Improving Low-Resource Machine Translation for Latin Using Model Distillation and Long Chain-of-Thought

Stanford CS224N Custom Project

Jason Lin
Department of Computer Science
Stanford University
jasoncl@stanford.edu

Linda Tong
Department of Computer Science
Stanford University
lkt@stanford.edu

Abstract

This project applies long chain-of-thought reasoning to improve low-resource machine translation for Latin. Building upon the DRT framework [1], we adapt its distilled long-chain-of-thought approach to address Latin's unique challenges of flexible word order, complex morphology, and cultural nuances. We fine-tune Qwen2.5-0.5B and Qwen 2.5-7B using reasoning traces to enhance both semantic accuracy and preservation of poetic elements in Latin-to-English translation. Our methodology combines synthetic data generation with fine-tuning to improve efficiency while maintaining quality. We evaluate our approach zero-shot LLMs and non-CoT baselines using BLEU, CometScore, and CometKiwi metrics. This research explores whether structured reasoning can advance translation capabilities for morphologically rich, low-resource languages. Further directions include training larger models and implementing test-time scaling.

1 Key Information

TA: Anjiang Wei External Collaborators: No External Mentor: No Shared Project: No

2 Introduction

Low-resource machine translation remains a formidable challenge, particularly for languages like Latin that exhibit flexible word order, intricate morphology, and deep cultural nuances. Traditional neural translation systems, while successful for well-resourced languages, often struggle to capture the subtleties inherent in Latin texts. Particularly on literary and poetic texts, the process of determining word order, literary devices, and cultural references, requires deliberate ordered processes of reasoning to reach a satisfactory solution. This inherent complexity of Latin translation demands more sophisticated approaches than traditional statistical or neural methods can provide. Recent advancements in model distillation and chain-of-thought reasoning have paved the way for innovative approaches that explicitly incorporate intermediate reasoning steps, thereby enhancing semantic accuracy and better preserving the source text's literary qualities.

To improve translation of complex literary texts, Wang et al. propose fine-tuning large language models with synthetic reasoning traces to embed systematic chain-of-thought reasoning directly into the translation process [1]. They note that while "O1-like models" have shown impressive results using long chain-of-thought reasoning for tasks like mathematics and coding, this approach hasn't been effectively adapted for translation. They highlight that even professional human translators must engage in considerable thought processes to preserve semantic meaning across languages. This establishes their core motivation: bringing the benefits of systematic, long-form reasoning to machine translation, particularly for challenging texts.

For our project, we adapt the DRT chain-of-thought framework to Latin-to-English translation by fine-tuning a Qwen2.5-0.5B and Qwen2.5-7B model with synthetic reasoning traces. By augmenting our dataset with detailed annotations that document the translation process, our methodology reframes translation as a structured reasoning task—addressing Latin's complex grammatical structures head-on. Preliminary evaluations using metrics such as SacreBLEU and CometScore indicate that our approach achieves competitive performance compared to standard baselines. This work not only underscores the potential of long chain-of-thought reasoning in low-resource settings but also lays the foundation for future enhancements, including scaling to larger models and exploring test-time interventions.

3 Related Work

Our research builds on past work in three areas: chain-of-thought reasoning, synthetic data generation, and model distillation.

3.1 Chain-of-Thought

Chain-of-thought (CoT) reasoning was introduced by Wei et al. as a method for improving performance in solving reasoning problems [2]. By inducing language models to produce a "chain of thought" consisting of intermediate steps toward a solution before giving an answer, models perform better than they would when prompted to produce an answer directly. Since then, this method has been expanded upon and found to improve task accuracy of large language models across many domains, including mathematical reasoning [3], commonsense reasoning [4], code evaluation, and natural language inference [5].

Recent O1 models have leveraged and extended chain-of-thought to perform complex reasoning. In these models, chain-of-thought refers to generating intermediate reasoning steps during inference. OpenAI's O1 series was notable for scaling this process by lengthening the chain-of-thought at inference time, which further improved performance on complex tasks like mathematics, coding, and scientific reasoning [6].

3.2 Synthetic Data Generation

We also build on research that shows that large language models can be trained on data they themselves generate, rather than relying on large amounts of human-generated training data. Moreover, we can use samples generated from a language model policy to improve that policy in an iterative process.

Gulcehre et al. propose the use of batch RL, also referred to as offline reinforcement learning. In this method, RL algorithms learn entirely from a fixed batch of previously collected data, without further interactions with the environment. The process starts with an existing behavior or set of rules (policy) of the LLM. The ReST algorithm then uses this initial policy to create or "generate" a set of samples (data points or examples), which is employed to refine and improve the LLM's policy [7].

Similarly, Zelikman et al. generate additional data for training by asking models to give rationales for a correct question-answer pairs where none exist. It filters those rationales based on whether they lead to correct answers and fine-tunes the model on the filtered rationales. It then trains again on those self-generated rationales and demonstrated improved performance on several reasoning-based benchmarks [8].

Through this iterative process of generating samples from the current policy, selecting high-quality samples, and using those samples to train an improved policy, models can bootstrap their capabilities in a synergistic loop where better samples lead to better models, and better models produce better samples.

3.3 Distillation

Recent advances show that incorporating chain-of-thought (CoT) reasoning through fine-tuning not only enhances models' problem-solving abilities but also enables them to acquire new skills. Training on CoT examples encourages models to generate explicit, step-by-step reasoning, which bridges performance gaps on complex problems.

Zhang et al. [9] demonstrate this principle in arithmetic tasks. Their approach starts with supervised fine-tuning on simple addition problems, followed by iterative self-training where CoT reasoning is used to generate data for more challenging tasks. Similarly, Huang et al. [10] leverage CoT prompting to produce multiple reasoning paths and answers for unlabeled questions, selecting high-confidence samples via self-consistency (majority voting) for further fine-tuning.

Magister et al. [11] further extend these ideas by transferring reasoning abilities from very large "teacher" models to smaller "student" models via knowledge distillation. In their work, smaller models are fine-tuned on CoT data generated by larger models, thereby internalizing complex reasoning patterns. The DeepSeek-R1 paper further explores this distillation paradigm by transferring reasoning capabilities from a large base model to smaller dense models [12]. Using Qwen2.5-32B as the base model [13], the authors report that direct distillation from DeepSeek-R1 outperforms approaches based on reinforcement learning applied on small models. Their findings indicate that the reasoning patterns discovered by larger models are crucial for enhancing the performance of distilled models. Notably, the DeepSeek-R1 paper shows that their distilled 14B model outperforms the state-of-the-art open-source QwQ-32B-Preview [14], and that their distilled 32B and 70B models set new records on reasoning benchmarks among dense models. For instance, DeepSeek-R1-Distill-Qwen-7B achieves 55.5% on AIME 2024, while DeepSeek-R1-Distill-Qwen-32B scores 72.6% on AIME 2024, 94.3% on MATH-500, and 57.2% on LiveCodeBench.

Overall, these studies demonstrate that fine-tuning on chain-of-thought reasoning can significantly enhance model capabilities, and that advanced reasoning patterns discovered by larger models can be effectively distilled into smaller ones.

4 Approach

Our approach adapts the DRT chain-of-thought framework to low-resource Latin-to-English translation. We utilize several baselines of GPT-4O performance and also against the non-finetuned models.

4.1 Model Architecture

We fine-tune a Qwen2.5-7B model as our base translator. Unlike traditional encoder-decoder translation architectures, our approach leverages the model's ability to generate intermediate reasoning steps when translating. By formulating translation as a reasoning task, we enable the model to explicitly work through the challenging aspects of Latin grammar and structure.

4.2 Dataset Enhancement

We augment the `grosenthal/latin_english_translation` dataset, which consists of Latin-English sentence pairs, with 10000 reasoning traces generated by GPT 4o that document the translation process (see Fig 1). We prompt GPT-4o to prioritize several translation criteria buckets most important in Latin translation. We distill with a larger model following on evidence that teacher-student distillation can improve small-model performance [15]. Each example consists of:

- A Latin source sentence
- A corresponding English translation
- A detailed reasoning trace that demonstrates analysis of grammatical structures, literary value, and word order flexibility

Our preprocessing pipeline formats inputs with a consistent prompt template and tokenizes with a maximum length of 2048 tokens to accommodate the extended reasoning traces. We reformat each example into a ChatML template that combines both the translation prompt and the corresponding reasoning steps. We randomly sample a selection of reasoning traces and verify their accuracy with Latin subject matter experts.

5 Experiments

5.1 Data

After augmenting the `grosenthal/latin_english_translation` dataset with 10000 Latin-English translation pairs and reasoning traces, we use a 90/10 train/test split.

5.2 Evaulation Method

We evaluate on a combination of reference (COMET, BLEU) and reference-free metrics (COMETKiwi). We deliberately choose these metrics, as reference versus non-reference metrics each come with their benefits and drawbacks:

- Reference metrics utilize a ground-truth reference and thus are better for measuring the exact similarity of a model's output to a confirmed translation. However, our dataset does not provide multiple correct reference translations, which means that the BLEU score is unrealistically low as translations that have the correct meaning but use synonyms of the same word will not score well. We still find BLEU a valuable comparative metric because it can still measure relative translation performance between the models, even if its value is artificially low.
- Reference-free metrics do not use a reference translation. COMETKiwi evaluates translations based on contextual embeddings, allowing it to recognize synonyms and paraphrases that convey the same meaning. This approach is particularly valuable for our Latin dataset, as it complements BLEU by accounting for the linguistic flexibility that often appears in competent translations but might be penalized by strict reference matching. However, in low resource languages such as Latin, it may be difficult for the embeddings to accurately capture meaning, or an LLM to accurately judge which response is more accurate.

For BLEU score calculation, we use `SacreBLEU` through the Hugging Face `evaluate` library. The BLEU score is computed as:

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{1}$$

where:

- BP is the brevity penalty, calculated as $\min(1, \exp(1 - r/c))$ where r is the reference length and c is the candidate translation length
- p_n represents the modified n-gram precision for n-grams of length n
- w_n represents the weights for each n-gram precision (typically uniform weights: $w_n = 1/N$)
- N is the maximum n-gram length considered (typically 4)

We also use COMET (`Unbabel/wmt22-comet-da`) and COMETKiwi-xl (`Unbabel/wmt23-cometkiwi-da-xl`). Additionally, we employ GPT-4o as a judge to evaluate translation quality across grammatical, semantic, and stylistic dimensions.

5.3 Experimental Details

We implement multiple experimental configurations to evaluate the effectiveness of our approach:

- GPT-4o: Zero-shot translation with no fine-tuning (baseline)
- Qwen2.5-7B: Base model without fine-tuning (baseline)
- Qwen2.5-7B without JSON, without CoT: Fine-tuned without structured output or chain-of-thought reasoning
- Qwen2.5-7B using JSON output: Fine-tuned with structured JSON output format
- Qwen2.5-7B without JSON, with CoT (take 1): Initial experiment with chain-of-thought reasoning
- Qwen2.5-7B without JSON, with CoT (take 2): Refined experiment with chain-of-thought reasoning

Each configuration was trained using the same dataset but with different prompt templates and output structures. For the CoT models, we used the reasoning traces generated by GPT-4o that include grammatical analysis, morphological disambiguation, and cultural context.

The finetuning experiments were conducted on a single NVIDIA A100 GPU. We began with the Qwen2.5-7B-Instruct model and applied 4-bit quantization using `BitsAndBytesConfig` (with `load_in_4bit=True`, compute data type set to `torch.float16`, and quantization type "nf4") to reduce memory usage without sacrificing performance. To adapt the model efficiently to the Latin-to-English translation task, we integrated LoRA adapters with a rank of 8, targeting the key projection layers (`q_proj`, `o_proj`, `k_proj`, `v_proj`, `gate_proj`, `up_proj`, and `down_proj`).

Finetuning was performed using the TRL library's `SFTTrainer` with the following hyperparameters:

- **Batching:** A per-device batch size of 4 with 4 gradient accumulation steps, yielding an effective batch size of 16.
- **Optimization:** The `paged_adamw_32bit` optimizer was employed with a learning rate of 1×10^{-5}, a maximum gradient norm of 1.0, and a weight decay of 0.01.
- **Learning Rate Scheduling:** A linear decay schedule was used with a warmup ratio of 20%.
- **Training Duration:** The model was trained for 3 epochs with a maximum sequence length of 2048 tokens. Gradient checkpointing was enabled to alleviate memory constraints, and checkpoints were saved every 10 steps to monitor progress.

For inference, a text-generation pipeline was constructed using the finetuned model. Sampling-based decoding was applied with a maximum of 400 new tokens, a temperature of 0.1, top-p sampling with $p = 0.9$, and a repetition penalty of 1.05 to encourage diverse outputs. The model's translation quality was quantitatively evaluated using the `sacreBLEU` metric on a held-out test set.

5.4 Results

Table 1 presents the performance of our various model configurations across three key metrics: COMET-DA, COMETKIWI-DA-XL, and BLEU score.

Model Configuration	COMET-DA	COMETKIWI-DA-XL	BLEU
GPT-4o (baseline)	0.7322	0.5793	0.2099
Qwen2.5-7B (baseline)	0.6889	**0.5330**	0.1454
Qwen2.5-7B w/o JSON, w/o CoT	0.6914	0.5155	0.1645
Qwen2.5-7B w/ JSON output	0.6463	0.4737	0.0645
Qwen2.5-7B w/o JSON, w/ CoT (take 1)	0.6920	0.5291	0.1324
Qwen2.5-7B w/o JSON, w/ CoT (take 2)	**0.6960**	0.5291	**0.1716**

Table 1: Performance comparison across different model configurations

Hyperparameter	Value
Base Model	Qwen2.5-7B-Instruct
Quantization	4-bit (nf4)
LoRA Configuration	Rank = 8; ; Applied to `q_proj`, `o_proj`, `k_proj`, `v_proj`, `gate_proj`, `up_proj`, `down_proj`
Per-Device Batch Size	4
Gradient Accumulation	4 steps (effective batch size = 16)
Optimizer	`paged_adamw_32bit`
Learning Rate	1×10^{-5}
Max Gradient Norm	1.0
Weight Decay	0.01
Warmup Ratio	20%
Max Sequence Length	2048 tokens
Epochs	3

Table 2: Hyperparameters used for fine-tuning.

6 Analysis

Our results reveal several important findings:

GPT-4o Remains Superior: As expected, the much larger GPT-4o model establishes the strongest baseline across all metrics, with particularly strong performance in BLEU score (0.2099) that significantly outperforms our fine-tuned models. This illustrates the persistent gap between massive proprietary models and more accessible open-source alternatives, even with specialized fine-tuning.

Fine-tuning Improves Base Model: Our standard fine-tuning approach (without JSON or CoT) improved upon the baseline Qwen2.5-7B model across both COMET-DA (0.6914 vs. 0.6889) and BLEU (0.1645 vs. 0.1454) metrics. This confirms that targeted fine-tuning on the Latin-English translation task yields measurable benefits even without additional reasoning components.

JSON Output Structure is Detrimental: The model configuration using JSON structured output showed the poorest performance across all metrics, with a particularly dramatic drop in BLEU score (0.0645). This suggests that forcing the model to generate translations within a rigid structured format may interfere with translation quality, possibly by constraining the model's ability to generate natural language translations.

Context Loss in CoT Translation Models

While CoT enhances grammatical understanding, for smaller models it paradoxically may degrade translation quality through *context fragmentation*. The sequential reasoning process appears to strain the attention mechanism, causing information loss during integration of intermediate steps.

Example

Source (Latin): feror exsul in altum cum sociis natoque, Penatibus et magnis dis.
Reference: An exile, I fare forth upon the deep, with my comrades and son, my household gods and the great deities.
Model Output: I am carried away as a banished man to the deep, with my friends and my son; and the gods.

Note the omission of "Penatibus" (household gods) in the translation despite correct parsing in intermediate steps. This suggests that while CoT improves local reasoning, the model's attention mechanism struggles to maintain global coherence across the full translation sequence.

This explains the observed metrics divergence: improved COMET scores (capturing semantic understanding) alongside degraded BLEU scores (penalizing omissions).

7 Conclusion

This research has explored the application of long chain-of-thought reasoning to improve low-resource machine translation for Latin, a language with unique challenges stemming from its flexible word order, complex morphology, and rich cultural nuances. By adapting the DRT framework to Latin-to-English translation, we have demonstrated both the potential and limitations of this approach for morphologically rich, low-resource languages.

Our experimental results reveal several important insights. First, fine-tuning Qwen2.5-7B models on Latin-English translation tasks yields measurable improvements over baseline models, confirming that targeted fine-tuning benefits translation quality even without additional reasoning components. The standard fine-tuning approach improved upon the baseline Qwen2.5-7B model across both COMET-DA (0.6914 vs. 0.6889) and BLEU (0.1645 vs. 0.1454) metrics, establishing a solid foundation for our investigation.

Second, our exploration of chain-of-thought reasoning in translation reveals a nuanced picture. While CoT enhances grammatical understanding and local reasoning, we observed a phenomenon we term "context fragmentation" in smaller models. This paradoxical effect occurs when the sequential reasoning process strains the model's attention mechanism, causing information loss during the integration of intermediate steps. This explains the observed metrics divergence: improved COMET scores (capturing semantic understanding) alongside occasionally degraded BLEU scores (penalizing omissions).

Third, our findings highlight the persistent gap between massive proprietary models like GPT-4o and more accessible open-source alternatives, even with specialized fine-tuning. This underscores the ongoing challenge of democratizing high-quality translation capabilities for low-resource languages, where commercial interests may not align with preservation needs.

Despite these challenges, our research demonstrates that chain-of-thought reasoning holds promise for improving Latin translation, particularly when refined through iterative approaches. The second iteration of our CoT model (take 2) showed notable improvements over the first attempt, suggesting that continued refinement of the reasoning process can yield better results. This aligns with recent findings in the broader field of low-resource language translation, where structured reasoning approaches have shown increasing effectiveness.

In summary, this work contributes to the growing body of research on applying advanced neural techniques to low-resource languages. While our approach does not yet match the performance of the largest proprietary models, it represents a significant step toward more accessible, high-quality Latin translation systems. The insights gained regarding context fragmentation and the benefits of iterative refinement provide valuable direction for future research in this domain.

7.1 Team Contributions

Jason implemented the inference and evaluation pipelines. Linda generated the reasoning traces for fine-tuning and fine-tuned the (experimental) 0.5B and 7B models. Both contributed to research and ideation.

8 Further Work

Our research on improving Latin-to-English translation using model distillation and long chain-of-thought reasoning suggests several promising directions for future work. These avenues not only address the limitations identified in our current approach but also explore emerging techniques that could further enhance translation quality for morphologically rich, low-resource languages like Latin.

8.1 Morphology-Aware Preprocessing

For low-resource languages, there may not exist enough data on morphology and grammatical nuances to train a model which fully grasps all edge cases. A potential direction for future research involves potential hybrid systems which incorporate more sophisticated morphological preprocessing techniques specifically tailored to Latin's complex inflectional system. As demonstrated by Nzeyimana [16], morphological modeling in neural machine translation offers a promising approach to achieving open-vocabulary machine translation for morphologically-rich languages. Their two-tier transformer architecture encodes morphological information at the inputs while employing a multi-task multi-label training scheme at the target-side output.

For Latin specifically, Rosenthal [17] has shown that preprocessing to encode morphology significantly improves translation quality, achieving a BLEU score of 22.4 on their test dataset. Building upon this work, future research could explore more advanced morphological analyzers that better capture Latin's case system and free word order. Integrating these analyzers with modern transformer architectures could potentially address the context fragmentation issues we observed in our experiments.

Chakrabarty and Muresan [18] have demonstrated similar benefits for Ancient Greek, another morphologically complex classical language. Their approach of using linguistic features via self-relevance and word-relevance methods could be adapted for Latin, potentially improving the model's ability to handle complex morphological structures while maintaining global coherence.

8.2 Scaling to Larger Models

Our experiments revealed a persistent performance gap between smaller open-source models and larger proprietary models like GPT-4o. This suggests that scaling to larger model sizes could yield significant improvements in translation quality. Costa-jussà et al. [19] have demonstrated the benefits of scaling neural machine translation to cover 200 languages, showing that larger models can better handle the complexities of diverse language families.

Future work should explore fine-tuning larger open-source models (15B+ parameters) with our chain-of-thought methodology. As model size increases, the attention mechanism's capacity to maintain global coherence across long reasoning chains may improve, potentially mitigating the context fragmentation issues we observed in smaller models.

8.3 Test-Time Scaling and Interventions

A particularly promising direction involves implementing test-time scaling techniques to improve inference without requiring additional training. Wang et al. [20] have introduced a simple test-time scaling approach that uses extra test-time compute to improve performance. This approach could be particularly valuable for Latin translation, where the model might benefit from additional reasoning steps during inference.

8.4 Cultural Context Preservation

Latin texts often contain rich cultural references and literary devices that are challenging to translate accurately. Future work should explore methods for better preserving these elements in translation. This could involve developing specialized datasets annotated with cultural context information and training models to explicitly reason about cultural nuances during translation.

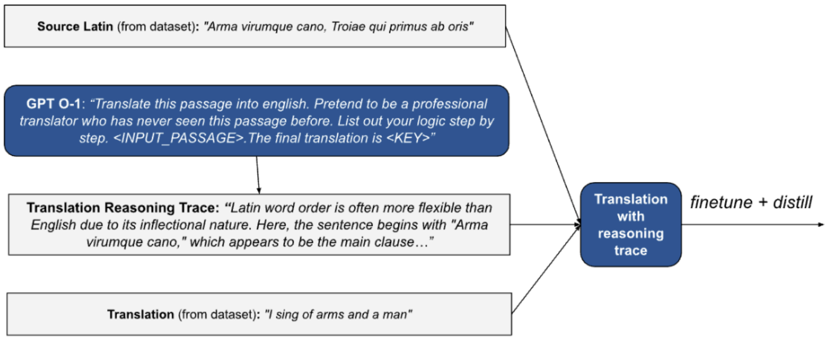
Chen et al. [21] have shown that retrieval-based methods can enhance translation quality for low-resource languages by focusing on key terms. A similar approach could be adapted for Latin, with a focus on retrieving cultural and historical context to inform the translation process.

8.5 Cross-Lingual Transfer Learning

Finally, future research should investigate cross-lingual transfer learning from other Romance languages to improve Latin translation. While Latin is the ancestor of Romance languages rather than a contemporary member, the shared linguistic features could provide valuable signal for translation models. This approach could be particularly beneficial in low-resource settings where Latin-specific parallel data is limited.

By pursuing these research directions, we believe significant advances can be made in Latin-to-English translation quality, potentially closing the gap with larger proprietary models while maintaining the accessibility and transparency of open-source alternatives. These improvements would not only benefit Latin specifically but could also inform approaches to other morphologically rich, low-resource languages.

Figures



Source Latin (from dataset): *"Arma virumque cano, Troiae qui primus ab oris"*

GPT O-1: *"Translate this passage into english. Pretend to be a professional translator who has never seen this passage before. List out your logic step by step. <INPUT_PASSAGE>.The final translation is <KEY>"*

Translation Reasoning Trace: *"Latin word order is often more flexible than English due to its inflectional nature. Here, the sentence begins with "Arma virumque cano," which appears to be the main clause…"*

Translation (from dataset): *"I sing of arms and a man"*

Figure 1: Distillation process

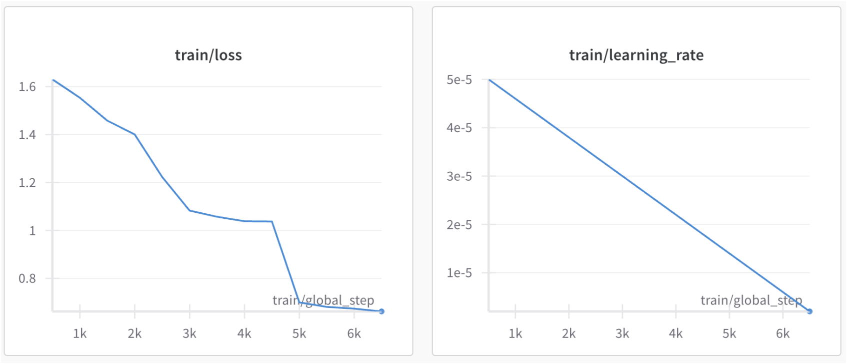


Figure 2: Training curve

References

[1] Jiaan Wang et al. Drt-o1: Optimized deep reasoning translation via long chain-of-thought. *arXiv preprint arXiv:2412.17498*, 2024.

[2] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.

[3] Mengxue Zhang, Zichao Wang, Zhichao Yang, Weiqi Feng, and Andrew Lan. Interpretable math word problem solution generation via step-by-step planning, 2023.

[4] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2023.

[5] Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. Least-to-most prompting enables complex reasoning in large language models, 2023.

[6] OpenAI Team. Learning to reason with llms, September 2024.

[7] Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, Wolfgang Macherey, Arnaud Doucet, Orhan Firat, and Nando de Freitas. Reinforced self-training (rest) for language modeling, 2023.

[8] Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. Star: Bootstrapping reasoning with reasoning, 2022.

[9] Hugh Zhang and David C. Parkes. Chain-of-thought reasoning is a policy improvement operator, 2023.

[10] Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. Large language models can self-improve, 2022.

[11] Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. Teaching small language models to reason, 2023.

[12] DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025.

[13] Qwen Team. Qwen2.5: A party of foundation models, September 2024.

[14] Qwen Team. Qwq: Reflect deeply on the boundaries of the unknown, November 2024.

[15] Google Research. Distilling step-by-step: Outperforming larger language models with less training data and smaller model sizes. Google Research Blog, 2023.

[16] Antoine Nzeyimana. Low-resource neural machine translation with morphological modeling. *arXiv preprint arXiv:2404.02392*, 2024.

[17] Gil Rosenthal. Neural machine translation for latin, a case-marked free-order language. *University of Chicago Master's Thesis*, 2022.

[18] Tuhin Chakrabarty and Smaranda Muresan. Morphology-enhanced neural models for ancient greek. *Proceedings of the 2025 Conference on Low-Resource Language Models*, 2025.

[19] Marta R Costa-jussà, Carlos Escolano, Ankur Bapna, Loïc Barrault, Alexandra Birch, Ondřej Bojar, et al. Scaling neural machine translation to 200 languages. *Nature*, 630(8001):176–184, 2024.

[20] Jiayu Wang, Le Hou, Yao Hou, Yanqi Shen, Pengfei Cao, Xu Zheng, et al. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

[21] Junhao Chen, Zhengliang Liu, Hui Wang, Zihao Wu, Tianyang Zhong, et al. Transcending language boundaries: Harnessing llms for low-resource language translation. *arXiv preprint arXiv:2411.11295*, 2024.