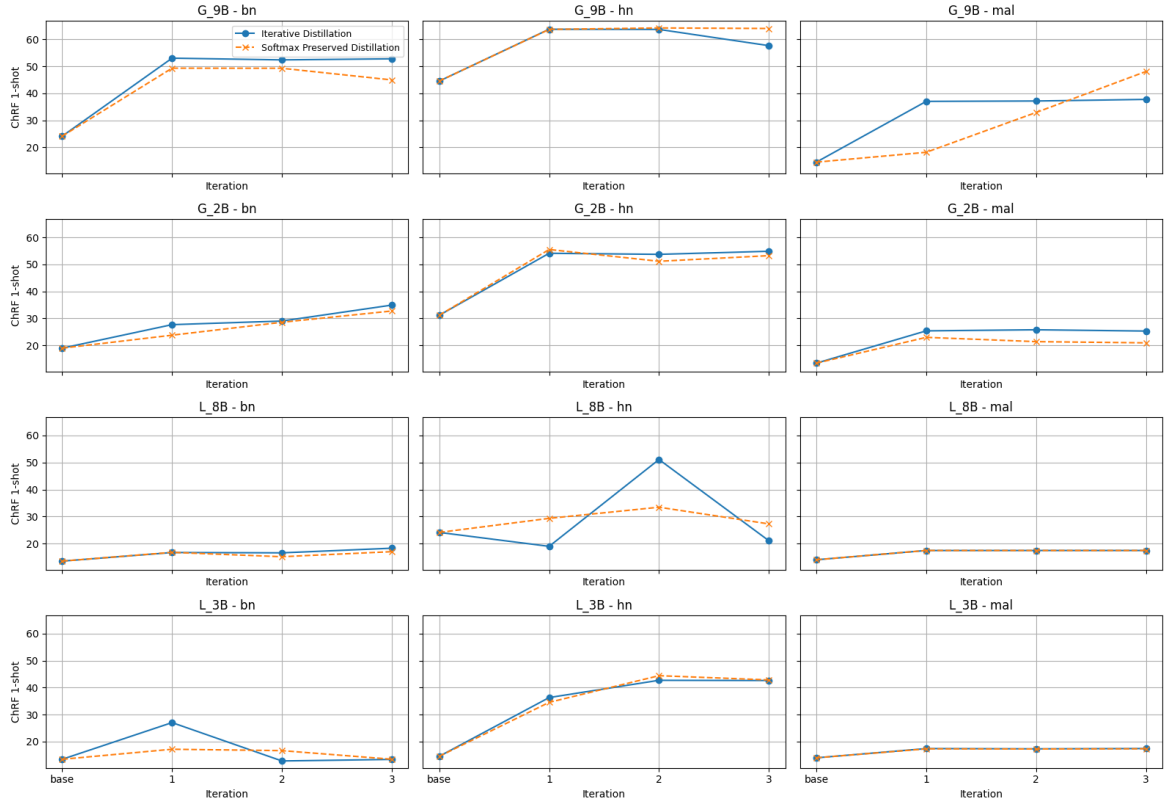
Figure 9: Comparison of the methods at 4-shot setting
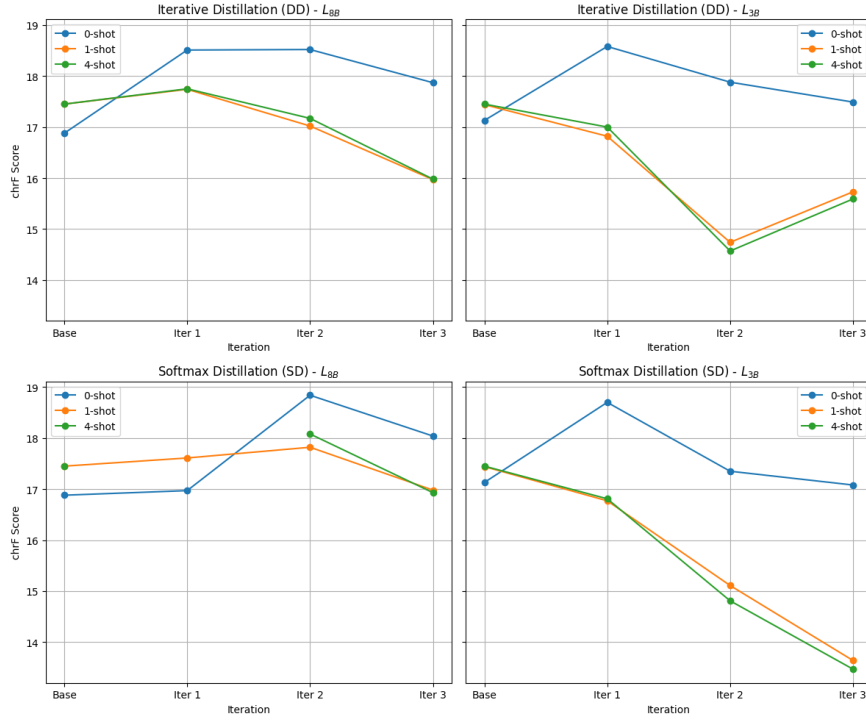
Figure 10: Comparison of the methods on the Manipuri
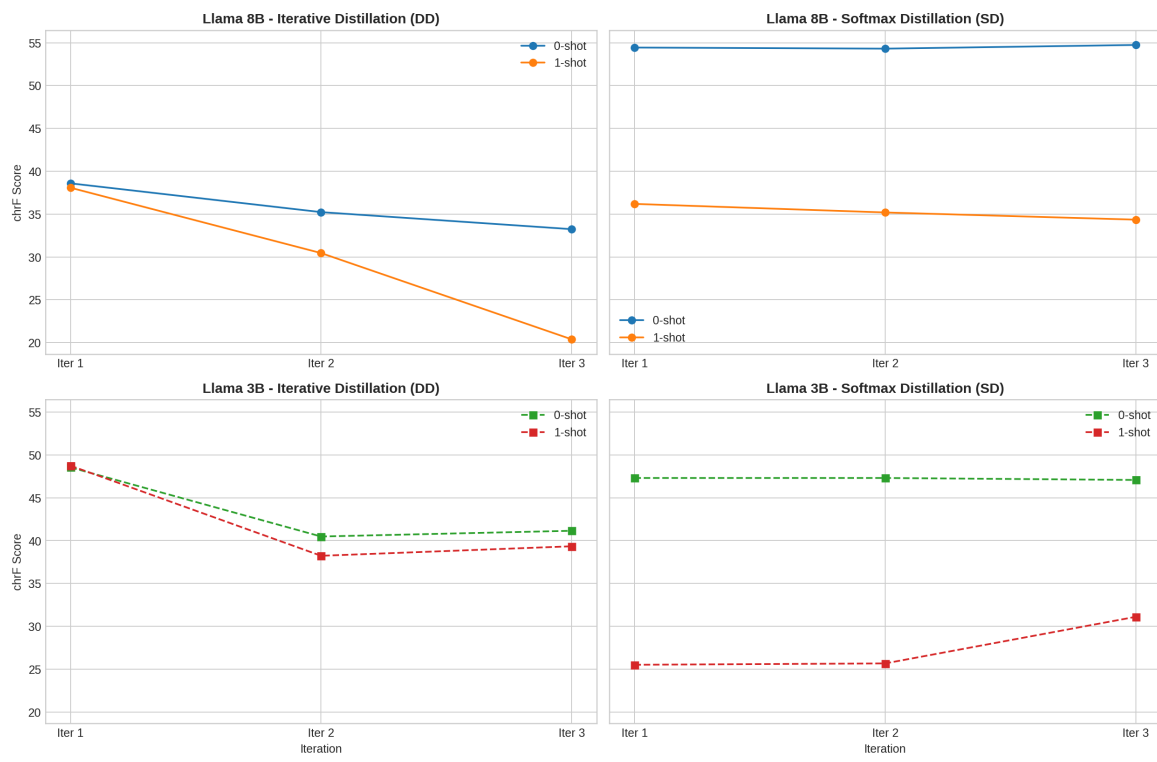
Figure 11: Comparison of the methods on the Nepali