

| | | |
|-----|---|-----------|
| 304 | Appendix Contents for SwS | |
| 305 | A Related Work | 11 |
| 306 | A.1 Reinforcement Learning for LLM Reasoning | 11 |
| 307 | A.2 Reasoning Data Synthesis | 12 |
| 308 | B Implementation Details | 12 |
| 309 | B.1 Training | 12 |
| 310 | B.2 Evaluation | 13 |
| 311 | C Co-occurrence Based Concept Sampling | 13 |
| 312 | D Data Analysis of the SwS Framework | 13 |
| 313 | D.1 Detailed Data Workflow | 13 |
| 314 | D.2 Difficulty Distribution of Synthetic Problems | 14 |
| 315 | E Details for Weak-to-Strong Generalization in SwS | 15 |
| 316 | F Details for Self-Evolving in SwS | 16 |
| 317 | G Details for Weakness-driven Selection | 17 |
| 318 | H Evaluation Benchmark Demonstrations | 18 |
| 319 | I Prompts | 19 |
| 320 | I.1 Prompt for Category Labeling | 19 |
| 321 | I.2 Prompt for Concepts Extraction | 22 |
| 322 | I.3 Prompt for Problem Synthetis | 22 |
| 323 | I.4 Prompt for Quality Evaluation | 23 |

Limitations and Future Work

This paper presents a comprehensive Self-aware Weakness-driven Problem Synthesis (SwS) framework to address model’s reasoning deficiencies through reinforcement learning (RL) training. Although the SwS framework is effective across a wide range of model sizes, employing both a strong instruction model and an answer-labeling reasoning model may lead to computation and time costs. Additionally, our work mainly focuses on the RL setting, as our primary goal is to mitigate model’s weakness by fully activating its inherent reasoning abilities without distilling external knowledge. Exploring how to leverage a similar pipeline for enhancing model capabilities through fine-tuning or distillation remains an open direction for future research.

In the future, we also aim to identify model weaknesses from multiple perspectives beyond simple answer accuracy, with the goal of synthesizing more targeted problems to improve sample efficiency. Additionally, we plan to extend the SwS framework to more general tasks beyond reasoning, incorporating an off-the-shelf reward model to provide feedback instead of verifiable answers.

A Related Work

A.1 Reinforcement Learning for LLM Reasoning

Recent advancements have led to significant integration of reinforcement learning with large language models (LLMs)[Ziegler et al., 2019, Ouyang et al., 2022], particularly in complex reasoning and coding tasks[Guo et al., 2025], with algorithms such as PPO [Schulman et al., 2017] and GRPO [Shao et al., 2024] demonstrating robustness and effectiveness. Compared to supervised fine-tuning (SFT) based on knowledge distillation, RL optimizes the model’s capabilities on its own generations using reward-based guidance, thereby promoting stronger generalization. In contrast, SFT models often depend on rote memorization of reasoning patterns and solutions [Chu et al., 2025], and may produce correct answers with flawed rationales [Wang et al., 2025]. In LLM reasoning, RL strengthens policy exploration and improves reasoning performance by using the verified correctness of the final answer in the responses as reward signals for training [Luong et al., 2024], which is commonly referred to as reinforcement learning with verifiable rewards (RLVR) [Yue et al., 2025].

RL Optimization. Recent efforts in improving RL optimization have focused on enhancing exploration [Yu et al., 2025a, Yuan et al., 2025, Liu et al., 2025b, Yeo et al., 2025] and adapting RL to the Long-CoT conditions [Jaech et al., 2024, Guo et al., 2025, Li et al., 2025b]. Yu et al. [2025a] found that the KL constraint may limit exploration under RLVR, while Liu et al. [2025b] proposed removing variance normalization in GRPO to prevent length bias. Building on PPO, Yuan et al. [2025] found that pre-training the value function prior to RL training and employing a length-adaptive GAE can improve training stability and efficiency in RLVR, preventing it from degrading to a constant baseline in value estimation.

Process Reward Modeling. In addition to answer-based reward, studies have also focused on leveraging process reward modeling to address reward sparsity in RL training [Cobbe et al., 2021, Lightman et al., 2023, Wang et al., 2023, Zhang et al., 2025], enabling a more fine-grained reward signal throughout the full solution [Wu et al., 2023]. Wang et al. [2023] successfully incorporated a process reward model (PRM), trained on process-level labels generated via Monte Carlo sampling at each step, into RL training and demonstrated its effectiveness. Beyond RL training, PRM can also be used to guide inference [Cobbe et al., 2021] and provide value estimates incorporated with search algorithms [Zhang et al., 2024, Guan et al., 2025]. However, Guo et al. [2025] found that the scalability of process-level RL is limited by the ambiguous definition of “step” and the high cost of process-level labeling. How to effectively scale process-level RL remains an open question.

Data Construction in RLVR. Although RL training on simpler mathematical questions can partially elicit a model’s reasoning ability [Zeng et al., 2025], the composition of RL training data is critical for enhancing the model’s reasoning capabilities [Luo et al., 2025, Yu et al., 2025a, Li et al., 2025a, Hu et al., 2025, He et al., 2025, Shen et al., 2025]. A well-curated problem set, with difficulty levels aligned with the model’s capabilities and a diverse distribution, is more likely to drive better performance. Implying a curriculum learning can also improve the efficiency [Shi et al., 2025]. In this work, we propose generating synthetic problems based on the model’s weaknesses for RL training, where the synthetic problems are tailored to align with the model’s capabilities and target its areas of weakness, fostering its exploration and improving performance.

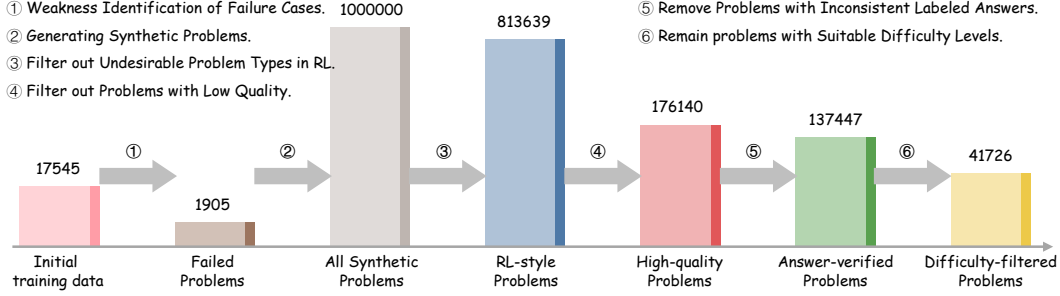


Figure 8: Demonstration of the SwS data workflow by tracing the process from initial training data to the final selection of synthetic problems in the 32B model experiments. For better visualization, the bar heights are scaled using the cube root of the raw data.

377 A.2 Reasoning Data Synthesis

378 Existing data synthesis strategies for enhancing LLM reasoning primarily focus on generating
 379 problem-response pairs [Huang et al., 2024, Tang et al., 2024, Yu et al., 2023, Zhao et al., 2025b,
 380 Liang et al., 2024, Luo et al., 2023, Liu et al., 2025a, Pei et al., 2025] or augmenting responses
 381 to existing questions [Toshniwal et al., 2024, Tong et al., 2024, He et al., 2025, Face, 2025, Wen
 382 et al., 2025, Yu et al., 2025b], utilizing advanced LLMs to produce these synthetic examples. A
 383 prominent line of work focuses on extracting and recombining key concepts from seed problems.
 384 KP-Math [Huang et al., 2024] and MathScale [Tang et al., 2024] decompose seed problems into
 385 underlying concepts and recombine them to generate new problems, using advanced models to
 386 produce corresponding solutions. PromptCoT [Zhao et al., 2025b] also leverages underlying concepts,
 387 but focuses on generating competition-level problems. DART-Math [Tong et al., 2024] introduces
 388 a difficulty-aware framework that prioritizes the diversity and richness of synthetic responses to
 389 challenging problems.

390 Recently, distillation from a stronger teacher model has been shown to be an effective shortcut for
 391 training smaller models [Guo et al., 2025]. Several works [Face, 2025, Ye et al., 2025, Muennighoff
 392 et al., 2025, Lu et al., 2025, Zhao et al., 2025a] employ advanced Long-CoT models to generate
 393 responses for distilling knowledge into smaller models. However, a significant disparity in capabilities
 394 between the teacher and student models can lead to hallucinations in the student’s outputs [Nguyen
 395 et al., 2025] and hinder generalization to out-of-distribution scenarios [Chu et al., 2025]. In contrast,
 396 our framework under the RL setting enables the model to identify and mitigate its own weaknesses
 397 by generating targeted synthetic problems from failure cases, enabling self-improvement without
 398 external knowledge distillation.

399 B Implementation Details

400 B.1 Training

401 We conduct our experiments using the verl [Sheng et al., 2024] framework and adopt GRPO [Shao
 402 et al., 2024] as the optimization algorithm. For all RL training experiments, we sample 8 rollouts
 403 per problem and use a batch size of 1024, with the policy update batch size set to 256. We employ
 404 a constant learning rate of 5×10^{-7} with a 20-step warm-up, and set the maximum prompt and
 405 response lengths to 1,024 and 8,192 tokens, respectively. We do not apply a KL penalty, as recent
 406 studies have shown it may hinder exploration and potentially cause training collapse [Yuan et al.,
 407 2025, Liu et al., 2025b, Yu et al., 2025a]. In the initial training stage, we train the model for 200
 408 steps. During augmented RL training, we continually train the initially trained model for 600 steps
 409 on the augmented dataset incorporated with synthetic problems, using only prompts with an accuracy
 410 between 10% and 90% as determined by the online policy model for updates.

411 Since the training data for the 32B and 14B models (a combination of DAPO [Yu et al., 2025a] and
 412 LightR1 [Wen et al., 2025] subsets) lack human-annotated category information, we leverage the
 413 LLaMA-3.3-70B-Instruct model to label their categories. This ensures consistency with our SwS
 414 pipeline, which combines concepts within the same category. The prompt is presented in Prompt 1.

B.2 Evaluation

For evaluation, we utilize the vLLM framework [Kwon et al., 2023] and allow for responses up to 8,192 tokens. For all the benchmarks, Pass@1 is computed using greedy decoding for baseline models and sampling (temperature 1.0, top-p 0.95) for RL-trained models. For Avg@32 on competition-level benchmarks, we sample 32 responses per model with the same sampling configuration as used in RL training. We adopt a hybrid rule-based verifier by integrating *Math-Verify* and the PRIME-RL verifier [Cui et al., 2025], as their complementary strengths lead to higher recall. For all the inference, we use the default chat template and enable CoT prompting by appending the instruction: “Let’s think step by step and output the final answer within “\boxed{” after each question.

C Co-occurrence Based Concept Sampling

Following Huang et al. [2024], Zhao et al. [2025b], we enhance the coherence and semantic fluency of synthetic problems by sampling concepts within the same category based on their co-occurrence probabilities and embedding similarities. Specifically, for each candidate concept $c \in \mathbf{C}$ from category \mathbf{D} , we define its score based on both co-occurrence statistics and embedding similarity as:

$$\text{Score}(c) = \begin{cases} \text{Co}(c) + \text{Sim}(c), & \text{if } c \notin \{c_1, c_2, \dots, c_k\} \\ -\infty, & \text{otherwise.} \end{cases}$$

The co-occurrence term $\text{Co}(c)$ is computed by summing the co-occurrence counts from a sparse matrix built over the entire corpus, generated by iterating through all available concept lists in the pool. For each list, we increment $\text{CooccurMatrix}[c, c']$ by one for every unordered pair where $c \neq c'$, yielding a sparse, symmetric matrix in which each entry $\text{CooccurMatrix}[c, c']$ records the total number of times concepts c and c' co-occur across all sampled lists:

$$\text{Co}(c) = \sum_{i=1}^k \text{CooccurMatrix}[c, c_i], \quad (3)$$

while the semantic similarity is given by the cosine similarity between the candidate’s embedding and the mean embedding of the currently selected concepts:

$$\text{Sim}(c) = \cos \left(\vec{e}_c, \frac{1}{k} \sum_{i=1}^k \vec{e}_{c_i} \right), \quad (4)$$

To efficiently support large-scale and high-dimensional concept spaces, we construct a sparse co-occurrence matrix over all unique concepts, where each entry represents the frequency with which a pair of concepts co-occurs within sampled concept lists. Simultaneously, concept embeddings are normalized and indexed via FAISS to facilitate fast similarity computation. During sampling, an initial seed concept is drawn in proportion to its empirical frequency. For each subsequent concept, scores are computed by efficiently summing its co-occurrence with the current set and its embedding similarity to the group mean, while previously selected concepts are masked out. The probability of sampling each candidate is determined via softmax over these scores with temperature τ :

$$P(c) = \frac{\exp(\text{Score}(c)/\tau)}{\sum_{c' \notin \{c_1, \dots, c_k\}} \exp(\text{Score}(c')/\tau)}. \quad (5)$$

This process iteratively constructs coherent, semantically related concept sets to serve as the inputs for synthetic problem generation, ensuring both diversity and fluency.

D Data Analysis of the SwS Framework

D.1 Detailed Data Workflow

Taking the 32B model experiments as an example, Figure 8 shows the comprehensive data workflow of the SwS framework, from identifying model weaknesses in the initial training data to the processing of synthetic problems. The initial training set, consisting of the DAPO and Light-R1 subsets for the

Positive Case # 1: Let z_1 , z_2 , and z_3 be complex numbers such that $|z_1| = |z_2| = |z_3| = 1$ and $z_1 + z_2 + z_3 = 0$. Using the symmetric polynomial $s_2 = z_1z_2 + z_1z_3 + z_2z_3$, find the value of $|s_2|^2$.

Negative Case # 1: In a village, there are 10 houses, each of which can be painted one of three colors: red, blue, or green. Two houses cannot have the same color if they are directly adjacent to each other. Using combinatorial analysis and considering the constraints, find the total number of distinct ways to paint the houses, taking into account the possibility of having a sequence where the same color repeats after two different colors (e.g., red, blue, red), and assuming that the color of one of the end houses is already determined to be red, and the colors of the houses are considered different based on their positions (i.e., the configuration red, blue, green is considered different from green, blue, red).

Negative Case # 2: A metal’s surface requires a minimum energy of 2.5 eV to remove an electron via the photoelectric effect. If light with a wavelength of 480 nm is shone on the metal, and 1 mole of electrons is ejected, what is the total energy, in kilojoules, transferred to the electrons, given that the energy of a photon is related to its wavelength by the formula $E = hc/\lambda$, where $h = 6.626 \times 10^{-34}$ J s and $c = 3.00 \times 10^8$ m/s, and Avogadro’s number is 6.02×10^{23} particles per mole?

Negative Case # 3: In triangle ABC , with $\angle A = 60^\circ$, $\angle B = 90^\circ$, $AB = 4$, and $BC = 7$, use the Law of Sines to find $\angle C$ and calculate the triangle’s area.

Table 4: Case study of quality filtering results in SwS, featuring one high-quality positive case and three low-quality negative cases. The low-quality segments are marked in pink.

451 Qwen2.5-32B model, contains 17,545 problem-answer pairs. During the weakness identification
 452 stage, 1,905 problems are identified as failure cases according to Eq. 1. These failure cases are
 453 subsequently used for concept extraction and targeted problem synthesis.

454 For problem synthesis, we set an initial budget of 1 million synthetic problems in all experiments,
 455 with allocations for each category determined as in Eq. 2. These problems then undergo several
 456 filtering stages: (1) removing multiple-choice, multi-part, or proof-required problems; (2) discarding
 457 problems evaluated as low quality; (3) filtering out problems where the answer generation model
 458 yields inconsistent answers, specifically when the most frequent answer among all generations appears
 459 less than 50%; and (4) removing problems whose difficulty levels are unsuitable for the current model
 460 in RL training. Among these, the quality-based filtering is the strictest, with a filtering rate of 78.35%,
 461 indicating that the SwS pipeline maintains rigorous quality control over the generated problems. This
 462 ensures both the stability and effectiveness of utilizing synthetic problems in subsequent training.

463 We present a case study of the quality-based filtering results in Table 4. As illustrated, the positive case
 464 that passed the model-based quality evaluation features a concise and precise problem description. In
 465 contrast, most synthetic problems identified as low-quality exhibit redundant and overly elaborate
 466 descriptions, sometimes including lengthy hints for solving the problem, as seen in the first negative
 467 case. Additionally, some low-quality problems incorporate excessive non-mathematical knowledge,
 468 such as Physics, as illustrated in the second negative case. The informal *LaTeX* formatting also
 469 contributes to their lower quality. Furthermore, problems with multiple question components, such as
 470 the third negative case, are also considered as low quality for RL training.

471 D.2 Difficulty Distribution of Synthetic Problems

472 In this section, we study the difficulty distribution of the synthetic problems generated for base models
 473 ranging from 3B to 32B, as shown in Figure 9. The red outlines in the pie plots highlight the subset
 474 of synthetic problems selected for subsequent augmented RL training, with accuracy falling within
 475 the [25%, 75%] range. These samples account for nearly 35% of all generated problems across the
 476 four models. The two largest wedges in the pie chart represent problems that the models answered
 477 either completely correctly or completely incorrectly. These cases do not provide effective training
 478 signals in GRPO [Shao et al., 2024, Yu et al., 2025a], and are thus excluded from the later augmented
 479 RL training stage. To further enhance stability and efficiency, we also exclude problems where the
 480 model produces only one correct or one incorrect response.

481 Since all synthetic problems are generated using the same instruction model (LLaMA-3.3-70B-
 482 Instruct) with similar competition-level difficulty levels (as illustrated in Prompt 3), and are based on

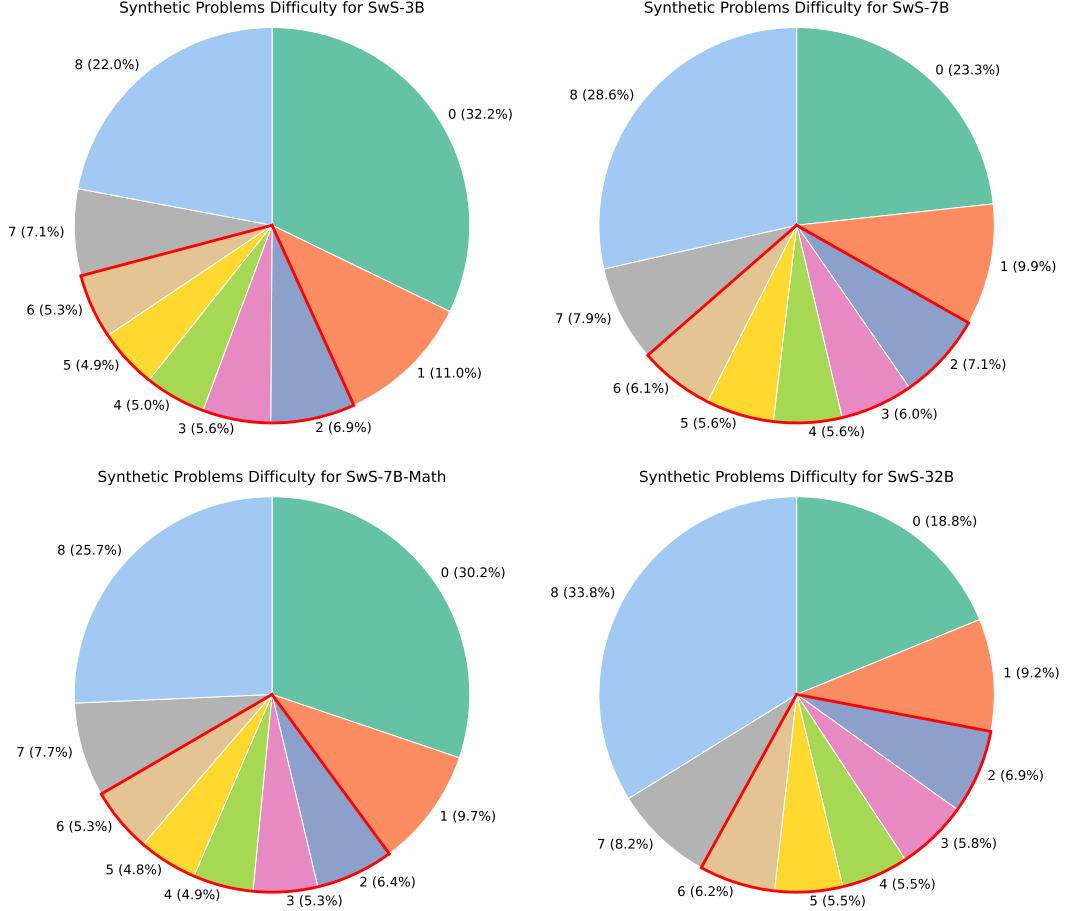


Figure 9: Difficulty distributions of synthetic problems for models from 3B to 32B in our work.

concepts derived from their respective weaknesses, the resulting difficulty distribution of the synthetic problems exhibits only minor differences across all models. Consistent with intuition, the initially trained 3B model achieved the lowest performance on the synthetic questions, with the highest ratio of all-incorrect and the lowest ratio of all-correct responses, while the 32B model showed the opposite trend, achieving the best performance.

E Details for Weak-to-Strong Generalization in SwS

To understand the capabilities of the weak teacher and the strong student model, we evaluated both of them on the MATH-500 test set by prompting them on each question for eight times. Although the teacher model generally exhibits weaker performance, we found that in 16.4% of problems, the weaker teacher outperforms the otherwise stronger student model. This highlights the potential for leveraging a weak teacher to distill its strengths into the student model. A case where the weaker teacher model outperforms the stronger student model is shown in Figure 10.

From the analysis of the SwS framework, as well as its *Weak-to-Strong Generalization* extension, we assert that the upper bound for answer labeling is a revised form of self-consistency score of the teacher model, where (1) the consistent answer must achieve an accuracy greater than 50% across all responses, and (2) the student model must provide the same answer as the teacher model’s consistent answer in at least 25% of responses. These revision procedures help ensure the correctness of the synthetic problem answers labeled by the teacher model.

In Table 5, we demonstrate the robustness of utilizing a weaker teacher for answer labeling, assuming that the MATH500 test set serves as our synthetic problems. As in the second line, even under the self-consistency setting, the teacher model only achieves an improvement of 4.8 points.

| Case Study for Weak-to-strong Generalization | | | | | | | | | |
|---|--|--|--|--|--|--|--|--|--|
| Question: Compute $99^2 + 99 + 1$ in your head. Please reason step by step, and put your final answer within <code>\boxed{}</code> . Golden Answer: <code>9901</code> | | | | | | | | | |
| >>> Teacher Response: | | | | | | | | | |
| To compute $99^2 + 99 + 1$, we can compute each part directly: | | | | | | | | | |
| $99^2 = 9801$ | | | | | | | | | |
| $99^2 + 99 = 9801 + 99 = 9900$ | | | | | | | | | |
| $99^2 + 99 + 1 = 9900 + 1 = 9901$ | | | | | | | | | |
| Final answer: <code>9901</code> | | | | | | | | | |
| >>>> Student Response: | | | | | | | | | |
| Rewrite the expression using a binomial identity: | | | | | | | | | |
| $99^2 + 99 + 1 = (99 + 1)^2 = 100^2 = 10000$ | | | | | | | | | |
| Final answer: <code>10000</code> | | | | | | | | | |

Figure 10: Case study in which a weak teacher provides the correct solution, while a strong student incorrectly applies a binomial identity and derives an incorrect answer.

| Setting | Size | Prealgebra | Intermediate Algebra | Algebra | Precalculus | Number Theory | Counting & Probability | Geometry | All |
|--------------------|------|------------|----------------------|---------|-------------|---------------|------------------------|----------|------|
| Pass@1 | 500 | 88.2 | 64.3 | 95.5 | 71.2 | 93.0 | 81.4 | 63.0 | 80.6 |
| + SC | 500 | 96.9 | 96.0 | 84.4 | 84.1 | 96.2 | 87.5 | 67.8 | 85.4 |
| + SC>50% | 444 | 96.9 | 97.3 | 93.2 | 94.7 | 98.0 | 94.4 | 89.6 | 94.4 |
| + SC>50% & Stu-Con | 407 | 96.8 | 97.2 | 97.7 | 100.0 | 100.0 | 96.8 | 94.9 | 97.5 |

Table 5: The performance of the weak teacher model used for answer generation on the MATH-500 test set under different strategies and their corresponding revisions. "Stu-Con" refers to filtering out problems where the student model’s accuracy falls below the defined threshold of 25%.

504 However, when we exclude problems for which self-consistency does not provide sufficient confidence—specifically, those where the most consistent answer accounts for less than 50% of all
505 responses—the self-consistency setting yields an additional 9.0-point improvement on the remaining
506 questions. Furthermore, in our SwS pipeline, we retain only problems where the student model
507 achieves over 25% accuracy to ensure an appropriate level of difficulty. After filtering out problems
508 where the student falls below this threshold, some mislabeled problems are also automatically
509 removed, resulting in the weak teacher achieving a performance of 97.5% on the final remaining
510 questions. The increase in labeling accuracy from 80.6% to 97.5% shows the potential of utilizing
511 the weaker teacher model for answer labeling as well as the robustness of the SwS framework itself.
512

513 F Details for Self-Evolving in SwS

514 As mentioned in Section 4.2, the *Self-evolving* SwS extension enables the policy to achieve better
515 performance on simple to medium-level mathematical reasoning benchmarks but remains suboptimal
516 on AIME-level competition benchmarks. In this section, we further analyze the reasons behind
517 this phenomenon. Figure 11 visualizes the model’s self-quality assessment and difficulty evaluation
518 within the SwS framework. Notably, the model assigns a much higher proportion of “perfect” and
519 “acceptable” labels, and fewer “bad” labels, to its self-generated problems compared to the standard
520 framework shown in Figure 8. This observation is consistent with findings from LLM-as-a-Judge [Li
521 et al., 2024b], which indicate that models tend to be more favorable toward and assign higher scores
522 to their own generations. Such behavior may result in overlooking low-quality problems or mis-
523 classifying problems that are too complex for the model’s reasoning abilities as unsolvable or of
524 poor quality. Beyond the risk of filtering out over-complex problems, the model may also have
525 difficulty in accurately labeling answers for over-challenging problems, thereby limiting the potential
526 of incorporating complex problems through the *Self-evolving* SwS framework.

527 Additionally, the initially trained model achieves nearly 50% all-correct responses on its generated
528 problems, whereas only 31% of problems remain after SwS difficulty filtering. This suggests that
529 the self-generated problems may be significantly simpler than those produced using a stronger

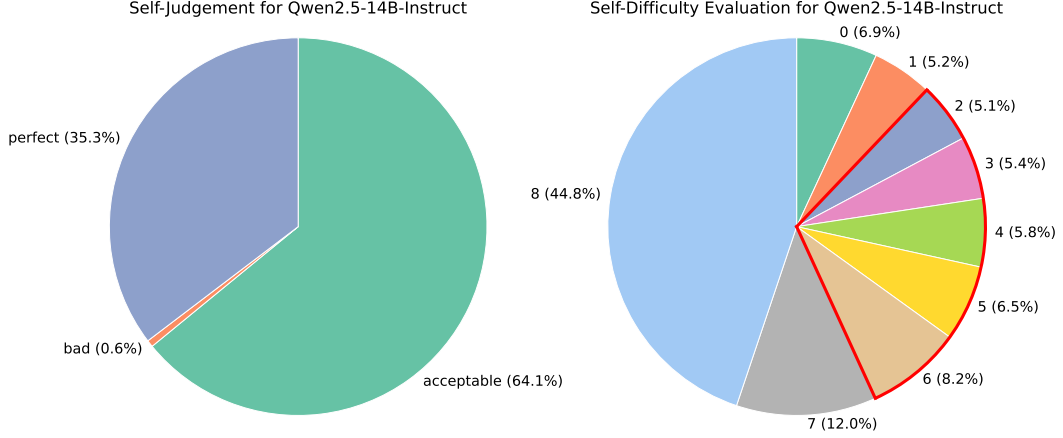


Figure 11: Illustration of the quality assessment and difficulty evaluation for Qwen2.5-14B-Instruct under the *Self-evolving* SwS framework.

Algorithm 1 Weakness-Driven Selection Pipeline

Require: Failed Problems \mathbf{X}_S ; Total Budget $|T|$; Target Set \mathbf{T}_X ; Domains $\{\mathbf{D}_i\}_{i=0}^n$

Ensure: Selected problems \mathbf{T}_S

- 1: **Embed** all failed problems in \mathbf{X}_S and all questions in \mathbf{T}_X
 - 2: **for** each domain \mathbf{D}_i in $\{\mathbf{D}_i\}_{i=0}^n$ **do**
 - 3: **Compute selection budget** $|T_i|$ for \mathbf{D}_i according to Eq. 2
 - 4: **Extract** failed problems $\mathbf{X}_{S,i}$ belonging to \mathbf{D}_i
 - 5: **for** each $q \in \mathbf{T}_X$ **do** ▷ Domain-level KNN
 - 6: Compute $d_i(q) = \min_{f \in \mathbf{X}_{S,i}} \text{distance}(\vec{e}_q, \vec{e}_f)$
 - 7: **end for**
 - 8: **Select top** $|T_i|$ questions from \mathbf{T}_X with the smallest $d_i(q)$ as \mathcal{S}_i
 - 9: **end for**
 - 10: **return** Selected problems $\mathbf{T}_S = \bigcup_{i=0}^n \mathcal{S}_i$ ▷ Final Selected Set
-

530 instruction model [Grattafiori et al., 2024], thus it could lead to data inefficiency and limit the model’s
531 performance on more complex problems during RL training.

532 G Details for Weakness-driven Selection

533 As described in Section 4.3, we utilize the failed problems identified by Qwen2.5-7B [Yang et al.,
534 2024a] on the MATH-12k [Hendrycks et al., 2021] training set, which comprises 915 problems,
535 to select additional data from Big-Math [Albalak et al., 2025] to mitigate the model’s weaknesses
536 through the augmented RL training. The complete *Weakness-driven Selection* extension of SwS is
537 presented in Algorithm 1. For the data embedding, we use LLaMA-3.1-8B-base [Grattafiori et al.,
538 2024] to embed both the collected failure cases and the problems from the target dataset. The failure
539 cases are then grouped by categories, following the concept sampling strategy in standard SwS. We
540 employ a binary *K-Nearest Neighbors* [Cover and Hart, 1967] algorithm to select weakness-driven
541 problems from the target set, based on their embedding distances to the failure cases within each
542 category. The selection budget for each category is also determined according to Eq. 2. We then
543 aggregate the retrieved problems from all categories, forming a selected set of 40k problems, which
544 are then incorporated with the initial set for the subsequent RL training.

545 H Evaluation Benchmark Demonstrations

| Dataset | Size | Category | Example Problem | Answer |
|----------------|------|---------------|--|------------------|
| GSM8k | 1319 | Prealgebra | The ice cream parlor was offering a deal, buy 2 scoops of ice cream, get 1 scoop free. Each scoop cost \$1.50. If Erin had \$6.00, how many scoops of ice cream should she buy? | 6 |
| MATH-500 | 500 | Geometry | For a constant c , in cylindrical coordinates (r, θ, z) , find the shape described by the equation $z = c$. (A) Line (B) Circle (C) Plane (D) Sphere (E) Cylinder (F) Cone. Enter the letter of the correct option. | (C) Plane |
| Minerva Math | 272 | Precalculus | If the Bohr energy levels scale as Z^2 , where Z is the atomic number of the atom (i.e., the charge on the nucleus), estimate the wavelength of a photon that results from a transition from $n = 3$ to $n = 2$ in Fe, which has $Z = 26$. Assume that the Fe atom is completely stripped of all its electrons except for one. Give your answer in Angstroms, to two significant figures. | 9.6 |
| Olympiad-Bench | 675 | Geometry | Given a positive integer n , determine the largest real number μ satisfying the following condition: for every $4n$ -point configuration C in an open unit square U , there exists an open rectangle in U , whose sides are parallel to those of U , which contains exactly one point of C , and has an area greater than or equal to μ . | $\frac{1}{2n+2}$ |
| Gaokao2023 | 385 | Geometry | There are three points A, B, C in space such that $AB = BC = CA = 1$. If 2 distinct points are chosen in space such that they, together with A, B, C , form the five vertices of a regular square pyramid, how many different ways are there to choose these 2 points? | 9 |
| AMC23 | 40 | Algebra | How many complex numbers satisfy the equation $z^5 = \bar{z}$, where \bar{z} is the conjugate of the complex number z ? | 7 |
| AIME24 | 30 | Number Theory | Let N be the greatest four-digit positive integer with the property that whenever one of its digits is changed to 1, the resulting number is divisible by 7. Let Q and R be the quotient and remainder, respectively, when N is divided by 1000. Find $Q + R$. | 699 |
| AIME25 | 30 | Geometry | On $\triangle ABC$ points A, D, E , and B lie that order on side \overline{AB} with $AD = 4, DE = 16$, and $EB = 8$. Points A, F, G , and C lie in that order on side \overline{AC} with $AF = 13, FG = 52$, and $GC = 26$. Let M be the reflection of D through F , and let N be the reflection of G through E . Quadrilateral $DEGF$ has area 288. Find the area of heptagon $AFNBCEM$. | 588 |

Table 6: Statistics and examples of the eight evaluation benchmarks utilized in the paper.

546 We present statistics and examples of the eight evaluation benchmarks utilized in our work in Table 6.
547 Among these, the GSM8K [Cobbe et al., 2021] benchmark is the simplest one, which consists of
548 grade school math word problems that require multi-step reasoning and basic arithmetic. The MATH-
549 500 [Hendrycks et al., 2021] and Gaokao2023 [Zhang et al., 2023] benchmarks include high school
550 problems covering a wide range of topics and difficulty levels, while Minerva Math [Lewkowycz
551 et al., 2022] may additionally include problems from other subjects, such as physics, with comparable
552 difficulty to MATH-500. The AIME [MAA, b] benchmarks are from the American Invitational
553 Mathematics Examination, a prestigious high school mathematics competition for top-performing
554 students, with each problem requiring deep mathematical insight and precise problem-solving skills.
555 Olympiad-Bench [He et al., 2024] and AMC23 [MAA, a] also consist of competition-level math
556 problems, with difficulty levels between those of MATH-500 and AIME.

I Prompts

I.1 Prompt for Category Labeling

Listing 1: The prompt for labeling the categories for mathematical problems, utilizing a few-shot strategy in which each category is represented by a labeled demonstration.

```
559 # CONTEXT #
560 I am a teacher, and I have some high-level mathematical problems.
561 I want to categorize the domain of these math problems.
562
563 # OBJECTIVE #
564 A. Provide a concise summary of the math problem, clearly identifying
565 the key concepts or techniques involved.
566 B. Assign the problem to one and only one specific mathematical domain
567 .
568 The following is the list of domains to choose from:
569 <math domains>
570 ["Intermediate Algebra", "Geometry", "Precalculus", "Number Theory", "
571 Counting & Probability", "Algebra", "Prealgebra"]
572 </math domains>
573
574 # STYLE #
575 Data report.
576
577 # TONE #
578 Professional, scientific.
579
580 # AUDIENCE #
581 Students. Enable them to better understand the domain of the problems.
582
583 # RESPONSE: MARKDOWN REPORT #
584 ## Summarization
585 [Summarize the math problem in a brief paragraph.]
586 ## Math domains
587 [Select one domain from the list above that best fits the problem.]
588
589 # ATTENTION #
590 - You must assign each problem to exactly one of the domains listed
591 above.
592 - If you are genuinely uncertain and none of the listed categories
593 applies, you may use "Other", but this should be a last resort.
594 - Be thoughtful and accurate in your classification. Default to the
595 listed categories whenever possible.
596 - Add "=== report over ===" at the end of the report.
597
598 <example math problem>
599 **Question**:
600 Let  $n(\geq 2)$  be a positive integer. Find the minimum  $m$ , so that
601 there exists  $x_{ij}(1 \leq i, j \leq n)$  satisfying:
602 (1) For every  $1 \leq i \leq n$ ,  $x_{ij} = \max\{x_{i1}, x_{i2}, \dots, x_{in}\}$ 
603 or  $x_{ij} = \max\{x_{1j}, x_{2j}, \dots, x_{nj}\}$ .
604 (2) For every  $1 \leq i \leq n$ , there are at most  $m$  indices  $k$  with
605  $x_{ik} = \max\{x_{i1}, x_{i2}, \dots, x_{in}\}$ .
606 (3) For every  $1 \leq j \leq n$ , there are at most  $m$  indices  $k$  with
607  $x_{kj} = \max\{x_{1j}, x_{2j}, \dots, x_{nj}\}$ .
608 </example math problem>
609
610 ## Summarization
611 The problem involves an  $(n \times n)$  matrix where each element  $x_{ij}$ 
612 is constrained by the maximum values in its respective row
613 or column. The goal is to determine the minimum possible value of  $m$ 
614 such that, for each row and column, the number of indices
615 attaining the maximum value is limited to at most  $m$ . This
```

```

618 problem requires understanding matrix properties, maximum functions,
619 and combinatorial constraints on structured numerical arrangements.
620
621 ## Math domains
622 Algebra
623
624 === report over ===
625
626 </example math problem>
627 **Question**:
628 In an acute scalene triangle  $ABC$ , points  $D, E, F$  lie on sides  $BC$ ,
629  $CA$ ,  $AB$ , respectively, such that  $AD \perp BC$ ,  $BE \perp CA$ ,  $CF \perp$ 
630  $AB$ . Altitudes  $AD$ ,  $BE$ ,  $CF$  meet at orthocenter  $H$ . Points  $P$  and
631  $Q$  lie on segment  $EF$  such that  $AP \perp EF$  and  $HQ \perp EF$ .
632 Lines  $DP$  and  $QH$  intersect at point  $R$ . Compute  $HQ/HR$ .
633 </example math problem>
634
635 ## Summarization
636 The problem involves an acute scalene triangle with three
637 perpendicular cevians intersecting at the orthocenter. Additional
638 perpendicular constructions are made from specific points on segment
639  $(EF)$ , leading to an intersection at point  $(R)$ . The goal is to
640 determine the ratio  $(HQ/HR)$ , requiring knowledge of triangle
641 geometry, perpendicularity, segment ratios, and properties of the
642 orthocenter.
643
644 ## Math domains
645 Geometry
646
647 === report over ===
648
649 </example math problem>
650 **Question**:
651 Three cards are dealt at random from a standard deck of 52 cards.
652 What is the probability that the first card is a 4, the second card is
653 a  $\clubsuit$ , and the third card is a 2?
654 </example math problem>
655
656 ## Summarization
657 This problem involves calculating the probability of a specific
658 sequence of events when drawing three cards from a standard 52-card
659 deck without replacement. It requires understanding conditional
660 probability, the basic rules of counting, and how probabilities change
661 as cards are removed from the deck.
662
663 ## Math domains
664 Counting & Probability
665
666 === report over ===
667
668 </example math problem>
669 **Question**:
670 Let  $x$  and  $y$  be real numbers such that  $3x + 2y \leq 7$  and  $2x + 4y$ 
671  $\leq 8$ . Find the largest possible value of  $x + y$ .
672 </example math problem>
673
674 ## Summarization
675 This problem involves optimizing a linear expression  $(x + y)$ 
676 subject to a system of linear inequalities. It requires understanding
677 of linear programming concepts, such as identifying feasible regions,
678 analyzing boundary points, and determining the maximum value of an
679 objective function within that region.
680
681 ## Math domains
682 Intermediate Algebra

```

```

683
684 === report over ===
685
686 </example math problem>
687 **Question**:
688 Solve
689  $\arccos 2x - \arccos x = \frac{\pi}{3}$ . Enter all the solutions,
690 separated by commas.
691 </example math problem>
692
693 ## Summarization
694 This problem requires solving a trigonometric equation involving
695 inverse cosine functions. The equation relates two expressions with  $\arccos(2x)$  and  $\arccos(x)$ , and asks for all real solutions
696 satisfying the given identity. It involves knowledge of inverse
697 trigonometric functions, their domains, and properties, as well as
698 algebraic manipulation.
699
700
701 ## Math domains
702 Precalculus
703
704 === report over ===
705
706 </example math problem>
707 **Question**:
708 What perfect-square integer is closest to 273?
709 </example math problem>
710
711 ## Summarization
712 The problem asks for the perfect square integer closest to 273. This
713 involves understanding the distribution and properties of perfect
714 squares, and comparing them with a given integer. It relies on number-
715 theoretic reasoning related to squares of integers and their proximity
716 to a target number.
717
718 ## Math domains
719 Number Theory
720
721 === report over ===
722
723 </example math problem>
724 Voldemort bought  $\overline{6}$  ounces of ice cream at an ice cream
725 shop. Each ounce cost  $\$0.60$ . How much money, in dollars, did he
726 have to pay?
727 </example math problem>
728
729 ## Summarization
730 The problem involves multiplying a repeating decimal,  $\overline{6}$ , by a fixed unit price,  $\$0.60$ , to find the total cost in
731 dollars. This requires converting a repeating decimal into a fraction
732 or using decimal multiplication, both of which are foundational
733 arithmetic skills.
734
735
736 ## Math domains
737 Prealgebra
738
739 === report over ===
740
741 <math problem>
742 {problem}
743 </math problem>

```

I.2 Prompt for Concepts Extraction

Listing 2: Prompt template for extracting internal concepts from a mathematical question.

```
As an expert in educational assessment, analyze this problem:
<problem>
{problem}
</problem>

Break down and identify {num_concepts} foundational concepts being
tested. List these knowledge points that:
- Are core curriculum concepts typically taught in standard courses,
- Are precise and measurable (not vague like "understanding math"),
- Are essential building blocks needed to solve this problem,
- Represent fundamental principles rather than problem-specific
techniques.

Think through your analysis step by step, then format your response as
a Python code snippet containing a list of {num_concepts} strings,
where each string clearly describes one fundamental knowledge point.
```

I.3 Prompt for Problem Synthetis

Listing 3: Prompt template for synthesizing math problems from specified concepts, difficulty levels, and pre-defined mathematical categories. Following [Zhao et al., 2025b], the difficulty levels are consistently set to the competition level to prevent the generation of overly simple questions.

```
### Given a set of foundational mathematical concepts, a mathematical
domain, and a specified difficulty level, generate a well-constructed
question that meaningfully integrates multiple listed concepts and
reflects the stated level of complexity.

### Foundational Concepts:
{concepts}

### Target Difficulty Level:
{level}

### Mathematical Domain:
{domain}

### Instructions:
1. Begin by outlining which concepts you will combine and how you plan
to structure the question.
2. Ensure that the question is coherent, relevant, and appropriately
challenging for the specified level.
3. The question must be a single standalone problem, not split into
multiple sub-questions.
4. Do not generate proof-based, multiple-choice, or true/false
questions.
5. The answer to the question should be expressible using numbers and
mathematical symbols.
6. Provide a final version of the question that is polished and ready
for use.

### Output Format:
- First, provide your brief outline and planning for the question
design.
- Then, present only the final version of the question in the
following format:

'''
[Your developed question here]
```

```

802  '''
803
804  Do not include any placeholder, explanatory text, hints, or solutions
805  to the question in the output block

```

807 I.4 Prompt for Quality Evaluation

Listing 4: The quality evaluation prompt utilized to filter out low-quality math problems. Following prior work [Zhao et al., 2025b], we assess synthetic problems based on five criteria: **format, factual accuracy, difficulty alignment, concept coverage, and solvability**. Each problem is then assigned one of three quality levels: ‘bad’, ‘acceptable’, or ‘perfect’.

```

808
809  As a critical expert in educational problem design, evaluate the
810  following problem components:
811
812  === GIVEN MATERIALS ===
813  1. Problem & Design Rationale:
814  {rationale_and_problem}
815  (The rationale describes the author’s thinking process and
816  justification in designing this problem)
817
818  2. Foundational Concepts:
819  {concepts}
820
821  3. Target Difficulty Level:
822  {level}
823
824
825  === EVALUATION CRITERIA ===
826  Rate each criterion as: [Perfect | Acceptable | Bad]
827  1. FORMAT
828  - Verify correct implementation of markup tags:
829  <!-- BEGIN RATIONALE -> [design thinking process] <!-- END RATIONALE ->
830  <!-- BEGIN PROBLEM -> [problem] <!-- END PROBLEM ->
831
832  2. FACTUAL ACCURACY
833  - Check for any incorrect or misleading information in both problem
834  and rationale
835  - Verify mathematical, scientific, or logical consistency
836
837  3. DIFFICULTY ALIGNMENT
838  - Assess if problem complexity matches the specified difficulty level
839  - Evaluate if cognitive demands align with target level
840
841  4. CONCEPT COVERAGE
842  - Evaluate how well the problem incorporates the given foundational
843  concepts
844  - Check for missing concept applications
845
846  5. SOLVABILITY
847  - Verify if the problem has at least one valid solution
848  - Check if all necessary information for solving is provided
849
850  === RESPONSE FORMAT ===
851  For each criterion, provide:
852  1. Rating: [Perfect | Acceptable | Bad]
853  2. Justification: Clear explanation for the rating
854
855  === FINAL VERDICT ===
856  After providing all criterion evaluations, conclude your response with
857  :
858  ‘Final Judgement: [verdict]’
859  where verdict must be one of:

```

860 - ‘perfect’ (if both FACTUAL ACCURACY and SOLVABILITY are Perfect, at
861 least two other
862 criteria are Perfect, and no Bad ratings)
863 - ‘acceptable’ (if no Bad ratings and doesn’t qualify for perfect)
864 - ‘bad’ (if ANY Bad ratings)
865
866 Note: The ‘Final Judgement: [verdict]’ line must be the final line of
867 your response.
868

869 References

- 870 Alon Albalak, Duy Phung, Nathan Lile, Rafael Rafailov, Kanishk Gandhi, Louis Castricato, Anikait
871 Singh, Chase Blagden, Violet Xiang, Dakota Mahan, et al. Big-math: A large-scale, high-quality
872 math dataset for reinforcement learning in language models. *arXiv preprint arXiv:2502.17387*,
873 2025.
- 874 Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner,
875 Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, et al. Weak-to-strong generalization:
876 Eliciting strong capabilities with weak supervision. *arXiv preprint arXiv:2312.09390*, 2023.
- 877 Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V
878 Le, Sergey Levine, and Yi Ma. Sft memorizes, rl generalizes: A comparative study of foundation
879 model post-training. *arXiv preprint arXiv:2501.17161*, 2025.
- 880 Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser,
881 Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve
882 math word problems. *arXiv preprint arXiv:2110.14168*, 2021.
- 883 Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on*
884 *information theory*, 13(1):21–27, 1967.
- 885 Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu
886 Yu, Qixin Xu, Weize Chen, et al. Process reinforcement through implicit rewards. *arXiv preprint*
887 *arXiv:2502.01456*, 2025.
- 888 Hugging Face. Open r1: A fully open reproduction of deepseek-r1, January 2025. URL <https://github.com/huggingface/open-r1>.
889
- 890 Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad
891 Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of
892 models. *arXiv preprint arXiv:2407.21783*, 2024.
- 893 Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang.
894 rstar-math: Small llms can master math reasoning with self-evolved deep thinking. *arXiv preprint*
895 *arXiv:2501.04519*, 2025.
- 896 Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu,
897 Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms
898 via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- 899 Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu,
900 Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiad-
901 bench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal
902 scientific problems, 2024.
- 903 Jujie He, Jiakai Liu, Chris Yuhao Liu, Rui Yan, Chaojie Wang, Peng Cheng, Xiaoyu Zhang, Fuxiang
904 Zhang, Jiacheng Xu, Wei Shen, Siyuan Li, Liang Zeng, Tianwen Wei, Cheng Cheng, Bo An, Yang
905 Liu, and Yahui Zhou. Skywork open reasoner series. [https://capricious-hydrogen-41c.](https://capricious-hydrogen-41c.notion.site/Skywork-Open-Reasoner-Series-1d0bc9ae823a80459b46c149e4f51680)
906 [notion.site/Skywork-Open-Reasoner-Series-1d0bc9ae823a80459b46c149e4f51680](https://capricious-hydrogen-41c.notion.site/Skywork-Open-Reasoner-Series-1d0bc9ae823a80459b46c149e4f51680),
907 2025. Notion Blog.

908 Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song,
909 and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *Sort*, 2(4):
910 0–6, 2021.

911 Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum.
912 Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base
913 model. *arXiv preprint arXiv:2503.24290*, 2025.

914 Yiming Huang, Xiao Liu, Yeyun Gong, Zhibin Gou, Yelong Shen, Nan Duan, and Weizhu Chen.
915 Key-point-driven data synthesis with its enhancement on mathematical reasoning. *arXiv preprint*
916 *arXiv:2403.02333*, 2024.

917 Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec
918 Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint*
919 *arXiv:2412.16720*, 2024.

920 Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E.
921 Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model
922 serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating*
923 *Systems Principles*, 2023.

924 Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ra-
925 masesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative
926 reasoning problems with language models. *Advances in Neural Information Processing Systems*,
927 35:3843–3857, 2022.

928 Chen Li, Weiqi Wang, Jingcheng Hu, Yixuan Wei, Nanning Zheng, Han Hu, Zheng Zhang, and
929 Houwen Peng. Common 7b language models already possess strong math capabilities. *arXiv*
930 *preprint arXiv:2403.04706*, 2024a.

931 Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita
932 Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, et al. From generation to judgment:
933 Opportunities and challenges of llm-as-a-judge. *arXiv preprint arXiv:2411.16594*, 2024b.

934 Xuefeng Li, Haoyang Zou, and Pengfei Liu. Limr: Less is more for rl scaling. *arXiv preprint*
935 *arXiv:2502.11886*, 2025a.

936 Zhong-Zhi Li, Duzhen Zhang, Ming-Liang Zhang, Jiaxin Zhang, Zengyan Liu, Yuxuan Yao, Haotian
937 Xu, Junhao Zheng, Pei-Jie Wang, Xiuyi Chen, et al. From system 1 to system 2: A survey of
938 reasoning large language models. *arXiv preprint arXiv:2502.17419*, 2025b.

939 Xiao Liang, Xinyu Hu, Simiao Zuo, Yeyun Gong, Qiang Lou, Yi Liu, Shao-Lun Huang, and Jian
940 Jiao. Task oriented in-domain data augmentation. *arXiv preprint arXiv:2406.16694*, 2024.

941 Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan
942 Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let’s verify step by step. In *The Twelfth*
943 *International Conference on Learning Representations*, 2023.

944 Haoxiong Liu, Yifan Zhang, Yifan Luo, and Andrew C Yao. Augmenting math word problems via
945 iterative question composing. In *Proceedings of the AAAI Conference on Artificial Intelligence*,
946 volume 39, pages 24605–24613, 2025a.

947 Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min
948 Lin. Understanding rl-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*,
949 2025b.

950 Dakuan Lu, Xiaoyu Tan, Rui Xu, Tianchu Yao, Chao Qu, Wei Chu, Yinghui Xu, and Yuan Qi. Scp-
951 116k: A high-quality problem-solution dataset and a generalized pipeline for automated extraction
952 in the higher education science domain, 2025. URL <https://arxiv.org/abs/2501.15587>.

953 Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng,
954 Qingwei Lin, Shifeng Chen, and Dongmei Zhang. Wizardmath: Empowering mathematical
955 reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*,
956 2023.

957 Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai,
958 Jeffrey Luo, Li Erran Li, Raluca Ada Popa, and Ion Stoica. Deepscaler: Surpassing o1-preview
959 with a 1.5b model by scaling rl. DeepScaleR Notion Page, 2025. Notion Blog.

960 Trung Quoc Luong, Xinbo Zhang, Zhanming Jie, Peng Sun, Xiaoran Jin, and Hang Li. Reft:
961 Reasoning with reinforced fine-tuning. *arXiv preprint arXiv:2401.08967*, 3, 2024.

962 MAA. American mathematics competitions (AMC 10/12). Mathematics Competition Series, 2023a.
963 URL <https://maa.org/math-competitions/amc>.

964 MAA. American invitational mathematics examination (AIME). Mathematics Competition Series,
965 2024b. URL <https://maa.org/math-competitions/aime>.

966 Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke
967 Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time
968 scaling. *arXiv preprint arXiv:2501.19393*, 2025.

969 Hieu Nguyen, Zihao He, Shoumik Atul Gandre, Ujjwal Pasupulety, Sharanya Kumari Shivakumar,
970 and Kristina Lerman. Smoothing out hallucinations: Mitigating llm hallucination with smoothed
971 knowledge distillation. *arXiv preprint arXiv:2502.11306*, 2025.

972 Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong
973 Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow
974 instructions with human feedback. *Advances in neural information processing systems*, 35:27730–
975 27744, 2022.

976 Qizhi Pei, Lijun Wu, Zhuoshi Pan, Yu Li, Honglin Lin, Chenlin Ming, Xin Gao, Conghui He, and
977 Rui Yan. Mathfusion: Enhancing mathematic problem-solving of llm through instruction fusion.
978 *arXiv preprint arXiv:2503.16212*, 2025.

979 John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy
980 optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

981 Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang,
982 Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical
983 reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

984 Wei Shen, Guanlin Liu, Zheng Wu, Ruofei Zhu, Qingping Yang, Chao Xin, Yu Yue, and Lin Yan.
985 Exploring data scaling trends and effects in reinforcement learning from human feedback. *arXiv*
986 *preprint arXiv:2503.22230*, 2025.

987 Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng,
988 Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint*
989 *arXiv:2409.19256*, 2024.

990 Taiwei Shi, Yiyang Wu, Linxin Song, Tianyi Zhou, and Jieyu Zhao. Efficient reinforcement finetuning
991 via adaptive curriculum learning. *arXiv preprint arXiv:2504.05520*, 2025.

992 Zhengyang Tang, Xingxing Zhang, Benyou Wang, and Furu Wei. Mathscale: Scaling instruction
993 tuning for mathematical reasoning. In *International Conference on Machine Learning*, pages
994 47885–47900. PMLR, 2024.

995 Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun
996 Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with
997 llms. *arXiv preprint arXiv:2501.12599*, 2025.

998 Qwen Team. Qwq-32b: Embracing the power of reinforcement learning, March 2025. URL
999 <https://qwenlm.github.io/blog/qwq-32b/>.

1000 Yuxuan Tong, Xiwen Zhang, Rui Wang, Ruidong Wu, and Junxian He. Dart-math: Difficulty-aware
1001 rejection tuning for mathematical problem-solving. *Advances in Neural Information Processing*
1002 *Systems*, 37:7821–7846, 2024.

1003 Shubham Toshniwal, Ivan Moshkov, Sean Narenthiran, Daria Gitman, Fei Jia, and Igor Gitman.
1004 Openmathinstruct-1: A 1.8 million math instruction tuning dataset. *Advances in Neural Information*
1005 *Processing Systems*, 37:34737–34774, 2024.

1006 Peiyi Wang, Lei Li, Zhihong Shao, RX Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui.
1007 Math-shepherd: Verify and reinforce llms step-by-step without human annotations. *arXiv preprint*
1008 *arXiv:2312.08935*, 2023.

1009 Yu Wang, Nan Yang, Liang Wang, and Furu Wei. Examining false positives under inference scaling
1010 for mathematical reasoning. *arXiv preprint arXiv:2502.06217*, 2025.

1011 Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Lifu
1012 Tang, Xiaowei Lv, et al. Light-rl: Curriculum sft, dpo and rl for long cot from scratch and beyond.
1013 *arXiv preprint arXiv:2503.10460*, 2025.

1014 Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A Smith,
1015 Mari Ostendorf, and Hannaneh Hajishirzi. Fine-grained human feedback gives better rewards for
1016 language model training. *Advances in Neural Information Processing Systems*, 36:59008–59033,
1017 2023.

1018 Wei Xiong, Jiarui Yao, Yuhui Xu, Bo Pang, Lei Wang, Doyen Sahoo, Junnan Li, Nan Jiang, Tong
1019 Zhang, Caiming Xiong, et al. A minimalist approach to llm reasoning: from rejection sampling to
1020 reinforce. *arXiv preprint arXiv:2504.11343*, 2025.

1021 An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li,
1022 Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint*
1023 *arXiv:2412.15115*, 2024a.

1024 An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jian-
1025 hong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2. 5-math technical report: Toward mathematical
1026 expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024b.

1027 Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. Limo: Less is more for
1028 reasoning. *arXiv preprint arXiv:2502.03387*, 2025.

1029 Edward Yeo, Yuxuan Tong, Morry Niu, Graham Neubig, and Xiang Yue. Demystifying long
1030 chain-of-thought reasoning in llms. *arXiv preprint arXiv:2502.03373*, 2025.

1031 Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo
1032 Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for
1033 large language models. *arXiv preprint arXiv:2309.12284*, 2023.

1034 Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong
1035 Liu, Lingjun Liu, Xin Liu, et al. Dapo: An open-source llm reinforcement learning system at scale.
1036 *arXiv preprint arXiv:2503.14476*, 2025a.

1037 Yiyao Yu, Yuxiang Zhang, Dongdong Zhang, Xiao Liang, Hengyuan Zhang, Xingxing Zhang, Ziyi
1038 Yang, Mahmoud Khademi, Hany Awadalla, Junjie Wang, et al. Chain-of-reasoning: Towards
1039 unified mathematical reasoning in large language models via a multi-paradigm perspective. *arXiv*
1040 *preprint arXiv:2501.11110*, 2025b.

1041 Yufeng Yuan, Qiyang Yu, Xiaochen Zuo, Ruofei Zhu, Wenyan Xu, Jiaze Chen, Chengyi Wang,
1042 TianTian Fan, Zhengyin Du, Xiangpeng Wei, et al. Vapo: Efficient and reliable reinforcement
1043 learning for advanced reasoning tasks. *arXiv preprint arXiv:2504.05118*, 2025.

1044 Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does
1045 reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *arXiv*
1046 *preprint arXiv:2504.13837*, 2025.

1047 Weihao Zeng, Yuzhen Huang, Qian Liu, Wei Liu, Keqing He, Zejun Ma, and Junxian He. Simplerl-
1048 zoo: Investigating and taming zero reinforcement learning for open base models in the wild. *arXiv*
1049 *preprint arXiv:2503.18892*, 2025.

- 1050 Dan Zhang, Sining Zhoubian, Ziniu Hu, Yisong Yue, Yuxiao Dong, and Jie Tang. Rest-mcts*: Llm
1051 self-training via process reward guided tree search. *Advances in Neural Information Processing*
1052 *Systems*, 37:64735–64772, 2024.
- 1053 Shimao Zhang, Xiao Liu, Xin Zhang, Junxiao Liu, Zheheng Luo, Shujian Huang, and Yeyun Gong.
1054 Process-based self-rewarding language models. *arXiv preprint arXiv:2503.03746*, 2025.
- 1055 Xiaotian Zhang, Chunyang Li, Yi Zong, Zhengyu Ying, Liang He, and Xipeng Qiu. Evaluating the
1056 performance of large language models on gaokao benchmark. *arXiv preprint arXiv:2305.12474*,
1057 2023.
- 1058 Han Zhao, Haotian Wang, Yiping Peng, Sitong Zhao, Xiaoyu Tian, Shuaiting Chen, Yunjie Ji, and
1059 Xiangang Li. 1.4 million open-source distilled reasoning dataset to empower large language model
1060 training. *arXiv preprint arXiv:2503.19633*, 2025a.
- 1061 Xueliang Zhao, Wei Wu, Jian Guan, and Lingpeng Kong. Promptcot: Synthesizing olympiad-level
1062 problems for mathematical reasoning in large language models. *arXiv preprint arXiv:2503.02324*,
1063 2025b.
- 1064 Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul
1065 Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv*
1066 *preprint arXiv:1909.08593*, 2019.
- 1067 Yuxin Zuo, Kaiyan Zhang, Shang Qu, Li Sheng, Xuekai Zhu, Biqing Qi, Youbang Sun, Ganqu
1068 Cui, Ning Ding, and Bowen Zhou. Ttrl: Test-time reinforcement learning, 2025. URL <https://arxiv.org/abs/2504.16084>.
1069