

Rationale-Aware Answer Verification by Pairwise Self-Evaluation

Akira Kawabata
The Asahi Shimbun Company
kawabata-a@asahi.com

Saku Sugawara
National Institute of Informatics
saku@nii.ac.jp

Abstract

Answer verification identifies correct solutions among candidates generated by large language models (LLMs). Current approaches typically train verifier models by labeling solutions as correct or incorrect based solely on whether the final answer matches the gold answer. However, this approach neglects any flawed rationale in the solution yielding the correct answer, undermining the verifier’s ability to distinguish between sound and flawed rationales. We empirically show that in StrategyQA, only 19% of LLM-generated solutions with correct answers have valid rationales. Furthermore, we demonstrate that training a verifier on valid rationales significantly improves its ability to distinguish valid and flawed rationales. To make a better verifier without extra human supervision, we introduce REPS (Rationale Enhancement through Pairwise Selection), a method for selecting valid rationales from candidates by iteratively applying pairwise self-evaluation using the same LLM that generates the solutions. Verifiers trained on solutions selected by REPS outperform those trained using conventional training methods on three reasoning benchmarks (ARC-Challenge, DROP, and StrategyQA). Our results suggest that training reliable verifiers requires ensuring the validity of rationales in addition to the correctness of the final answers, which would be critical for models assisting humans in solving complex reasoning tasks.

1 Introduction

Reasoning in large language models (LLMs) (Brown et al., 2020; Chowdhery et al., 2022; OpenAI et al., 2024) plays a vital role in their intelligent behavior (Wei et al., 2022a; Kojima et al., 2022). Recent advancements, owing to progressive scaling (Kaplan et al., 2020) and high-quality corpora (Penedo et al., 2024), facilitate LLMs in solving complex reasoning tasks including mathematical

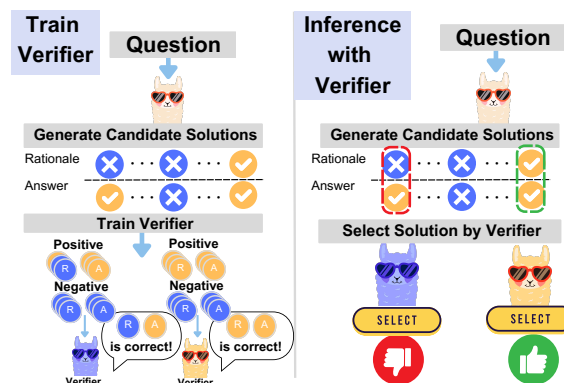


Figure 1: Importance of considering rationale quality in answer verification. Verifiers trained on correct answers with flawed reasoning (blue) fail to identify valid rationales at inference. In contrast, verifiers trained on solutions with correct answers and rationales (yellow) can distinguish valid reasoning.

reasoning and multi-hop reasoning. Further improvements are possible by prompting LLMs to generate intermediate reasoning steps that validate why the generated answer is correct (Wei et al., 2022b). However, LLM-generated reasoning often exhibits factual and logical consistency error (Maynez et al., 2020; Laban et al., 2023; Lyu et al., 2023; Jacovi et al., 2024; Dai et al., 2024).

Erroneous reasoning can be mitigated using a trained verifier model (Cobbe et al., 2021) to select the best solution from multiple generated solutions, each consisting of an answer and its intermediate reasoning (rationale). This methodology delivers more accurate answers than a single-answer setting (Cobbe et al., 2021; Li et al., 2023; Hosseini et al., 2024). However, existing studies mainly define the training labels for the verifier by solely checking the final answer. This approach may lead to a verifier that prefers faulty reasoning when LLMs generate the correct answers for wrong reasons (Figure 1). This is particularly problematic when models assist humans in complex tasks (Bowman et al., 2022; Saunders et al., 2022), wherein humans

can not fully examine the model’s feedback.

This raises two questions: (1) How often do models generate correct answers with flawed rationales? (2) Does training on flawed rationale as positive samples yield an untrustworthy verifier that cannot discriminate between flawed and valid rationale?

We investigate these questions using StrategyQA (Geva et al., 2021) as the testbed (detailed in Section 3). We use GPT-4 (OpenAI et al., 2024) to assess the validity of the rationales based on their factual and logical consistency with the annotated factual information. For the first question, we find that although 59% of the model-generated solutions contain the correct answer, only 19% of these correct-answer solutions are judged as having valid rationale, which is also supported by our manual inspection. For the second question, we create training datasets with varying levels of reasoning quality for the verifier. The verifier’s accuracy in selecting valid reasoning significantly increases as the quality improves.

To enhance the verifier’s ability to discern the validity of rationales, we introduce REPS (Rationale Enhancement through Pairwise Selection), a training method that leverages LLMs’ pairwise self-evaluation to select high-quality rationales from candidates. By refining the training data with these valid rationales, REPS enables to train more reliable and rationale-aware verifiers (Section 4).

Experiments across three reasoning datasets, ARC-Challenge (Clark et al., 2018), DROP (Dua et al., 2019), and StrategyQA demonstrate that verifiers trained on REPS consistently prefer solutions with valid rationales compared to those trained on datasets labeled solely based on the final answer’s correctness. While REPS serves as an effective quality filter for rationales without extra human supervision, our analysis implies that excessive pairwise evaluation may amplify inherent preference biases in LLMs, such as favoring longer text (Sections 5 and 6). We hope future work will explore methods to mitigate these biases and further enhance the reliability of verifiers, in preparation for models becoming capable of solving tasks that humans cannot examine. This work takes the first step towards building trustworthy verifiers that can discern the validity of rationales.

Our contributions can be summarized as follows:

- We empirically demonstrate the crucial role of rationale quality in training reliable verifiers. Our analysis reveals that a large portion of

correct answer solutions have invalid reasoning, and verifiers trained on such rationales tend to select solutions with similarly faulty reasoning.

- We introduce REPS, a method for selecting high-quality rationales by applying iterative pairwise self-evaluation. Verifiers trained with REPS significantly outperform baseline verifiers in selecting solutions with valid rationale on ARC-Challenge, DROP, and StrategyQA.
- Our analysis reveals that while iterative pairwise evaluation enhances the rationale quality, it can amplify inherent preference bias in LLM-based evaluators toward selecting longer rationales.¹

2 Related Work

2.1 Answer Verification

Recent studies have explored using verifier models for scoring or ranking generated solutions in domains such as mathematical reasoning (Cobbe et al., 2021; Wang et al., 2023b; Yu et al., 2023; Lightman et al., 2024; Miao et al., 2024), coding (Ling et al., 2023; Ni et al., 2023), and common-sense reasoning (Li et al., 2023; Weng et al., 2023; Zhang et al., 2024). Verifiers are trained to distinguish between correct and incorrect solutions using several approaches, such as generative models (Korbak et al., 2023; Asai et al., 2024), reward models (Cobbe et al., 2021), or a combination (Rafailov et al., 2023; Hosseini et al., 2024). When labeling solutions as “correct” and “incorrect”, existing studies mainly rely on checking consistency with the final answer (Yu et al., 2023; Hosseini et al., 2024). A notable exception is Li et al. (2023), which attempts to extract reasonable reasoning steps from wrong answer solutions through comparisons of the steps with those in correct solutions.

However, reasoning paths generated by LLMs often contain logical or factual errors (Lyu et al., 2023; Turpin et al., 2023; Golovneva et al., 2023), despite having a correct final answer (Jacovi et al., 2024). This study challenges the assumption that a correct answer indicates a valid reasoning path to build rationale-aware verifiers.

¹The code and data are available at <https://github.com/AkiraKawabata/REPS>.

2.2 LLM-as-Judge

Existing studies have explored using LLMs to evaluate natural language generation tasks (Chen et al., 2023; Zheng et al., 2023; Pan et al., 2024; Kim et al., 2024; Zeng et al., 2024). These methods can be categorized into three types: providing natural language feedback (Madaan et al., 2023; Paul et al., 2024), scoring with scalar values (Liu et al., 2023b), and comparing multiple outputs (Xie et al., 2023; Qin et al., 2024; Liu et al., 2024b). LLM-based evaluation can also improve training by offering feedback as supervision (Xu et al., 2023; Liu et al., 2023a, 2024a), filtering the training dataset by score (Gulcehre et al., 2023) or using pairwise comparison (Bai et al., 2022). This study applies pairwise evaluation, which aligns better with human judgment than direct scoring (Wang et al., 2023a), to refine the verifier’s training data.

3 Does Flawed Rationale Lead to Untrustworthy Verification?

LLMs may generate responses that contain correct answers but invalid reasoning paths (Jacovi et al., 2024). Thus, labeling generated solutions as positive samples based on the correctness of the final answer, without considering the validity of the reasoning, may yield a verifier that fails to distinguish between sound and flawed reasoning. Herein, we investigate this hypothesis by decomposing it into two questions: (1) How often do LLMs generate correct answers with invalid reasoning? (Section 3.3) (2) How does the rationale quality in the positive training data affect the verifier’s performance in selecting valid reasoning? (Section 3.4)

3.1 Task Setting of Answer Verification

In answer verification, a verifier model evaluates the correctness of multiple candidate solutions generated by an answer-generation model and selects the highest-scoring candidate as the final answer.

Verification A verifier model M_v takes a solution s generated by an answer-generation model M_g for a question q , and returns the probability p that the solution is correct. The solution s consists of an answer a and a reasoning path r that represents the rationale justifying the answer a , i.e., $s = (a, r)$. The verifier’s output probability can be expressed as: $p = M_v(s | q)$.

Training Verifier Models To train a verifier, we sample solutions from the generator M_g for each

question q with temperature $T = 0.7$. These solutions are then classified as correct or incorrect based on arbitrary criteria (e.g., whether the answer matches the gold answer). The verifier model M_v is trained to judge the correctness of each solution s as a reward model, following Cobbe et al. (2021). We use a binary cross-entropy loss function for training:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^N [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)], \quad (1)$$

where N is the number of training samples, y_i is the binary label (1 for correct and 0 for incorrect) for the i -th solution, and p_i is the probability predicted by the verifier for the i -th solution being correct.

Answer Selection by Verifiers At inference time, given a set of candidates s_1, s_2, \dots, s_n generated by M_g for a question q , the verifier model M_v predicts the probability p_i for each candidate s_i and selects the highest:

$$s^* = \arg \max_{s_i} M_v(s_i | q) \quad (2)$$

where s^* is the selected answer.

3.2 Experimental Settings

Model We use Llama-2 7B, a publicly available LLM, as the base model for both the answer generator and verifier owing to its popularity (Xie et al., 2023; Hosseini et al., 2024), thus facilitating a comparison with existing research.

Dataset We use StrategyQA, a yes/no question dataset requiring multi-hop reasoning using factual knowledge. We select StrategyQA as our testbed because it requires broad factual knowledge and consistent reasoning, areas where LLMs often struggle (Jacovi et al., 2024). We split the official train (2,290 questions) subset into 1,603 training and 687 test questions, as the official test set does not contain gold answers.

Validity Evaluation We evaluate rationale quality based on factuality and logical consistency, following previous studies on model-generated reasoning evaluation (Ott et al., 2023; Golovneva et al., 2023; Radhakrishnan et al., 2023; Press et al., 2023; Jacovi et al., 2024). Factuality assesses the grounding of a model’s reasoning in real-world knowledge, while logical consistency checks the coherence of the reasoning process.

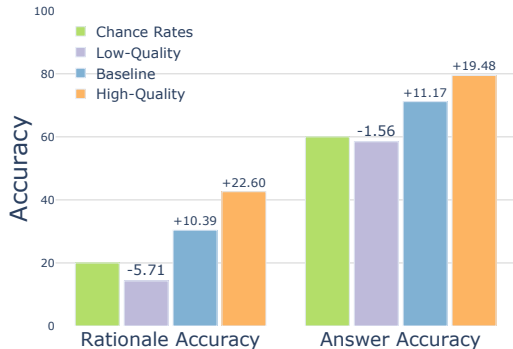


Figure 2: Rationale Accuracy (%) and Answer Accuracy (%) of verifier models trained on datasets with varying levels of rationale quality.

We use GPT-4 to annotate the validity of generated rationales. To ensure that GPT-4’s judgment is not hallucinated, we provide it with supporting facts annotated in each question in StrategyQA.² GPT-4’s validity annotations show a Cohen’s kappa of 0.62 with our manual annotations on 100 randomly sampled rationales.³

3.3 How Often Does LLM Generate Correct Answers with Invalid Reasoning?

We examine how often model-generated solutions have invalid reasoning despite having correct answers. For each question, we prompted the model with 6-shot examples to generate solutions s , consisting of a reasoning path r and an answer a .

GPT-4’s evaluation showed that while 59% of the generated solutions contained the correct answer, only 19% of those correct-answer solutions were judged as having valid reasoning paths. This highlights the limitations of relying solely on answer correctness to determine the validity of a solution, as it does not guarantee sound reasoning.

3.4 How Does Rationale Quality Affect Verifier Performance in Selecting Valid Reasoning?

We examine how rationale quality in positive samples affects verifier performance in selecting valid solutions. We conduct experiments with different levels of rationale quality.

Training Datasets We create three different training datasets for the verifier model. Following pre-

²Examples of GPT-4’s annotations are provided in Appendix D.

³More details for our annotations are in Appendix F.

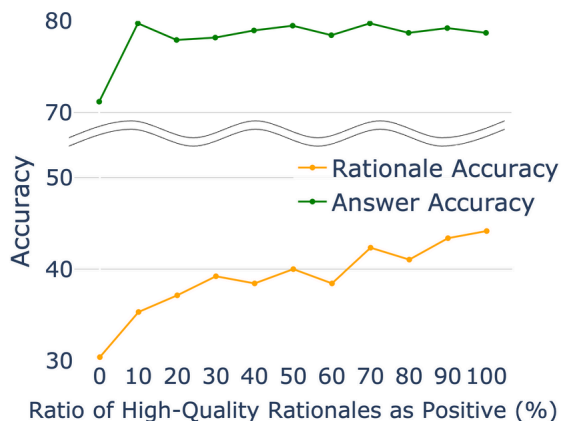


Figure 3: Rationale Accuracy (%) and Answer Accuracy (%) as a function of the ratio of high-quality rationales mixed into the baseline dataset.

vious studies (Cobbe et al., 2021; Li et al., 2023), we consider solutions with incorrect answers as negative samples.

- Low-quality setting:** Positive samples are created by replacing the final answer of incorrect solutions with the correct one, resulting in little to no valid reasoning.
- Baseline setting:** Positive samples are solutions with correct final answers, regardless of reasoning quality.
- High-quality setting:** Positive samples are solutions with correct answers and valid rationales, as validated by GPT-4.

The quality of rationales in the positive samples is expected to increase from the Low-quality setting to the High-quality setting. To collect valid solutions for the High-quality setting, we evaluate up to 20 generated solutions per question by GPT-4 in a zero-shot manner. When any one solution passes the evaluation, we include the solution in the high-quality training dataset as a positive example. If none of the 20 solutions pass, the question is discarded. Consequently, we obtain 1,318 training examples. We keep dataset sizes consistent across all settings for fair comparison.

Distribution-Controlled Test Set Design We design a test set to evaluate the ability of the verifier to select solutions independently of the generator model’s output distribution. For each question in the test set, we create a set of solution candidates consisting of:

- One valid solution $s_{valid} = (r_+, a_+)$ where r_+ denotes a valid rationale and a_+ is the corresponding correct answer.
- Two solutions $s_{correct} = (r_-, a_+)$ with correct answer a_+ but invalid rationale r_- .
- Two solutions $s_{incorrect} = (r_-, a_-)$ with incorrect answer a_- and invalid rationale r_- .

N_{valid} , $N_{correct}$, and $N_{incorrect}$ denote the number of s_{valid} , $s_{correct}$, and $s_{incorrect}$, respectively. We set $N_{valid} = 1$, $N_{correct} = 2$, and $N_{incorrect} = 2$. We report two metrics:

- **Rationale Accuracy (RA):** The percentage of questions when the verifier selects the valid solution from the set of candidates.
- **Answer Accuracy (AA):** The percentage of questions when the verifier selects a solution with a correct answer (regardless of whether the rationale is valid or not) from the set of candidates.

Formally, these metrics can be expressed as:

$$RA = \frac{1}{|D_{test}|} \sum_{i=1}^{|D_{test}|} \mathbb{I} \left[\arg \max_{s \in S_i} M_v(s) = s_{valid} \right]$$

$$AA = \frac{1}{|D_{test}|} \sum_{i=1}^{|D_{test}|} \mathbb{I} \left[\arg \max_{s \in S_i} M_v(s) \in s_{good} \right]$$

where $s_{good} = s_{valid} \cup s_{correct}$, D_{test} is the test dataset, S_i is the set of answer candidates for the i -th question, and $\mathbb{I}[\cdot]$ is the indicator function. The chance rates for RA and AA are 20% and 60%, respectively.

3.5 Results

Figure 2 shows RA and AA of verifier models trained on datasets with varying rationale quality. Training on the high-quality dataset, where the rationales are validated by GPT-4, results in a verifier that is significantly better at identifying valid reasoning compared to the baseline setting. Conversely, the low-quality dataset, where positive samples lack valid reasoning, leads to a verifier with RA near chance rates.

To further investigate the impact of rationale quality on the verifier’s performance, we incrementally replace 10% of the baseline dataset with high-quality rationales. Figure 3 shows a clear trend: as

the proportion of high-quality rationales increases, RA improves while AA remains largely stable.

These results highlight the importance of high-quality rationales in training verifier models, particularly in distinguishing between valid and flawed solutions. Next, we propose a method for automatically constructing a training dataset with high-quality rationales.

4 Rationale Enhancement through Pairwise Selection (REPS)

As shown in Section 3, a large portion of LLM-generated answers contain flawed reasoning, even if the final answer is correct. Moreover, increasing the ratio of valid rationales in the positive samples improves the verifier’s ability to identify sound reasoning. Thus, filtering out solutions with flawed rationales is crucial for building a reliable verifier.

We propose REPS, which uses LLMs’ pairwise comparison to iteratively select high-quality rationales from diverse candidates. The overall process of REPS is illustrated in Figure 4. For a question q , we first prompt the generator model M_g to produce a set of candidate solutions $S = s_1, s_2, \dots$, where each solution s_i consists of an answer a_i and a rationale r_i . From this set, we select N solutions with correct answers a_+ to form the candidate solution set for the tournament-style pairwise evaluation.

We use tournament-style pairwise evaluation to find the best rationale among the N candidates. In each round, the generator model M_g acts as the evaluator that compares the rationales (r_i, r_j) of two solutions $(s_i, s_j) \in S$ and selects the more factually grounded and logically consistent one as the winner through majority voting. The evaluator is provided with the question, answer, two candidate rationales, and (depending on the dataset) passage and answer options. Given hand-crafted few-shot exemplars, the evaluator model outputs a justification for which rationale is better and why, followed by the preferred rationale’s index (1 or 2).⁴ To mitigate the position bias reported in LLM-based evaluations (Wang et al., 2023a), we alternate the presentation order of rationale 1 and 2 across the S comparisons within each round.

We perform this pairwise comparison for all $\frac{N}{2}$ pairs of solutions. After the pairwise comparisons, the $\frac{N}{2}$ preferred rationales form a new set of candidate solutions S' , and the pairwise compari-

⁴The prompt format used for the pairwise evaluation are provided in Appendix C.

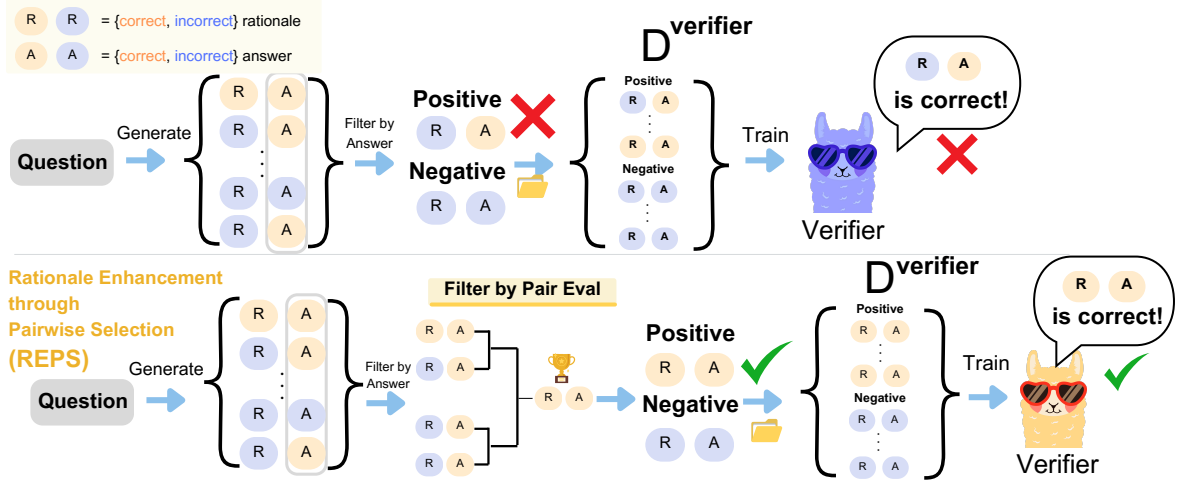


Figure 4: Rationale Enhancement through Pairwise Selection (REPS). The generator model produces candidate solutions and filters out those with incorrect answers. Unlike the conventional pipeline (top), REPS (bottom) employs a tournament-style pairwise evaluation to iteratively select the better solution. This refined solution is then used to train a rationale-aware verifier.

Algorithm 1 Rationale Enhancement through Pairwise Selection (REPS)

Input: Question q , Generator M_g , Num candidates N , Num voting S
 $C \leftarrow \{s_i \mid s_i \in M_g(q) \wedge s_i = (r_i, a_+)\}$
 $S \leftarrow \text{sample}(C, N)$
while $|S| > 1$ **do**
 $S' \leftarrow \emptyset$
 for (s_i, s_j) **do**
 $s^* \leftarrow \text{pairwise_eval}(s_i, s_j)$
 $S' \leftarrow S' \cup \{s^*\}$
 end for
 $S \leftarrow S'$
end while
 $s^* \leftarrow S[0]$
Output: s^*

son process is repeated on this reduced set. This tournament-style elimination continues until only one rationale remains, which we consider the refined rationale r^* . The refined rationale r^* , along with its corresponding answer a^* , is then added to the training set for the verifier model M_v as a positive sample. Repeating this for all questions refines the training dataset.

5 Experiment

5.1 Setup

Model We employ Llama-2 7B as the backbone for both the generator (M_g) and verifier (M_v) models. We report the performance of two verifiers: the

baseline verifier and the REPS verifier. The baseline verifier is trained using the conventional approach, labeling solutions as positive based solely on the correctness of the final answer. The REPS verifier is trained on a dataset where solutions refined through REPS are labeled as positive.⁵

Metrics We report two metrics: Rationale Accuracy and Task Performance. Rationale Accuracy is evaluated on a distribution-controlled test set (Section 3.3). The test set is constructed by sampling $N_{\text{valid}} = 1$, $N_{\text{correct}} = 2$, and $N_{\text{incorrect}} = 2$ solutions for each question. Valid solutions are selected by GPT-4, and questions for which none of the 20 evaluated solutions are judged valid are discarded as done in Section 3. Task Performance is evaluated by sampling five solutions for each question, ranking them with the verifier, and selecting the answer in the highest-scoring solution.

Datasets We evaluate our method on three diverse datasets that assess different aspects of reasoning capabilities:

1. StrategyQA: Dataset consisting of questions that require multi-hop reasoning using factual knowledge to arrive at the correct answer. We use 1,603 data points for training, 687 for testing Task Performance, and 385 of the test questions to evaluate Rationale Accuracy. We report accuracy as Task Performance.

⁵Details of the training and the inference are provided in Appendix A and B, respectively.

	ARC-Challenge		DROP		StrategyQA	
	Rationale Acc.	Task Perf.	Rationale Acc.	Task Perf.	Rationale Acc.	Task Perf.
Baseline	38.90	52.40	36.02	45.80	30.13	67.10
REPS	53.05	54.75	40.90	46.90	38.96	67.25

Table 1: Rationale Accuracy (%) and Task Performance (%) of the baseline verifier and REPS. REPS consistently outperforms the baseline in selecting valid rationales while maintaining or slightly improving Task Performance.

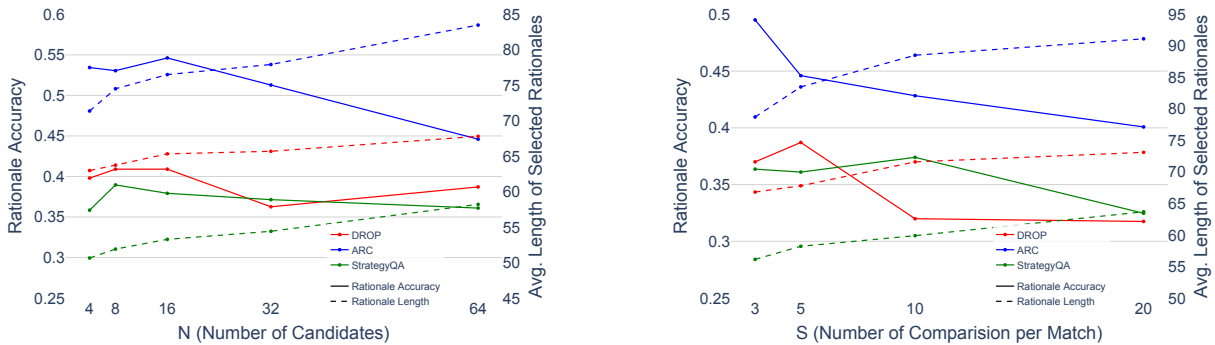


Figure 5: The effect of varying the number of candidate solutions (N) and the number of pairwise comparisons per match (S) on the Rationale Accuracy (%) and average length of selected rationales. Increasing N and S leads to a decrease in Rationale Accuracy and an increase in the average length of selected rationales.

- ARC-Challenge: Challenging subset of the AI2 Reasoning Challenge (ARC) dataset that evaluates commonsense reasoning about scientific knowledge. We use 1,119 questions for training, 1,172 for testing Task Performance, and 509 of the test questions to evaluate Rationale Accuracy. We report accuracy as Task Performance.
- DROP: Reading comprehension dataset that requires arithmetic reasoning to answer the questions. We select 2,000 questions for training, 1,000 for testing Task Performance, and 819 for evaluating Rationale Accuracy. We report Exact Match as Task Performance.

Additional References for GPT-4’s Annotation

We provided additional references for each dataset to support GPT-4’s annotation of valid solutions. For StrategyQA, as described in Section 3, we provide the supporting facts annotated in the dataset. For ARC, we provided the top five Wikipedia paragraphs with the highest BM25 scores against the concatenated question and answer options. We did not provide any additional references for DROP, as it is a reading comprehension task that does not require external factual knowledge.

REPS Parameters We set the number of pairwise comparisons per round (S) to 5 and the num-

ber of candidate solutions (N) to 8 for ensuring diversity and robustness in voting.

6 Results and Analysis

6.1 Effectiveness of REPS

Main Results Table 1 shows REPS improves Rationale Accuracy without affecting Task Performance across all datasets. The improvement is particularly significant in ARC and StrategyQA, where the REPS-trained verifier substantially outperforms the baseline, with improvements of 14.1% and 8.8%, respectively. This improvement can be attributed to the nature of the reasoning required in these datasets. In arithmetic reasoning tasks like DROP, which involve more deductive reasoning, it is rare for the model to arrive at the correct answer using flawed reasoning. Thus, the correctness of the final answer can be a good indicator of the validity of the reasoning process. In contrast, inductive and abductive reasoning tasks like ARC and StrategyQA allow more room for flawed reasoning to reach correct answers, emphasizing the importance of evaluating the intermediate reasoning.

Win Rate by GPT-4 We conduct a head-to-head evaluation using GPT-4 to compare the rationales selected by REPS and the baseline. For each question, we generate five candidate solutions, from

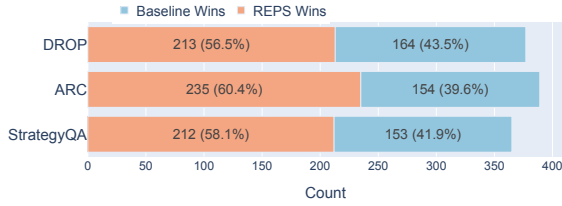


Figure 6: Win rate of REPS vs. baseline verifier when their selected rationales are compared head-to-head using GPT-4 as the judge.

which REPS and the baseline each select one. GPT-4 compares rationales for selected solutions, excluding those where both methods choose the same. In each pairwise evaluation, GPT-4 chooses the more factually grounded and logically consistent rationale, given the question and answer. We alternate the presentation order of the rationales selected by REPS and the baseline to mitigate position bias. Figure 6 shows REPS-selected rationales consistently outperform baseline-selected across datasets, with 56-60% win rates. Manual annotation of 100 random rationale pairs also confirms this trend, with REPS win rates of 63%, 60%, and 61% for ARC, DROP, and StrategyQA, respectively.⁶ REPS chooses better rationales than the baseline, even without guaranteed valid rationales among candidates.

6.2 Effects of Parameter Choice

We study how parameters in REPS, candidate pool size (N), and pairwise comparisons per round (S) affect Rationale Accuracy. A larger N is expected to increase the diversity of the candidate pool. Similarly, a higher S should enhance the reliability of the majority voting process. Thus, increasing both factors are expected to contribute to a higher Rationale Accuracy. Figure 5 illustrates the Rationale Accuracy as N varies from 4 to 64 and S from 3 to 20. Unexpectedly, Rationale Accuracy decreases as N and S grow. This phenomenon can be seen as the emergence of biases toward the superficial cue, i.e., the length of rationale, in model-based evaluation, as shown in Figure 5. As N and S increase, i.e., the number of pairwise evaluations per tournament grows, the length of the selected reasoning paths also tends to increase. This implies model-based evaluation’s bias for longer answers (Zheng et al., 2023; Koo et al., 2023; Dubois et al., 2024) amplifies over repeated pairwise evaluations,

⁶Appendix F shows details in our manual annotation.

	ARC	DROP	StrategyQA
G-EVAL	44.99	40.66	30.39
REPS	53.05	40.90	38.96

Table 2: Rationale Accuracy (%) of verifiers trained on rationales selected by REPS vs. G-EVAL.

causing deviation from valid rationale distribution.

One approach to handle the bias amplification is to treat the parameters N and S in REPS as hyperparameters that can be tuned using a validation set. Adjusting these parameters makes it possible to identify an optimal balance between effective rationale filtering and the minimization of bias amplification.

6.3 Importance of Pairwise Evaluation

We investigate whether the iterative pairwise evaluation employed by REPS produces more effective training samples for rationale-aware verifiers compared to single-answer evaluation. We use G-EVAL (Liu et al., 2023b) as a representative of the single-answer evaluation, which assigns a weighted score to each solution s based on the probabilities of outputting score tokens:

$$g(s) = \sum_{t_i=1}^5 p(t_i | s) \times t_i \quad (3)$$

where t_i represents the i -th score token (e.g., “1”), and $p(t_i | s)$ is the probability of outputting that token given the solution s . For each of the N candidate solutions, we compute the score S times and select the reasoning path with the highest average score. For fair comparison, we set candidate solutions N to 8 and evaluations per solution S to 5 for both methods. Table 2 shows that verifiers trained on REPS-selected rationales outperform those trained on G-EVAL-selected ones in identifying valid reasoning. This result demonstrates that REPS’s iterative pairwise comparison is more effective than direct scoring in providing higher-quality training samples for verifiers.

6.4 Error Types in Invalid Reasoning Paths

To gain insight into the types of errors present in reasoning paths judged as invalid by GPT-4, we conduct a manual analysis of 100 randomly sampled invalid reasoning paths for each dataset (ARC, StrategyQA, and DROP). The errors are categorized into four types: (1) both factual and logical

Error Type	ARC	StrategyQA	DROP
Factual only	16	15	35
Logical only	43	46	22
Fact. and Logic.	32	31	24
Other	2	5	9
No errors	7	3	10

Table 3: Distribution of error types (%) in reasoning paths judged as invalid by GPT-4.

errors, (2) factual errors only, (3) logical errors only, and (4) other errors that are neither factual nor logical (e.g., incomplete explanations or mere repetitions of the question). We define factual errors as reasoning that uses information contradicting the supporting facts or passage provided for the question. Logical errors are defined as reasoning that is irrelevant to a valid explanation or draws conclusions that cannot be derived from the preceding steps. Table 3 presents the distribution of these error types across the three datasets.

In DROP, factual errors are notably more prevalent (35% factual only and 24% both factual and logical). The model often struggles with extracting necessary information from the given passage, frequently using hallucinated or incorrect information in its reasoning process.

Conversely, ARC and StrategyQA exhibit a higher proportion of logical errors (43% and 46% logical only, respectively). The model often fails to identify the relevant information necessary to solve the problem, or misunderstands what can be inferred from commonsense knowledge.

7 Discussion

The Discrepancy Between Rationale Accuracy and Task Performance While REPS effectively improves Rationale Accuracy of verifiers, it does not significantly enhance overall Task Performance. This aligns with Section 3.3 and recent findings suggesting a weak link between rationale quality and answer correctness (Wiegrefe et al., 2022; Jacovi et al., 2024). This unfaithfulness becomes particularly critical in the context of scalable oversight (Bowman et al., 2022), where humans may find it challenging to evaluate the correctness of the model’s outputs. In such scenarios, humans might be misled by seemingly plausible yet unfaithful rationales. Future research should focus on improving rationales regarding quality and faithfulness.

8 Conclusion

We investigate how rationale quality affects a verifier’s ability to select valid answers. We empirically demonstrate that many model-generated rationales contain errors, even when the final answer is correct, leading to untrustworthy answer verification. We introduce REPS, a method that uses LLMs’ pairwise comparison to iteratively refine generated rationales. Experiments on three reasoning datasets show that REPS significantly outperforms baseline verifier models, particularly in selecting solutions with valid rationales. Our analysis shows iterative pairwise evaluation improves rationale quality but may amplify LLM-based evaluators’ biases.

Limitations

Dataset Diversity The experiments in this study are limited to three reasoning datasets: ARC-Challenge, DROP, and StrategyQA. While these datasets cover various aspects of reasoning, they may not be representative of all reasoning tasks. To further validate the effectiveness of REPS, it would be beneficial to evaluate the method on a more diverse set of datasets, such as those involving coding (Chen et al., 2021) or instruction following (Zheng et al., 2023).

Reliance on GPT-4 for Judging Although we confirmed a high agreement between GPT-4 and human judgments, GPT-4 may still be subject to biases not present in human evaluations, potentially leading to invalid evaluation.

Limited Training Data We use a relatively small amount of training data, ranging from 1,000 to 2,000 instances per dataset. It would be valuable to investigate REPS’s scalability on larger datasets.

Limited Model Size This study focuses on a single model, Llama-2 7B, as a case study for the generator and verifier models. It would be valuable to explore the impact of model size and architecture on REPS, as larger models may provide more accurate and coherent pairwise evaluations, leading to further improvements in the selected rationales.

Acknowledgments

We would like to thank the anonymous reviewers for their constructive feedback and Hideaki Tamori for his valuable comments. This work was supported by JSPS KAKENHI Grant Number JP22K17954.

References

- Asari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. [Self-RAG: Learning to retrieve, generate, and critique through self-reflection](#). In *The Twelfth International Conference on Learning Representations*.
- Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022. [Constitutional AI: Harmlessness from AI feedback](#). *Preprint*, arXiv:2212.08073.
- Samuel R. Bowman, Jeeyoon Hyun, Ethan Perez, Edwin Chen, Craig Pettit, Scott Heiner, Kamilė Lukošiuė, Amanda Askell, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Christopher Olah, Daniela Amodei, Dario Amodei, Dawn Drain, Dustin Li, Eli Tran-Johnson, Jackson Kernion, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Liane Lovitt, Nelson Elhage, Nicholas Schiefer, Nicholas Joseph, Noemí Mercado, Nova DasSarma, Robin Larson, Sam McCandlish, Sandipan Kundu, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Timothy Telleen-Lawton, Tom Brown, Tom Henighan, Tristan Hume, Yuntao Bai, Zac Hatfield-Dodds, Ben Mann, and Jared Kaplan. 2022. [Measuring progress on scalable oversight for large language models](#). *Preprint*, arXiv:2211.03540.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language models are few-shot learners](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebguss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. [Evaluating large language models trained on code](#). *Preprint*, arXiv:2107.03374.
- Yi Chen, Rui Wang, Haiyun Jiang, Shuming Shi, and Ruifeng Xu. 2023. [Exploring the use of large language models for reference-free text quality evaluation: An empirical study](#). In *Findings of the Association for Computational Linguistics: IJCNLP-AACL 2023 (Findings)*, pages 361–374, Nusa Dua, Bali. Association for Computational Linguistics.
- Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayanan Pilla, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. [Palm: Scaling language modeling with pathways](#). *Preprint*, arXiv:2204.02311.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. [Think you have solved question answering? Try ARC, the AI2 reasoning challenge](#). *Preprint*, arXiv:1803.05457.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. [Training verifiers to solve math word problems](#). *Preprint*, arXiv:2110.14168.
- Chengwei Dai, Kun Li, Wei Zhou, and Songlin Hu. 2024. [Beyond imitation: Learning key reasoning](#)

- steps from dual chain-of-thoughts in reasoning distillation. *Preprint*, arXiv:2405.19737.
- Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. **DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs**. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378, Minneapolis, Minnesota. Association for Computational Linguistics.
- Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2024. **Length-controlled alpacaeval: A simple way to debias automatic evaluators**. *Preprint*, arXiv:2404.04475.
- Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. **Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies**. *Transactions of the Association for Computational Linguistics (ACL)*.
- Olga Golovneva, Moya Peng Chen, Spencer Poff, Martín Corredor, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2023. **ROSCOE: A suite of metrics for scoring step-by-step reasoning**. In *The Eleventh International Conference on Learning Representations*.
- 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. 2023. **Reinforced self-training (rest) for language modeling**. *Preprint*, arXiv:2308.08998.
- Arian Hosseini, Xingdi Yuan, Nikolay Malkin, Aaron Courville, Alessandro Sordani, and Rishabh Agarwal. 2024. **V-STaR: Training verifiers for Self-Taught reasoners**. *Preprint*, arXiv:2402.06457.
- Alon Jacovi, Yonatan Bitton, Bernd Bohnet, Jonathan Herzig, Or Honovich, Michael Tseng, Michael Collins, Roei Aharoni, and Mor Geva. 2024. **A chain-of-thought is as strong as its weakest link: A benchmark for verifiers of reasoning chains**. *Preprint*, arXiv:2402.00559.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. **Scaling laws for neural language models**. *Preprint*, arXiv:2001.08361.
- Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoon Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2024. **Prometheus: Inducing fine-grained evaluation capability in language models**. In *The Twelfth International Conference on Learning Representations*.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. **Large language models are zero-shot reasoners**. In *Advances in Neural Information Processing Systems*.
- Ryan Koo, Minhwa Lee, Vipul Raheja, Jong Inn Park, Zae Myung Kim, and Dongyeop Kang. 2023. **Benchmarking cognitive biases in large language models as evaluators**. *Preprint*, arXiv:2309.17012.
- Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Vinayak Bhalerao, Christopher Buckley, Jason Phang, Samuel R. Bowman, and Ethan Perez. 2023. **Pretraining language models with human preferences**. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 17506–17533. PMLR.
- Philippe Laban, Wojciech Kryściński, Divyansh Agarwal, Alexander R. Fabbri, Caiming Xiong, Shafiq Joty, and Chien-Sheng Wu. 2023. **LLMs as factual reasoners: Insights from existing benchmarks and beyond**. *Preprint*, arXiv:2305.14540.
- Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. 2023. **Making language models better reasoners with step-aware verifier**. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5315–5333, Toronto, Canada. Association for Computational Linguistics.
- Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. **Let’s verify step by step**. In *The Twelfth International Conference on Learning Representations*.
- Zhan Ling, Yunhao Fang, Xuanlin Li, Zhiao Huang, Mingu Lee, Roland Memisevic, and Hao Su. 2023. **Deductive verification of chain-of-thought reasoning**. In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Jiacheng Liu, Ramakanth Pasunuru, Hannaneh Hajishirzi, Yejin Choi, and Asli Celikyilmaz. 2023a. **Crystal: Introspective reasoners reinforced with self-feedback**. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11557–11572, Singapore. Association for Computational Linguistics.
- Weize Liu, Guocong Li, Kai Zhang, Bang Du, Qiyuan Chen, Xuming Hu, Hongxia Xu, Jintai Chen, and Jian Wu. 2024a. **Mind’s mirror: Distilling self-evaluation capability and comprehensive thinking from large language models**. *Preprint*, arXiv:2311.09214.
- Yang Liu, Dan Iter, Yichong Xu, Shuhang Wang, Ruochen Xu, and Chenguang Zhu. 2023b. **G-eval: NLG evaluation using gpt-4 with better human alignment**. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

- Yinhong Liu, Han Zhou, Zhijiang Guo, Ehsan Shareghi, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2024b. [Aligning with human judgement: The role of pairwise preference in large language model evaluators](#). *Preprint*, arXiv:2403.16950.
- Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. [Faithful chain-of-thought reasoning](#). In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 305–329, Nusa Dua, Bali. Association for Computational Linguistics.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. [Self-refine: Iterative refinement with self-feedback](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. [On faithfulness and factuality in abstractive summarization](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.
- Ning Miao, Yee Whye Teh, and Tom Rainforth. 2024. [Selfcheck: Using LLMs to zero-shot check their own step-by-step reasoning](#). In *The Twelfth International Conference on Learning Representations*.
- Ansong Ni, Srini Iyer, Dragomir Radev, Ves Stoyanov, Wen-tau Yih, Sida I Wang, and Xi Victoria Lin. 2023. [Lever: Learning to verify language-to-code generation with execution](#). In *Proceedings of the 40th International Conference on Machine Learning*.
- OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altmenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O’Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong

- Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2024. [Gpt-4 technical report](#). *Preprint*, arXiv:2303.08774.
- Simon Ott, Konstantin Hebenstreit, Valentin Liévin, Christoffer Egeberg Hother, Milad Moradi, Maximilian Mayrhauser, Robert Praas, Ole Winther, and Matthias Samwald. 2023. [Thoughtsource: A central hub for large language model reasoning data](#). *arXiv preprint*.
- Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2024. [Automatically Correcting Large Language Models: Surveying the Landscape of Diverse Automated Correction Strategies](#). *Transactions of the Association for Computational Linguistics*, 12:484–506.
- Debjit Paul, Mete Ismayilzada, Maxime Peyrard, Beatriz Borges, Antoine Bosselut, Robert West, and Boi Faltings. 2024. [REFINER: Reasoning feedback on intermediate representations](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1100–1126, St. Julian’s, Malta. Association for Computational Linguistics.
- Guilherme Penedo, Hyněk Kydlíček, Leandro von Werra, and Thomas Wolf. 2024. [Fineweb](#).
- Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah Smith, and Mike Lewis. 2023. [Measuring and narrowing the compositionality gap in language models](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5687–5711, Singapore. Association for Computational Linguistics.
- Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2024. [Large language models are effective text rankers with pairwise ranking prompting](#). *Preprint*, arXiv:2306.17563.
- Ansh Radhakrishnan, Karina Nguyen, Anna Chen, Carol Chen, Carson Denison, Danny Hernandez, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiušė, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Sam McCandlish, Sheer El Showk, Tamera Lanham, Tim Maxwell, Venkatesa Chandrasekaran, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. 2023. [Question decomposition improves the faithfulness of model-generated reasoning](#). *Preprint*, arXiv:2307.11768.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. [Direct preference optimization: Your language model is secretly a reward model](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. 2022. [Self-critiquing models for assisting human evaluators](#). *Preprint*, arXiv:2206.05802.
- Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. 2023. [Language models don’t always say what they think: Unfaithful explanations in chain-of-thought prompting](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghui Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023a. [Large language models are not fair evaluators](#). *Preprint*, arXiv:2305.17926.
- Peiyi Wang, Lei Li, Zhihong Shao, R X Xu, Damai Dai, Yifei Li, Deli Chen, Y Wu, and Zhifang Sui. 2023b. [Math-Shepherd: Verify and reinforce LLMs step-by-step without human annotations](#).
- Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022a. [Emergent abilities of large language models](#). *Transactions on Machine Learning Research*. Survey Certification.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022b. [Chain of thought prompting elicits reasoning in large language models](#). In *Advances in Neural Information Processing Systems*.
- Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao. 2023. [Large language models are better reasoners with self-verification](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2550–2575, Singapore. Association for Computational Linguistics.
- Sarah Wiegrefe, Jack Hessel, Swabha Swayamdipta, Mark Riedl, and Yejin Choi. 2022. [Reframing human-AI collaboration for generating free-text explanations](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 632–658, Seattle, United States. Association for Computational Linguistics.
- Yuxi Xie, Kenji Kawaguchi, Yiran Zhao, Xu Zhao, Min-Yen Kan, Junxian He, and Qizhe Xie. 2023. [Self-evaluation guided beam search for reasoning](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Wenda Xu, Danqing Wang, Liangming Pan, Zhenqiao Song, Markus Freitag, William Wang, and Lei Li. 2023. [INSTRUCTSCORE: Towards explainable text generation evaluation with automatic feedback](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5967–5994, Singapore. Association for Computational Linguistics.

Fei Yu, Anningzhe Gao, and Benyou Wang. 2023. [Outcome-supervised verifiers for planning in mathematical reasoning](#). *Preprint*, arXiv:2311.09724.

Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. 2024. [Evaluating large language models at evaluating instruction following](#). In *The Twelfth International Conference on Learning Representations*.

Yunxiang Zhang, Muhammad Khalifa, Lajanugen Logeswaran, Jaekyeom Kim, Moontae Lee, Honglak Lee, and Lu Wang. 2024. [Small language models need strong verifiers to self-correct reasoning](#). *Preprint*, arXiv:2404.17140.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. [Judging LLM-as-a-judge with MT-bench and chatbot arena](#). In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

A Training Details

We train the verifier models using Llama-2 7B as the base model. The hyperparameters for training are as follows: batch size per device of 32, 10 training epochs, learning rate of 3×10^{-7} , AdamW optimizer, and a random seed of 42. The models are trained on three NVIDIA A100 GPUs using mixed precision (bfloat16) to reduce memory usage and training time.

B Inference Details

B.1 Answer Generation

For generating candidate answers, we use the Llama-2 7B model with a temperature of 0.7. The number of few-shot examples provided to the model varies depending on the dataset. For ARC, we use 4 examples; for DROP, we use 4 examples; and for StrategyQA, we use 6 examples. The generated answers are then fed into the verifier models for scoring and selection.

B.2 REPS Pairwise Evaluation

In REPS, we perform pairwise comparisons of the generated rationales using the Llama-2 7B model with a temperature of 0.7. The number of few-shot examples used for each dataset is determined by the maximum number of tokens that can fit within the model’s context size of 4096. For ARC, we use 8 examples; for DROP, we use 3 examples; and for StrategyQA, we use 7 examples. The pairwise comparisons are conducted iteratively, with

the winning rationales from each round advancing to the next round until a single rationale remains.

B.3 GPT-4 Evaluations

For evaluating the validity of the generated rationales and measuring the win rates between the REPS-selected rationales and the baseline, we employ the GPT-4 model with a temperature of 0.0. We use the gpt-4-0314 version of OpenAI API. The instructions used for evaluating the rationales are following:

You are a helpful assistant who evaluates explanations’ factual and logical consistency. Given a question and the corresponding reference information (evidence), you will assess whether the explanation meets the following criteria with justifications:

(1) Factual correctness: The explanation should be factually correct based on the given evidence. If the explanation contains claims not directly supported by or contradicting the evidence, it should be judged incorrect. The explanation should not include information or make assertions not mentioned in the evidence.

(2) Logical coherence: The explanation should follow a clear and logical chain of reasoning. Each step should be appropriately justified based on the evidence and preceding steps. Leaps in logic, assumptions, or opinions not grounded in the given information are unacceptable.

If the explanation meets all of the above criteria, output True. Otherwise, the output will be false if the explanation fails to meet any criteria. Your output must be only either True or False. Aim to prefer False if there are doubts about whether the criteria are fully satisfied.

C REPS Pairwise Evaluation Prompts

We use dataset-specific formats for pairwise evaluation prompts in REPS, with a common instruction across all datasets:

You are a helpful assistant who evaluates explanations’ factual accuracy and logical consistency. Given a question and an answer, decide which of the two

provided explanations is more factually grounded and logically valid. Your output must be 1 or 2, where 1 corresponds to the first explanation, and 2 corresponds to the second explanation.

Pairwise evaluation is performed by providing the model with prompts in the following format:

```
{{instruction}}
Question: {{question}}
Answer: {{gold answer}}
Explanation 1: {{explanation1}}
Explanation 2: {{explanation2}}
Justification: {{justification}}
Preferred Explanation: {{1 or 2}}
```

D Examples of GPT-4’s Validity Judge

We provide examples of GPT-4’s judgments on the validity of reasoning paths generated by the Llama-2 7B in Table 4. **Logical errors** in the reasoning paths are highlighted in red, while **factual errors** are highlighted in blue.

E Examples of Rationales Selected by REPS and Baseline

Examples of rationales selected by REPS and the baseline verifier for the same question are provided across three datasets: ARC (Table 6), DROP (Table 7), and StrategyQA (Table 5).

F Human Annotation

Validity Judgement To measure the alignment between human judgments and GPT-4’s judgments, the authors conducted a validity assessment task. 100 questions were randomly selected from the StrategyQA dataset. For each of these questions, the authors evaluated whether the generated rationale was factually accurate and logically consistent, given the supporting facts for each question. Each rationale was manually labeled as either valid or invalid based on these criteria. The validity annotations obtained from GPT-4 show a Cohen’s kappa of 0.62 with the authors’ manual validity annotations.

Pairwise Win Rate Evaluation In the pairwise win rate evaluation, the authors compared 100 pairs of rationales selected by REPS and the baseline verifier for each dataset (ARC, DROP, StrategyQA).

For ARC and StrategyQA, the evaluation of factuality was conducted with reference to supporting facts. We chose the rationale that was more factually accurate and logically consistent, with the rationales presented in a randomized order to mitigate bias.

ARC Example:**Question:**

Farmers in Wyoming were concerned because some of their chickens were being preyed upon by hawks that lived in areas around their ranches. The farmers grouped together and hunted the hawks until they were no longer in their area. Which would most likely happen next?

Choices:

A. The chicken population would go down. B. Populations of mice and rats would increase. C. Another bird of prey would replace the hawk. D. The chickens would have a lower rate of disease.

Valid Rationale:

Removing the hawks from the area could affect the populations of other animals. If hawks are not present in an area, they are not eating mice and rats. Without hawks, more mice and rats would survive. This could increase the populations of mice and rats in surrounding areas. So, the correct answer is, "B. Populations of mice and rats would increase."

Invalid Rationale:

When the hawks were no longer in the area, the chickens had to **compete with other predators for food**. This led to a **decrease in the chicken population**. Since the hawks were no longer around, the populations of mice and rats would increase. **Therefore, the chickens would have a lower rate of disease** because there were fewer of them to catch diseases. Therefore, the answer is B.

DROP Example:**Passage:**

Skopje, as the Republic of Macedonia as a whole, is characterised by a large ethnic diversity. The city is located in a region where Ethnic Albanians and Macedonians meet, and it welcomed Romani, Turks, Jews and Serbs throughout its history. Skopje was mainly a Muslim city until the 19th century, when large numbers of Christians started to settle there. According to the 2002 census, Macedonians were the largest ethnic group in Skopje, with 338,358 inhabitants, or 66.75% of the population. Then came Albanians with 103,891 inhabitants (20.49%), Roma people with 23,475 (4.63%), Serbs (14,298 inhabitants), Turks (8,595), Bosniaks (7,585) and Vlachs (2,557). 8,167 people did not belong to any of these groups.

Question:

Who was the second most populous group in Skopje?

Valid Rationale:

The passage states that Albanians were the second most populous group in Skopje, with 103,891 inhabitants. The passage then goes on to list other groups, and mentions that the largest group is Macedonians. This indicates that the second most populous group is Albanians.

Invalid Rationale:

The passage mentions that Macedonians were the largest ethnic group in Skopje, with 338,358 inhabitants, or 66.75% of the population. According to the passage, the second most populous group in Skopje was Albanians, with 103,891 inhabitants, or 20.49% of the population. Albanians were the second most populous group, and the largest ethnic group was Macedonians, and **therefore, the Macedonians were the second most populous group in Skopje**.

StrategyQA Example:**Question:**

Do members of the Supreme Court of the United States have longer terms than most senators?

Valid Rationale:

A member of the Supreme Court of the United States is appointed to serve for life. A senator's term is six years. Thus, a member of the Supreme Court of the United States has a longer term than most senators. So the answer is yes.

Invalid Rationale:

The terms of Supreme Court justices are **9 months**, which is **longer than the 6-year** term of a senator. Therefore, the answer is yes.

Table 4: Examples of GPT-4's validity judgments on reasoning paths generated by Llama-2 7B across three datasets: ARC (top), DROP (middle), and StrategyQA (bottom). The valid rationales demonstrate sound reasoning, while the invalid rationales contain **factual errors** and **logical inconsistencies**.

Question:

Did University of Pittsburgh founder have great deal in common with Judith Sheindlin?

Rationale selected by REPS:

Judith Sheindlin is 76 years old. The founder of the University of Pittsburgh died in 1811, which was 128 years before Judith Sheindlin was born. Thus, the founder of the University of Pittsburgh did not have a great deal in common with Judith Sheindlin.

Rationale selected by Baseline:

The founder of the University of Pittsburgh was William Pitt. Judith Sheindlin ("The Judge") has a first name that is similar to William. Thus, they have a great deal in common.

Question:

Is Disney associated with Los Angeles County?

Rationale selected by REPS:

Disney has theme parks in Los Angeles County, California. Thus, Disney is associated with Los Angeles County.

Rationale selected by Baseline:

Disney is a large company that has many locations. Disneyland is located in Anaheim, which is 24 miles south of Los Angeles. Thus, Disney does not have a location in Los Angeles County.

Question:

Could a bee hummingbird balance a scale with a single pea on it?

Rationale selected by REPS:

A bee hummingbird weighs about 0.4g. A single pea weighs about 0.2g. Thus, a bee hummingbird could balance a scale with a single pea on it.

Rationale selected by Baseline:

The mass of a bee is about 1.8 milligrams. The mass of a pea is about 1.5 grams. Thus, a bee could not balance a scale with a single pea on it.

Question:

Can photography be considered abstract art?

Rationale selected by REPS:

Photography is art, but it is not abstract art. Photography can be realistic or non-realistic. Abstract art is non-realistic. Thus, photography cannot be considered abstract art.

Rationale selected by Baseline:

Photography is a type of art. Abstract art is a type of art. Thus, photography can be considered abstract art.

Question:

Are aggressive bumblebees suicidal?

Rationale selected by REPS:

Bumblebees can become aggressive when they are protecting their colony. These bees can be very aggressive. Thus, aggressive bumblebees are not suicidal.

Rationale selected by Baseline:

Bumblebees are both aggressive and suicidal.

Table 5: Examples of rationales selected by REPS and the baseline verifier for questions from StrategyQA.

Question:

Devil facial tumor disease (DFTD) is a disease that is decimating the population of Tasmanian devils. The disease passes from one animal to another through bites and is caused by parasites. The parasites cause cancerous tumors that spread throughout an infected animal's body and kill it. What is the best description of DFTD?

Choices:

A. a non-infectious, cell-cycle disease B. an infectious, cell-cycle disease C. a non-infectious, chronic disease D. an infectious, chronic disease

Rationale selected by REPS:

DFTD is a deadly infectious disease that is spread by parasites and causes cancerous tumors that spread throughout an infected animal's body. Thus, DFTD is an infectious, cell-cycle disease.

Rationale selected by Baseline:

The disease is caused by parasites, which are foreign organisms that invade a host body and cause disease. The parasites cause cancerous tumors that spread throughout the body and kill the host. Therefore, DFTD is an infectious, cell-cycle disease.

Question:

A type of small mammal from the mountain regions of the western United States makes its home out of piles of rock. During summer months, the mammal places grasses and seeds in protected places in the rock piles. Which of the following is the most likely reason for this behavior?

Choices:

A. to prepare for migration before winter B. to provide warmth during the cold winter months C. to store food that will be eaten over the winter months D. to protect the grasses and seeds from decay before winter

Rationale selected by REPS:

This is the behavior of a prairie dog, which stores grass, seeds, and other food items in rock piles during summer months. Prairie dogs will consume the stored food during winter months. Thus, the prairie dog's behavior is to store food for use during the winter months.

Rationale selected by Baseline:

To protect and preserve its food, the mammal buries grasses and seeds in piles of rock. Grasses and seeds decay over time. Thus, the mammal buries its food to protect it from decay.

Question:

According to cell classification, prokaryotic cells are separated from eukaryotic cells. Which feature is often used to distinguish prokaryotic cells from eukaryotic cells?

Choices:

A. life processes B. size differences C. plasma membranes D. energy molecules

Rationale selected by REPS:

Prokaryotes are bacteria and archaea, whereas eukaryotes are animals, plants, fungi, and other multicellular organisms. Prokaryotes do not contain membrane-bound organelles (structures) such as the nucleus. Therefore, the presence or absence of membrane-bound organelles is an important distinguishing feature of prokaryotes and eukaryotes.

Rationale selected by Baseline:

Prokaryotic cells lack a nucleus, while eukaryotic cells contain a nucleus. Thus, prokaryotic cells are distinguished from eukaryotic cells by the presence of a nucleus.

Table 6: Examples of rationales selected by REPS and the baseline verifier for questions from ARC-Challenge.

Passage:

Taunton has four art galleries: Taunton Art Association , Hughes/Donahue Gallery, Art Euphoric, and the Trescott Street Gallery. The Taunton Art Association founded in 1973, but had it roots at the Girl's Club in the early 1960s. Hughes/Donahue Gallery founded in 2007, a local community gallery serving local Taunton artists, surrounding areas of Southeastern Massachusetts and including the cities of Providence, and Washington DC. Art Euphoric founded in 2008 has both visual and craft exhibits and sales. The Trescott Street Gallery founded in 2012, primarily a visual arts gallery, but also exhibits crafts.

Question:

Which gallery was founded first, Hughes/Donahue Gallery or Art Euphoric?

Rationale selected by REPS:

The passage mentions that Hughes/Donahue Gallery was founded in 2007, and Art Euphoric was founded in 2008. The gallery that was founded first is Hughes/Donahue Gallery.

Rationale selected by Baseline:

The passage mentions that the Taunton Art Association was founded in 1973. The passage also mentions that Art Euphoric was founded in 2008. Therefore, Art Euphoric was founded after the Taunton Art Association. The passage does not mention when the Hughes/Donahue Gallery and the Trescott Street Gallery were founded.

Passage:

In 1085, Guadalajara was retaken by the Christian forces of Alfonso VI . The chronicles say that the Christian army was led by Alvar Fanez de Minaya, one of the lieutenants of El Cid. From 1085 until the Battle of Las Navas de Tolosa in 1212, the city suffered wars against the Almoravid and the Almohad Empires. In spite of the wars, the Christian population could definitely settle down in the area thanks to the repopulation with people from the North who received their first fuero in 1133 from Alfonso VII. In 1219, the king Fernando III gave a new fuero to the city. During the reign of Alfonso X of Castile, the protection of the king allowed the city to develop its economy by protecting merchants and allowing markets.

Question:

How many years after the people of the North received their first fuero from Alfonso VII did king Fernando III give a new fuero to the city?

Rationale selected by REPS:

After the people of the North received their first fuero from Alfonso VII in 1133, Fernando III granted a new fuero to the city in 1219. Therefore, the city received its fuero 86 years after the first.

Rationale selected by Baseline:

The passage mentions that the first fuero was given in 1133 by Alfonso VII. By looking at the dates, it can be concluded that 287 years later, the people of the North received their first fuero. After this, 86 more years passed before Fernando III gave a new fuero to the city. Thus, it took 373 years from the time the people of the North received their first fuero to the time Fernando III gave a new fuero to the city.

Table 7: Examples of rationales selected by REPS and the baseline verifier for questions from DROP.