Reinforcement Learning for Out-of-Distribution Reasoning in LLMs: An Empirical Study on Diagnosis-Related Group Coding

Hanyin Wang[1,2], Zhenbang Wu[2], Gururaj Kolar[3], Hariprasad Korsapati[1], Brian Bartlett[1], Bryan Hull[4], Jimeng Sun[2.5]

[1]Mayo Clinic Health System, [2]School of Computing and Data Science, UIUC,

[3]Mayo Clinic Rochester, [4]Mayo Clinic Phoenix,

[5]Carle Illinois College of Medicine, UIUC

The Challenge & Core Question

- **The Problem:** Automating Diagnosis-Related Group (DRG) coding is a high-value goal for hospitals, but it's a complex, manual task requiring expert clinical reasoning.

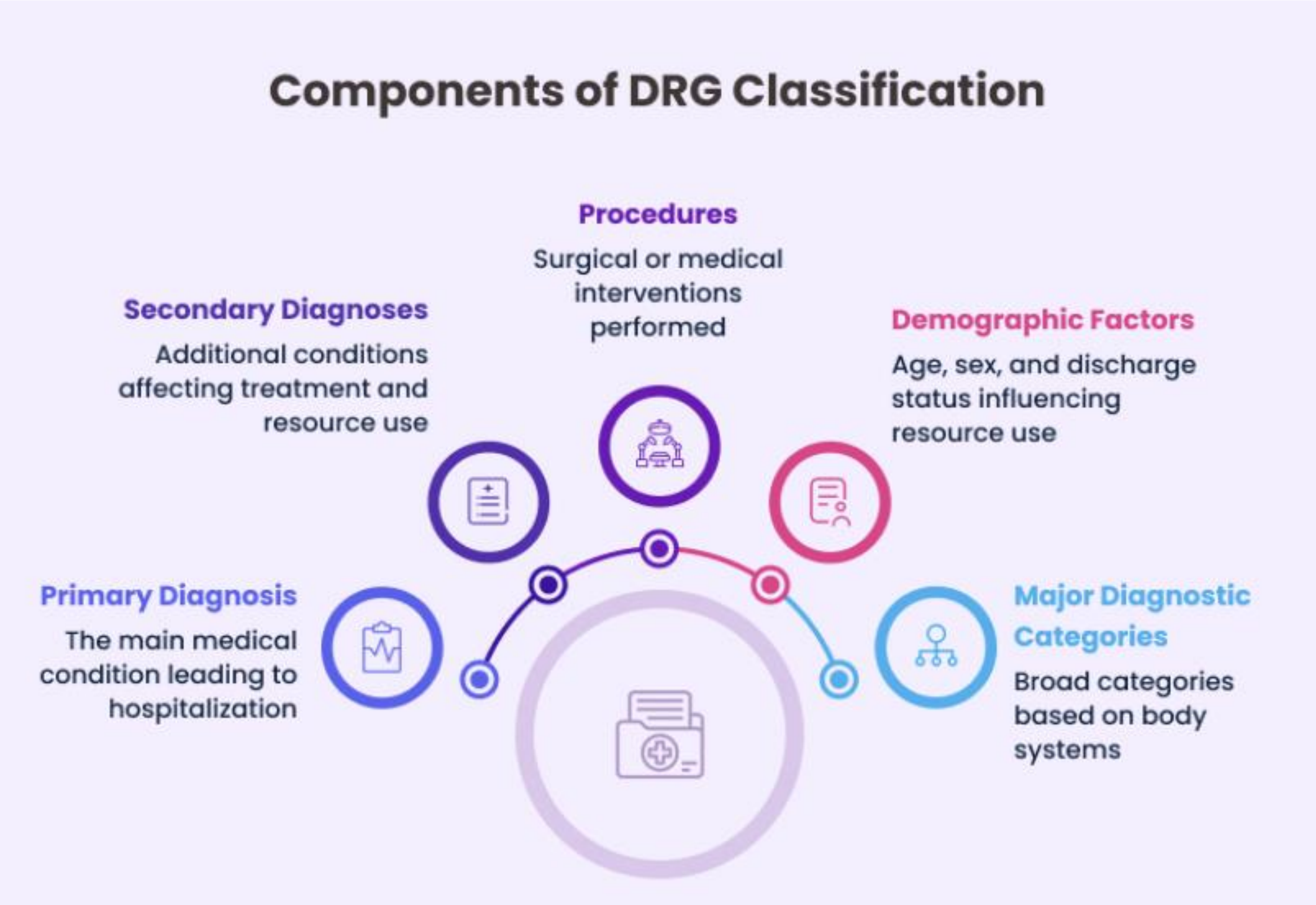
- **The LLM Hurdle:** This is a classic **Out-of-Distribution (OOD)** challenge for LLMs, which are not pretrained on the necessary private clinical or billing data.

- **Our Goal:** To use this task as a case study to understand how to apply Reinforcement Learning (RL) effectively to a novel OOD domain. We aim to answer:

1. What are the **prerequisites** for RL to succeed?

2. What is the crucial relationship between **Supervised Fine-Tuning (SFT)** and RL?

3. Can this approach yield a state-of-the-art model that is also clinically **explainable**?



Components of DRG Classification

Primary Diagnosis

The main medical condition leading to hospitalization

Secondary Diagnoses

Additional conditions affecting treatment and resource use

Procedures

Surgical or medical interventions performed

Demographic Factors

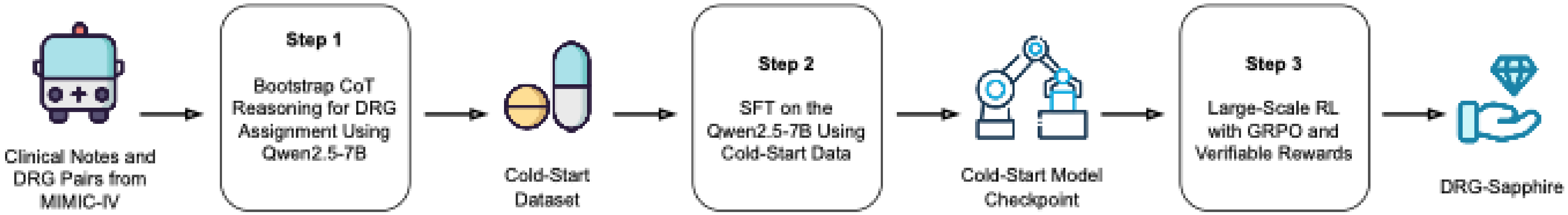
Age, sex, and discharge status influencing resource use

Major Diagnostic Categories

Broad categories based on body systems

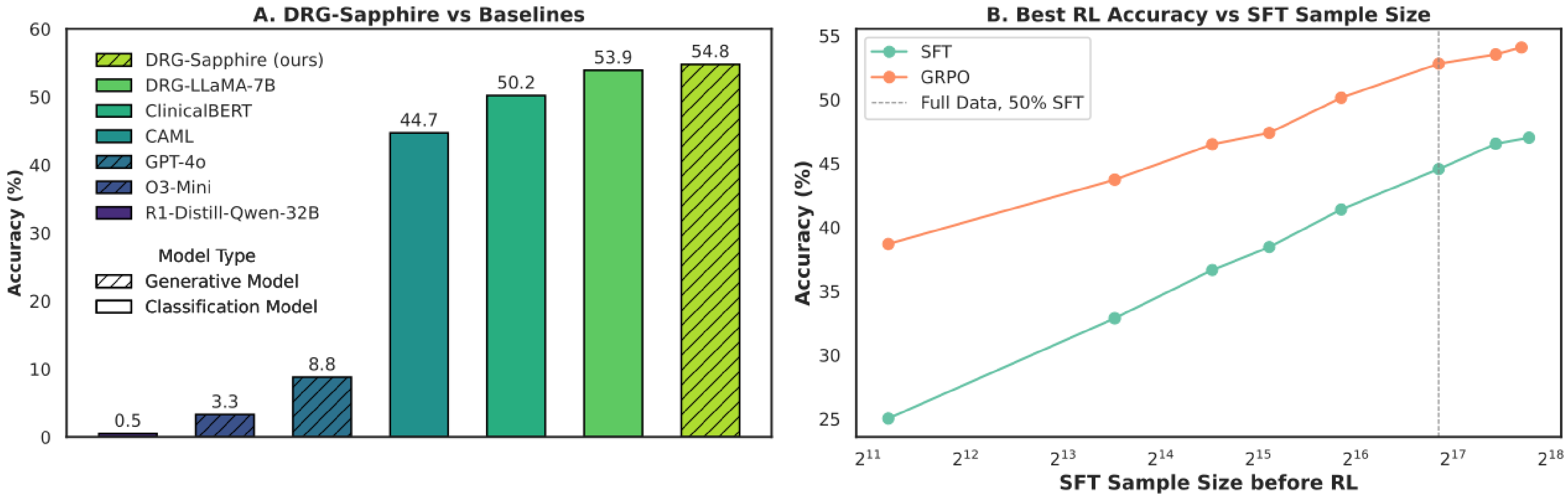
Source: https://bluebrix.health/articles/mastering-drg-coding-your-key-to-revenue-integrity-and-clinical-excellence-in-us-healthcare

Our Approach: DRG-SAPPHIRE


Clinical Notes and DRG Pairs from MIMIC-IV
Step 1
Bootstrap CoT Reasoning for DRG Assignment Using Qwen2.5-7B
Cold-Start Dataset
Step 2
SFT on the Qwen2.5-7B Using Cold-Start Data
Cold-Start Model Checkpoint
Step 3
Large-Scale RL with GRPO and Verifiable Rewards
DRG-Sapphire

- **Model:** DRG-SAPPHIRE, built on the Qwen2.5-7B model. Training and test data are from MIMIC-IV.
- **Core Technique:** Large-scale Reinforcement Learning (RL) using Group Relative Policy Optimization (GRPO).
- **How it Works:**
  1. **SFT:** First, teach the model the basics of the task and reasoning format using a specially prepared dataset.
  2. **RL:** Then, use GRPO with rule-based rewards to optimize for accuracy. The model generates multiple potential answers, and the policy is updated to favor the correct ones.
- The model generates Chain-of-Thought (CoT) reasoning, making its assignments explainable.

DRG-SAPPHIRE Performance



- **State-of-the-Art Accuracy:** DRG-SAPPHIRE achieved a **Pass@1 accuracy of 54.8%** on the MIMIC-IV test set, surpassing the previous best model, DRG-LLaMA (53.9%).

- **Outperforms Baselines:** The model significantly outperforms proprietary reasoning models like GPT-4o, as well as other classification and generative models.

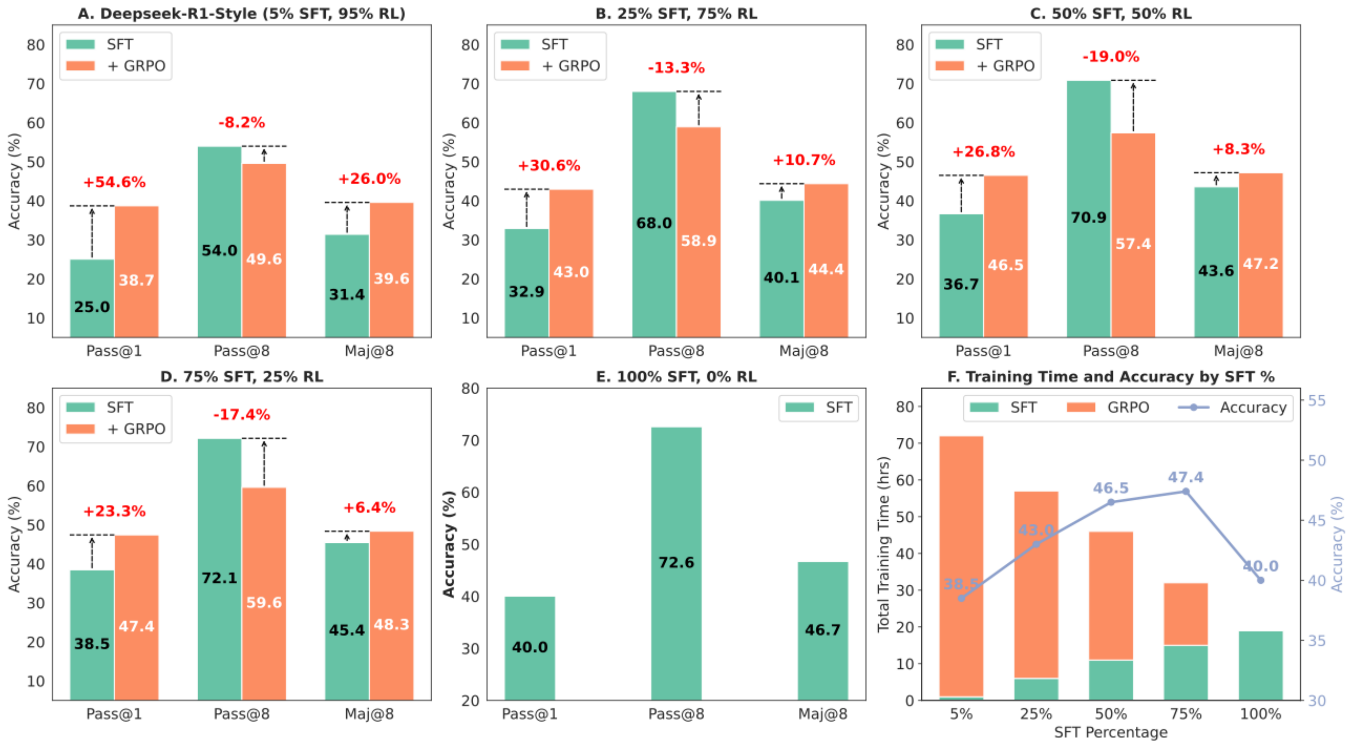
- **Log-Linear Scaling:** The final model's accuracy scales approximately linearly with the **logarithm of the SFT sample size**. Doubling SFT data leads to predictable gains.

Optimizing SFT vs. RL Data Allocation

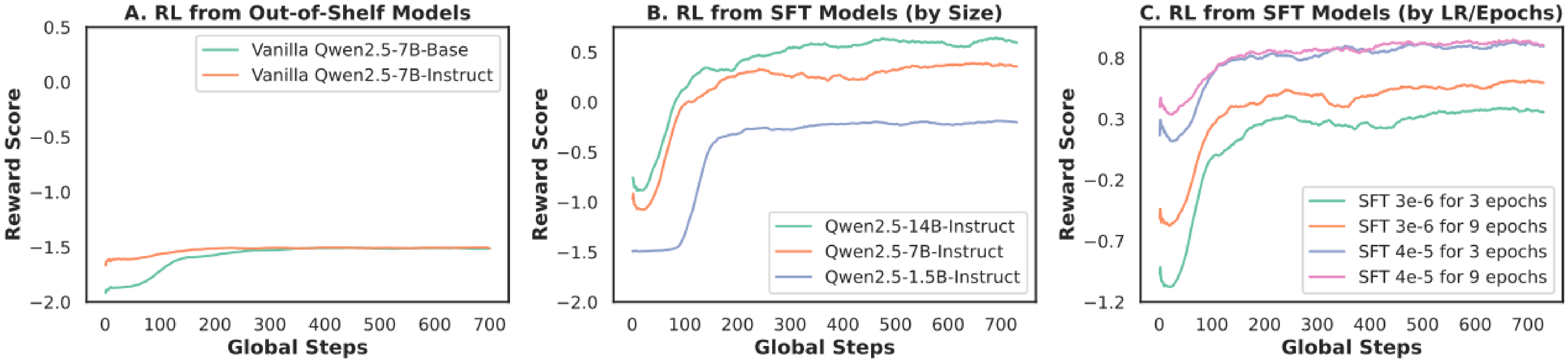
- **The "Bitter Lesson":** The key finding is that RL performance is fundamentally constrained by the knowledge the model gains from SFT)*before* RL begins.

- **RL Refines, Not Introduces New Capacity:** GRPO consistently improved the final accuracy (Pass@1) but often *reduced* the diversity of correct answers in the top 8 results (Pass@8).

- **Efficiency:** SFT is more computationally efficient than the costly generation steps required for GRPO.



Prerequisites for Effective RL Training



- **SFT is Mandatory:** Off-the-shelf, vanilla models **failed to learn** the DRG coding task with RL alone. They could mimic the output format but produced no correct answers.

- **Stronger SFT = Stronger RL:** The performance of the final RL model is directly tied to the quality of the SFT checkpoint it starts from.

- **Aggressive SFT Helps:** Using higher learning rates and more training epochs during the SFT phase led to better downstream performance after RL was applied.

Key Ablation Study Findings

- **Cognitive Pattern:** The model naturally converged to an **"Answer-First"** reasoning pattern. Attempts to enforce a "Think-First" (CoT-First) or "Differential-Thinking" style actually *degraded* performance, suggesting a direct prediction is more effective for this task.

- **KL Divergence:** Unlike in math reasoning, the KL divergence penalty was **critical for stable training**. Removing it often led to model collapse. A cosine decay schedule for the KL penalty proved beneficial in large-scale runs.

- **Reward Shaping:** A **"Strict Reward"** (only rewarding a fully correct final answer) outperformed "Dense Rewards" (giving partial credit for correct reasoning steps). This suggests denser signals may lead to suboptimal local optima.

- **Curriculum Learning:** Training on only **medium-difficulty cases** (removing both easy and hard examples) was found to be an effective strategy, aligning with findings in other domains

We present accuracy results from various ablation studies in Section 5.3, as shown in Figure 12.

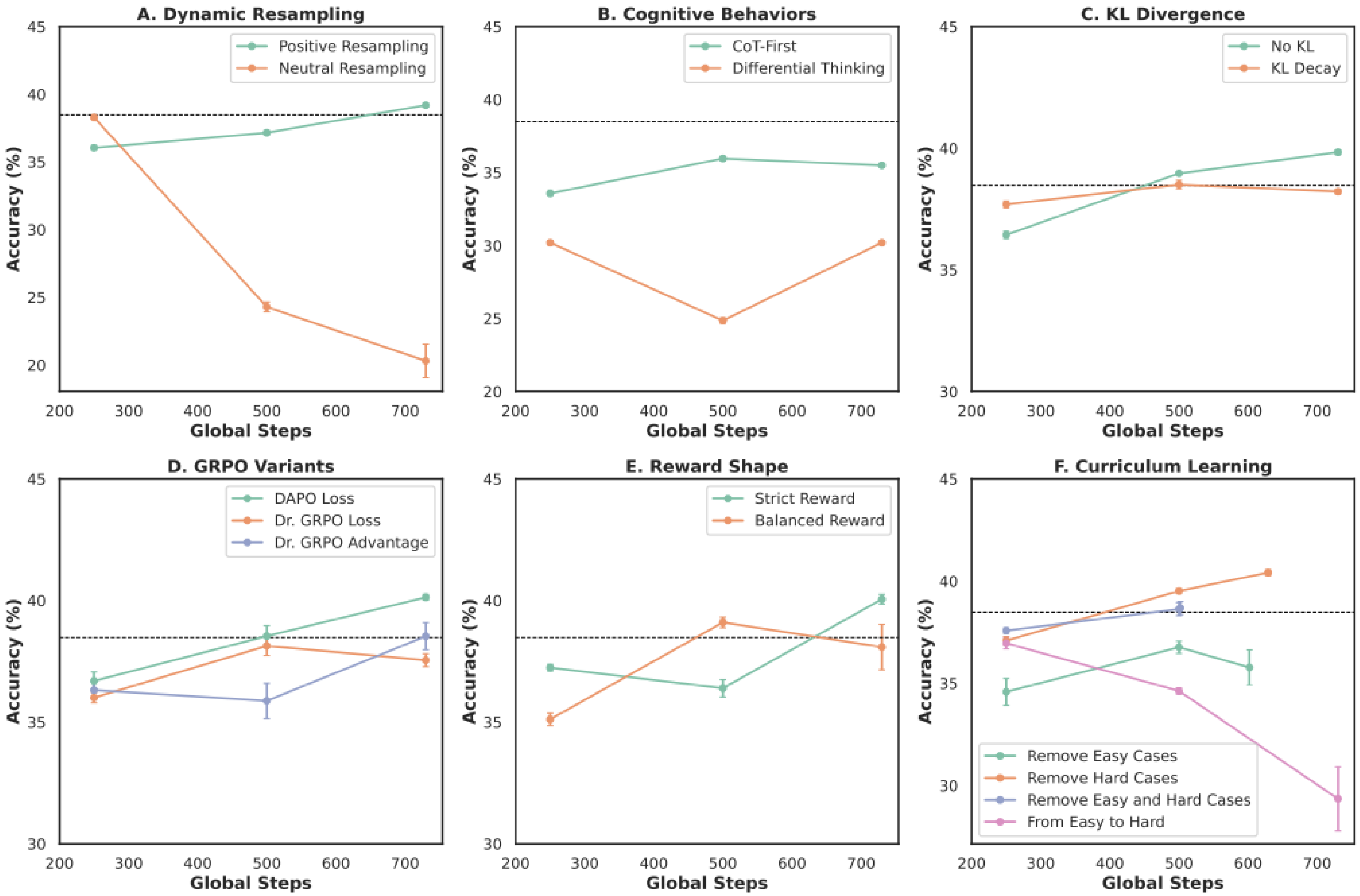


Figure 12: Accuracy with RL Training in Ablation Studies. The dashed line indicates the baseline performance of vanilla GRPO with dense rewards. Error bars indicate the standard deviation across 8 runs.

Conclusion & Takeaways

- We introduced **DRG-SAPPHIRE**, a new state-of-the-art model for the complex, out-of-distribution task of automated DRG coding.
- Our most critical finding is that for OOD tasks, **RL's success is constrained by the initial SFT model**. RL helps refine and sharpen the model's output but doesn't fundamentally create new reasoning abilities beyond what was learned in SFT.
- **Practical Implication:** To apply RL to new, specialized domains, the focus should be on building a strong SFT foundation first.

Thank you!

Questions?

- **Code & Data:** https://github.com/hanyin88/DRG-Sapphire
- **Contact:** wang.hanyin@mayo.edu, jimeng@illinois.edu