Brandon Howell
CSC 587

s1: Simple test-time scaling - Pros

This paper presents a model and methods aimed at replicating and improving upon language model reasoning capabilities by fine-tuning a non-reasoning model. The authors focus on achieving strong reasoning performance and test-time scaling with a simple and resource-efficient approach. I'm going to focus on their **dataset creation**, their **evaluation of test time scaling**, and **other miscellaneous things** they did well.

Firstly, let's focus on their **dataset**. The authors curated an initial dataset of 59K questions paired with reasoning traces down to 1K, which they named s1K. The curation process for s1K relied on three key principles: Quality, Difficulty, and Diversity. The **Quality** filter involved removing API errors and samples with formatting issues. **Difficulty** was assessed using model performance on questions by different smaller models and the length of the reasoning trace, using the assumption that longer traces tend toward more difficult problems, and questions solvable by either model were removed as potentially too easy. **Diversity** was measured by classifying questions into domains based on the Mathematics Subject Classification system, which includes topics in mathematics and other sciences like biology, physics, and economics. Ablation studies showed that combining Quality, Difficulty, and Diversity was crucial for sample-efficient reasoning training. Training on the full 59K dataset required substantially more resources (394 H100 GPU hours) but did not offer substantial gains over the carefully selected 1K sample dataset (7 H100 GPU hours for s1-32B training). This shows how **high quality data** has a huge impact on the fine-tuned model and results.

The paper also explores **test-time scaling**, which involves increasing compute at test time to improve performance. A key contribution is the development and analysis of **Budget Forcing**, a simple decoding-time intervention to control the model's thinking process. Budget Forcing can **enforce a maximum** number of thinking tokens by appending an **end-of-thinking token delimiter**, forcing the model to produce its answer. It can also **enforce a minimum** by suppressing the end-of-thinking token delimiter and **appending "Wait"** to encourage further reasoning. This technique can lead the model to double-check and fix incorrect reasoning steps. The authors propose **metrics** to evaluate test-time scaling methods, including **Control** (the ability to stay within a specified compute budget), **Scaling** (the slope of the accuracy-compute curve), and **Performance** (the accuracy achieved). Budget Forcing demonstrated perfect control over the compute budget and led to the best scaling and performance on the AIME24 benchmark compared to other methods like token-conditional control, step-conditional control, class-conditional control, and rejection sampling. The method showed **clear scaling trends** and could extrapolate performance, although it eventually flattened out. This is backed up by **other works**, such as the paper Incentivizing Reasoning Capability in Multimodal Large Language Models, which cited this paper's work as a contributor to "high-quality complex CoT reasoning and further advance[s in] the field by optimizing reasoning pathways" [1].

In terms of other pros of the paper, a significant aspect of this work is its open nature. The authors have made their model (s1-32B), the curated s1K dataset, and the code **open-source**. This contributes to transparency and fosters broader research progress in reasoning capabilities. Additionally, the paper's **scope** is characterized by its simplicity, seeking the simplest approach to achieve test-time scaling and strong reasoning. Next, the initial 59K dataset was **decontaminated** against the evaluation questions (MATH500, GPQA Diamond, AIME24) using 8-grams and deduplicated to prevent data leakage. The ability to fine-tune the Qwen2.5-32B-Instruct model on just 1,000 samples of s1K and achieve competitive reasoning performance with OpenAI's o1-preview in a **short training time** (~26 minutes on 16 NVIDIA H100 GPUs) is a major pro. **Chain-of-Thought** (CoT) prompting and related methods have been a significant line of work for enhancing language model reasoning, and this paper shows that models can be equipped with CoT abilities effectively and efficiently. Finally, many other papers, such as the Phi-4-Mini Technical Report use the paper as a building point, developing new ideas based on this work [2]. In total, this paper has almost 200 citations throughout other papers. **Overall**, the paper contributes a new open source dataset and Chain of Thought reasoning model derived through their analysis on test time scaling methods, making it highly significant for the field.

1. Vision-R1: Incentivizing Reasoning Capability in Multimodal Large Language Models
2. Phi-4-Mini Technical Report: Compact yet Powerful Multimodal Language Models via Mixture-of-LoRAs