CycleDistill: Bootstrapping Machine Translation using LLMs with Cyclical Distillation

Anonymous authors
Paper under double-blind review

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

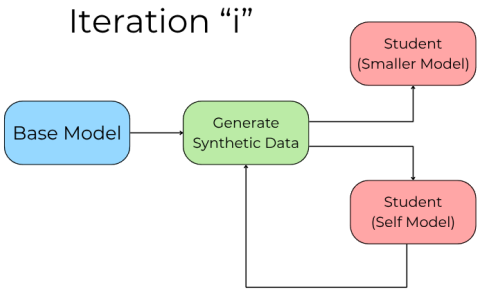

Figure 1: *Overview of CycleDistill, which iteratively builds synthetic data from monolingual corpora and refines models via cyclic distillation.*

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 & Rush, 2016; Hin-

ton et al., 2015), *CycleDistill* offers a practical and scalable solution for MT in low-resource scenarios.

The main contributions of this work are as follows:

- We present *CycleDistill*, a self-supervised MT framework that improves translation quality using only monolingual corpora and minimal 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 challenge due to limited parallel corpora and high linguistic diversity (Koehn et al., 2017; Gu et al., 2018). Knowledge distillation (KD) helps mitigate this by transferring knowledge from larger teacher models to smaller student models (Hinton et al., 2015). Approaches like sequence-level KD (Kim & Rush, 2016) and iterative self-training (Kim et al., 2021; Furlanello et al., 2018) have improved low-resource and multilingual MT (Tan et al., 2019). Advances such as continual KD (Zhang et al., 2023) and encoder-aware KD (Velayuthan et al., 2025) aim to enhance efficiency in constrained settings. Back-translation and its iterative variants are also effective in low-resource MT by generating synthetic data from monolingual corpora (**?**Edunov et al., 2018; Hoang et al., 2018). They perform well in extremely low-resource and Indic language contexts, especially with transfer learning and data filtering (Luo et al., 2020; Tars et al., 2021; Ahmed et al., 2023; Krishnamurthy et al., 2024).

Despite progress, integrating KD and back-translation and understanding their relative effectiveness in low-supervision settings remain open research questions.

Our proposed **CycleDistill** framework is novel in that it builds effective MT systems from only monolingual data and a few examples, without large-scale parallel corpora. It uniquely combines cyclical synthetic data generation with token-level soft distillation, allowing progressive model refinement and compression.

3 Methodology

This work enhances low-resource to English translation using two iterative distillation strategies: cyclic synthetic data generation and a refined distillation method that retains token-level details like softmax scores and subword patterns. Our approach builds on recent advances in knowledge distillation and self-training for neural MT (Kim & 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.
  - 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 & 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 $\{p_1^{(t)}, \ldots, p_k^{(t)}\}$, ensuring $\sum_{j=1}^{k} p_j^{(t)} \leq 1$

  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 & Leibler, 1951) loss:

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

  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 & 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}) and **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 Sequence Distillation principles (Kim & Rush, 2016). Our evaluation encompasses three Indic languages: **Hindi** (HIN), **Bengali** (BEN) and **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 ($n = 3$).

4.3 Training data

Models are fine-tuned using the **BPCC seed corpus**, a parallel Indic-to-English dataset (Gala et al., 2023). 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 Appendix A. .

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 Evaluation

Model performance is assessed using the **IN22 Gen corpus** (Gala et al., 2023), with translation quality 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

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 has 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, iterative 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.

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

Table 1: chrF scores across languages and shot settings for all models and iterations. Each cell indicates performance on a (language, shot) pair.

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 drops slightly in the four-shot setting, though iteration-wise trends remain similar. Both iterative and soft distribution-preserving distillation show gradual improvement. The decline is mainly due to reduced contextual clarity, as four-shot prompts are approximately 60% longer than one-shot, putting more strain on the model's context window. Maintaining coherence across multiple examples becomes more difficult, especially in linguistically complex languages where context dilution impacts grammatical richness. These findings emphasize the tradeoff between shot count and context efficiency in multilingual distillation with limited model capacity. We also observed notable error propagation across distillation iterations, where inaccuracies compound over time. Appendix C covers this in detail.

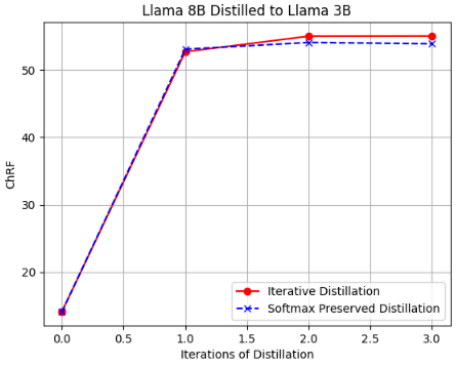

Figure 2: *chrF scores over distillation cycles for LLaMA 8B → 3B using Iterative and Softmax-Preserved Distillation under a zero-shot Hindi setting. Marginal gains observed across iterations.*

6 Conclusion

This work introduces *CycleDistill*, a structured and data-efficient framework for improving translation from low-resource languages to English. Using iterative synthetic data generation and token-level soft distillation, it enhances performance without needing large-scale parallel corpora. Experiments on several low-resource Indian languages show consistent chrF score gains, validating its effectiveness across linguistic and architectural variations.

Combining iterative self-distillation with soft distribution-based learning yields complementary benefits, though gains taper after the second iteration. Translation quality remains sensitive to error accumulation, especially in morphologically rich languages and low-supervision settings. Still, *CycleDistill* supports model refinement and compression, offering a scalable solution for low-resource MT and advancing multilingual NLP.

References

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

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

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

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

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. Indictrans2: Towards high-quality and accessible machine translation models for all 22 scheduled indian languages. *Transactions on Machine Learning Research*, 2023. URL https://arxiv.org/abs/2305.16307. Published 12/2023, reviewed on OpenReview: https://openreview.net/forum?id=vfT4YuzAYA.

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

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

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O.K. Li. Universal neural machine translation for extremely low resource languages. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2018.

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

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

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

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

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

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. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, 2017.

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

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

Anoop Kunchukuttan et al. 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)*, 2023.

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

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

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

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

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

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

Menan Velayuthan, Nisansa De Silva, and Surangika Ranathunga. 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)*, pp. 161–170, Albuquerque, New Mexico, U.S.A., 2025. Association for Computational Linguistics. ISBN 979-8-89176-230-5. doi: 10.18653/v1/2025.loresmt-1.15. URL https://aclanthology.org/2025.loresmt-1.15/.

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

A Prompt Used

The prompt utilized for the translation task described in Section 3.4 is shown in figure 3.

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

B 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 3-5 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.

C Error Propagation Analysis in the Iterative Methodology

The cumulative error at distillation iteration t, denoted as ϵ_t, can be expressed through the following recursive formulation:

$$\epsilon_t = \epsilon_{t-1} + \gamma(\delta_{\text{synth}} + \delta_{\text{KL}})$$

where:

- ϵ_t: Accumulated error at distillation iteration t

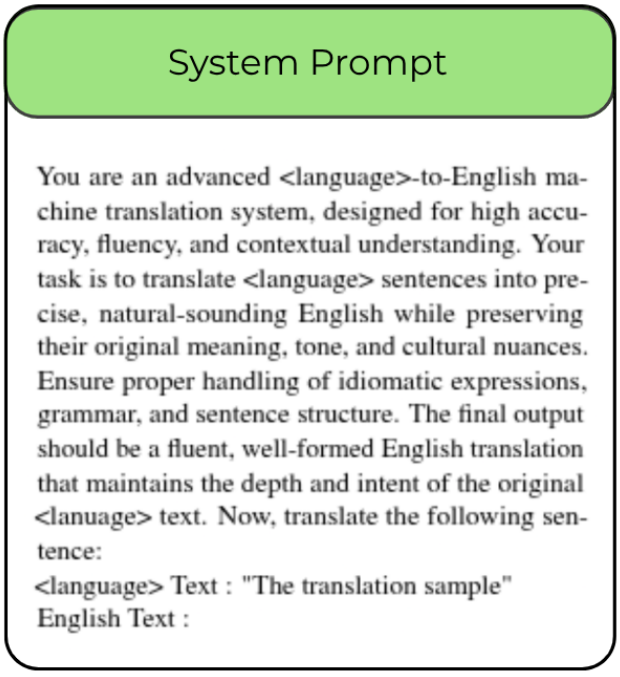

Figure 3: Example of the general prompt used for the translation task.

- γ: Learning rate coefficient that amplifies subsequent error terms
- δ_{synth}: Error component arising from synthetic data generation and imperfect distribution alignment
- δ_{KL}: Error component resulting from approximation inaccuracies in KL divergence minimization

C.1 Error source analysis

C.1.1 Synthetic data generation error

This loss function incentivizes the generator to produce exemplars where the student model's output exhibits maximal divergence from the teacher model's output, thereby concentrating the training process on challenging instances. The objective extends beyond mere generation of verisimilar samples to the production of samples that effectively enhance student model performance.

The generator objective function is defined as:

$$\mathcal{L}_G = \mathbb{E}_{z\sim p(z)}[\|T(G(z)) - S(G(z))\|^2]$$

This formulation introduces error through:

- **Architectural constraints**: The generator function $G(z)$ cannot achieve perfect replication of the true data distribution $p_{\text{real}}(x)$
- **Student capacity limitations**: The student model S exhibits insufficient representational capacity to precisely emulate the teacher model T's outputs

C.1.2 Weight initialization error

The initialization discrepancy is quantified as:

$$I_t = \|\theta_0^{\text{student}} - \theta_0^{\text{teacher}}\|$$

This discrepancy propagates through the gradient update mechanism:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta \mathcal{L}(\mathcal{S}, \theta_t)$$

where the parameter η governs the rate of error accumulation.

D Future Work

While CycleDistill demonstrates promising performance in the context of low-resource to English translation using synthetic parallel corpora and few-shot guidance from LLMs, several directions remain unexplored and merit further investigation to evaluate the robustness, generalizability, and scalability of the proposed framework.

First, we plan to extend our framework to include a highly under-documented and linguistically distant language, where even minimal digital resources are sparse or entirely unavailable. This would serve as a stress test of CycleDistill's adaptability in extreme low-resource regimes, and would provide insights into the method's resilience to noise, orthographic variability, and lack of standardized tokenization procedures.

Second, we intend to investigate the reverse translation direction (i.e., English to low-resource language) for at least one or two selected languages. This is particularly significant given the asymmetry in language modeling capabilities of LLMs, which may disproportionately favor high-resource language outputs. This line of inquiry will assess whether CycleDistill maintains its efficacy when tasked with generating fluent and culturally coherent translations into low-resource languages.

Additionally, we propose to conduct targeted ablation studies to systematically examine the stability and sensitivity of CycleDistill across different components, including the choice of synthetic data generation temperature, number of iterations, and the best iteration to stop at. These experiments will help disentangle the relative contributions of each element and offer a clearer understanding of the process dynamics.

Together, these directions aim to solidify the empirical foundations of CycleDistill and evaluate its broader applicability across varied linguistic settings.

E 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, which we have discussed in Appendix C. 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.

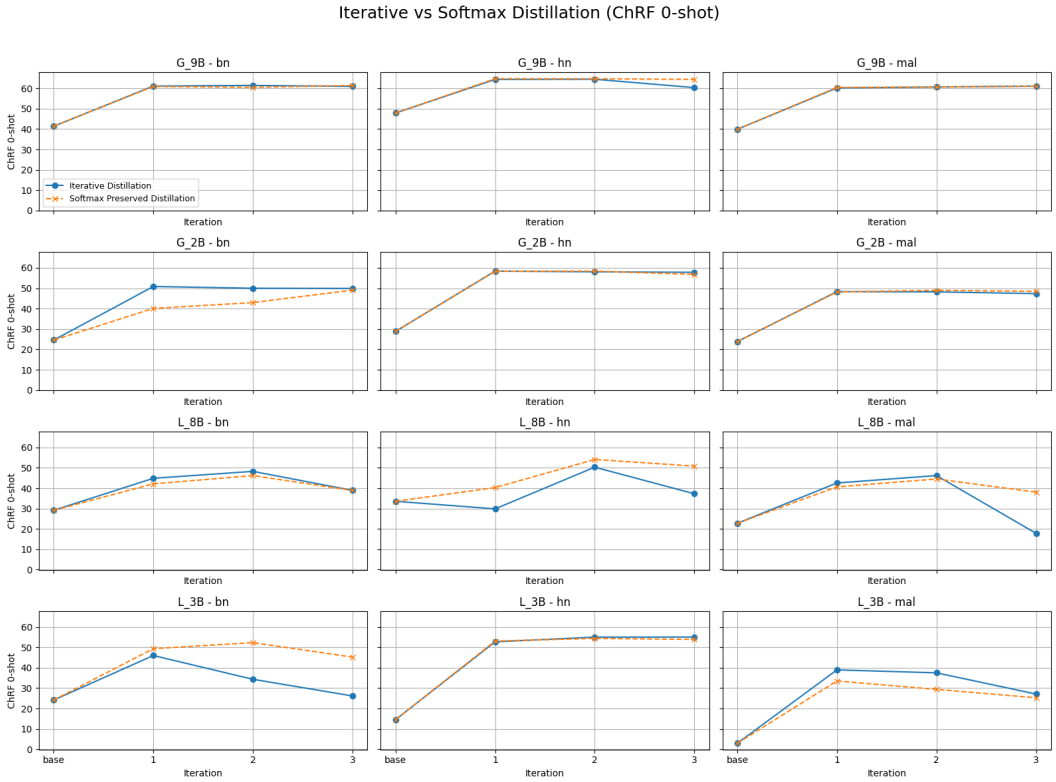

Figure 4: Comparison of the methods at 0-shot setting

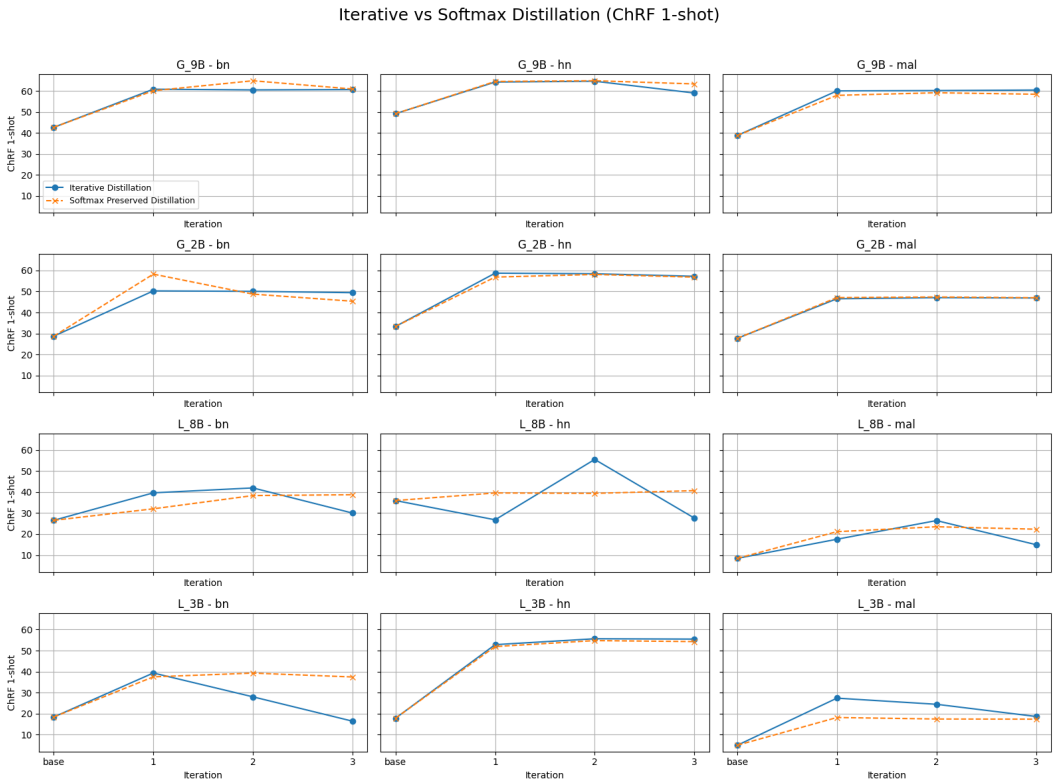

Figure 5: Comparison of the methods at 1-shot setting

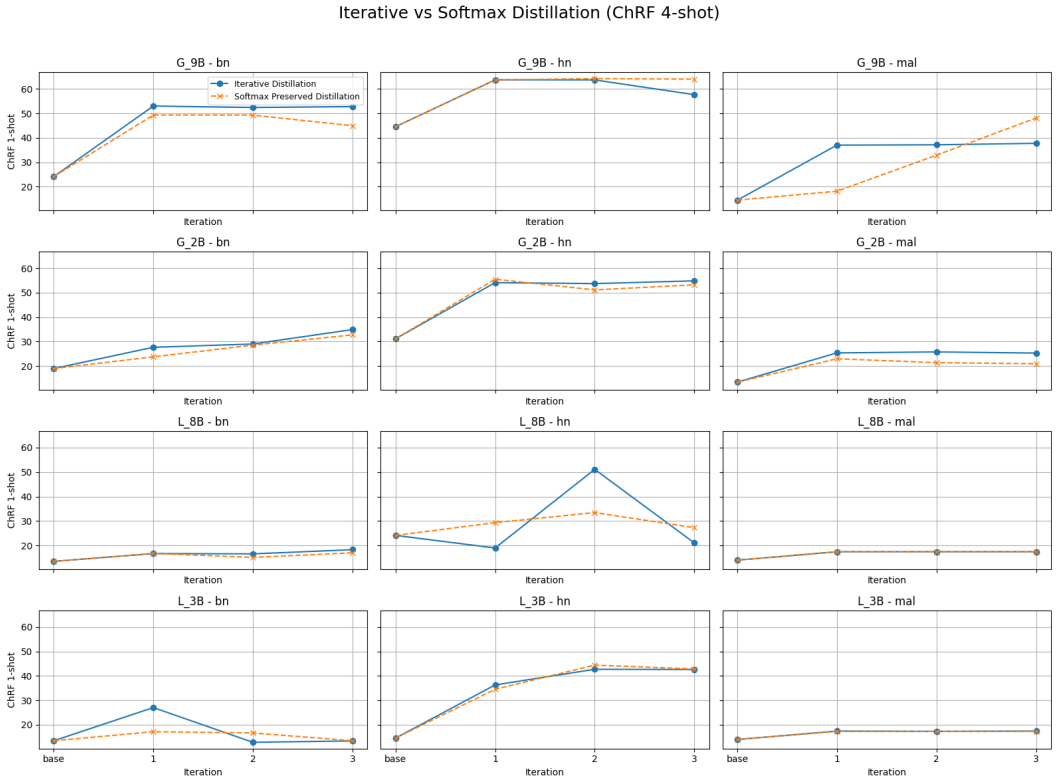

Figure 6: Comparison of the methods at 4-shot setting