Oops, Wait: Discourse Tokens Matter in Reasoning Model

Anonymous ACL submission

Abstract

Recent studies suggest that even data-efficient training with (≃1K) reasoning trajectories can induce non-trivial reasoning capabilities in large language models through post-training. Such training corpora often contain iconic tokens such as "wait", "so", and "alternatively", which frequently appear in reasoning trajectories and may play a role in this process. This paper focuses on characterizing observable token-level patterns in post-training and a case study of how data-efficient supervised fine-tuning (SFT) differs from, and falls short of, large-scale post-training. To this end, we first identify tokens that correlate with correct answers along reasoning trajectories across models and training setups. We then focus on the distribution and (functional) roles of the "wait" token to primarily study the model trained in a data-efficient manner compared with the counterpart. Our study finds that discourse tokens are associated with correctness and a reasoning accuracy jump, even in data-efficient SFT. This suggests data-efficient SFT can partially reproduce discourse-token patterns to mimic meaningful reasoning behavior, but the patterns are less aligned with high-confidence answer transitions than those from large-scale post-training.

1 Introduction

Large language models (LLMs) have recently made remarkable progress across a wide range of reasoning-intensive tasks (Achiam et al., 2023; Guo et al., 2025; Team, 2024; Yang et al., 2024, 2025a; Jiang et al., 2024; Team et al., 2025). Modern reasoning models explicitly generate a *reasoning trajectory*, which contains a step-by-step thinking process (Wei et al., 2022; Kojima et al., 2022; Achiam et al., 2023; Chen et al., 2025). These trajectories improve human interpretability (Lindsey et al., 2025), help to refine reasoning ability during post-training (Guo et al., 2025; Guha et al., 2025; Muennighoff et al., 2025; Ye et al., 2025). Most importantly, training on reasoning trajectories leads to substantial gains on complex benchmarks.

Beyond reasoning performance, recent studies highlight the frequent emergence of discourse tokens[1] within reasoning trajectories (Yang et al., 2025b; Liu et al., 2025; Qian et al., 2025). Likewise, in human language, discourse markers play a role in structuring arguments and using language fluently (Sun, 2013; Castro, 2009; Huneety et al., 2023; Stab and Gurevych, 2017). This suggests that how LLMs employ particular tokens during reasoning may be closely tied to reasoning ability. A few studies (Muennighoff et al., 2025; Zhang et al., 2025; Wang et al., 2025; Zhao et al., 2025) have investigated the use of discourse tokens to improve reasoning capabilities. In contrast, how these tokens are distributed and how their patterns relate to reasoning success across models and training setups remains underexplored.

In human language learning, Sun (2013) suggests that acquiring discourse markers is an essential step towards developing persuasive and logical argumentation. By analogy, models post-trained on reasoning trajectories may acquire and internalize token-level signals, especially the discourse tokens, related to reasoning ability. However, the reasoning training setting differs substantially from human learning. Humans develop such capabilities through repeated exposure across vast and diverse contexts, and LLMs are thus generally assumed to require millions of post-training examples to reliably induce comparable reasoning. Nevertheless, recent studies (Ye et al., 2025; Muennighoff et al., 2025) have shown that post-training with only a few curated reasoning examples can enhance LLMs' reasoning ability. Particularly, Muennighoff et al. (2025) demonstrated even ≃1K examples may suffice to elicit non-trivial reasoning

[1] We define tokens such as "wait", "so", "therefore", and "alternatively" as *discourse tokens* marking pivotal steps in problem solving and anchor reasoning structure and transitions.

behavior in LLMs - an unexpectedly strong effect given the limited supervision. A hypothesis of this success points to the discourse tokens that pervade training reasoning trajectories and may act as structural cues. However, while Muennighoff et al. (2025) suggests that data-efficient supervised fine-tuning (SFT) can induce reasoning behaviors, it remains underexplored how such behaviors relate to those observed in large-scale post-trained models. In particular, it is unknown whether these models exhibit similar discourse-level dynamics during reasoning.

In this paper, we focus on empirical analyses of discourse tokens in reasoning models and on a case study comparing data-efficient SFT with large-scale post-training. We begin by identifying which tokens are most salient, how they are distributed, and how they correlate with reasoning success. We then examine what is distilled by small, well-curated reasoning traces, through data-efficient SFT. In particular, we compare the model trained on only ∼1K examples (*e.g.,* s1.1-32B (Muennighoff et al., 2025)) with the models trained on orders of magnitude more data. These include DeepSeek-R1-Distill-Qwen-32B (Guo et al., 2025) with ∼800K examples and QwQ-32B (Team, 2024), which was trained on substantially larger but undisclosed data. We go beyond their reasoning performance to study what distinguishes these regimes and characteristic reasoning behaviors.

Our study finds that (1) the discourse tokens are associated with correctness and a reasoning accuracy jump, (2) these correlations vary by training regime but are consistent across model scales, (3) data-efficient SFT could transfer signals using the discourse tokens like "wait" that connect to reasoning success under limited data; however, the discourse tokens' patterns are less aligned with high-confidence answer transitions, and therefore, its overall effectiveness remains questionable. Our analyses motivate exploratory experiments to validate our insights, including an ensemble method based on our token-level signals that improves `pass@1` from 10%p to 13.4%p across the models: {R1-32B, QwQ-32B, s1.1-32B}. Finally, we aim to focus on observable patterns and correlations between discourse tokens and reasoning capability.

2 Related Work

Anthropomorphic expressions and discourse markers in reasoning. Recent studies have investigated how such markers appear in LLM reasoning. Yang et al. (2025b) analyzed external signals interpreted as anthropomorphic expressions associated with "aha-moments", while Liu et al. (2025) discussed the emergence of reflective patterns such as "wait" or "hmm" as discourse markers in R1-style reasoning models. Qian et al. (2025) further identified that thinking tokens correspond to peaks of mutual information within reasoning trajectories. These works suggest a link between linguistic discourse markers and reasoning behavior, but focus on individual models. In contrast, we examine how discourse tokens relate to both the success of data-efficient SFT models and their side effects.

"Wait" in reasoning of LLMs. Muennighoff et al. (2025) introduced test-time scaling via controlled reasoning extension with "wait". Building on this idea, Zhao et al. (2025) linked "wait" to activation patterns and proposed an activation control method, while Zhang et al. (2025) modeled reasoning as a balance between slow and fast thinking mediated by the "wait" token. In parallel, Wang et al. (2025) reported that removing "wait"-like tokens can even improve efficiency in some settings. While these studies highlight the importance of "wait" in shaping reasoning behavior, they primarily use it as control rather than as an analytical lens. Our work shifts the focus on how "wait" functions within reasoning trajectories for data-efficient SFT.

3 Discourse Tokens in Reasoning

This section studies token-level reasoning trajectories. We quantify the contribution of candidate discourse tokens to reasoning success, identify the most effective tokens, and examine these effects across model scales and training strategies.

3.1 Identifying discourse tokens

We quantify the discourse token signals by computing token probabilities, defined as the average probability of tokens generated immediately after "\n\n", rather than relying on simple frequency counts. This allows us 1) to compare how tokens behave across correct and incorrect reasoning, across various models; 2) to examine their relation to model confidence. By examining these dimensions, we identify which tokens consistently act as markers of successful reasoning and which are associated with errors, providing a quantitative basis for understanding how training strategies and model scales shape reasoning ability.

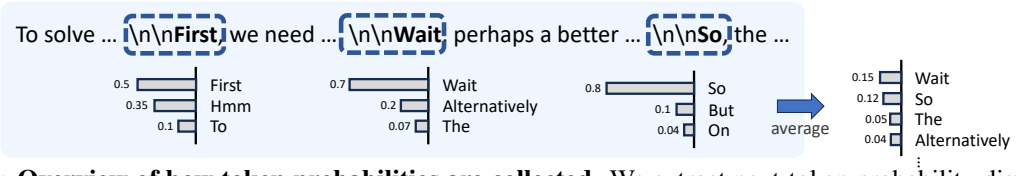


Figure 1: **Overview of how token probabilities are collected.** We extract next-token probability distributions specifically after "\n\n", which serve as natural segmentation points in the trajectories. We hypothesize that the first token plays a significant role in autoregressive generation.

Token	$\bar{p}_{true}(t)$	$\bar{p}_{false}(t)$	$\Delta(t)$
I	4.02%	1.30%	+2.72%
Therefore	4.14%	1.60%	+2.54%
The	5.02%	3.14%	+1.87%
Let	3.14%	1.40%	+1.74%
First	2.50%	0.82%	+1.68%
If	0.61%	0.96%	-0.34%
In	0.73%	1.66%	-0.94%
But	3.74%	7.23%	-3.49%
Alternatively	3.74%	8.98%	-5.23%
Wait	15.40%	25.75%	-10.34%

(a) R1-32B

Token	$\bar{p}_{true}(t)$	$\bar{p}_{false}(t)$	$\Delta(t)$
Therefore	7.58%	3.59%	+3.98%
So	5.57%	3.74%	+1.84%
First	3.00%	1.47%	+1.53%
I	1.74%	0.70%	+1.04%
Now	2.03%	1.19%	+0.84%
This	0.39%	0.76%	-0.37%
If	0.63%	1.17%	-0.54%
In	0.49%	1.40%	-0.91%
The	5.49%	8.54%	-3.05%
Alternatively	7.81%	17.75%	-9.94%

(b) QwQ-32B

Table 1: **Discourse tokens with statistically significant differences** (t-test, $p < 0.05$) between correct and incorrect trajectories for **DeepSeek-R1-distill-Qwen-32B (R1-32B)** and **QwQ-32B**. Tokens are sorted by $\Delta(t)$.

Mean token probability over the reasoning trajectory. Let $X = \{x_1, x_2, \ldots, x_T\}$ denote the token set in a trajectory, and let $I_X = \{i : x_i =$ "\n\n", $x_i \in X\}$ be a set of positions where "\n\n" occurs. The average token probability for trajectory X is $p_X = \frac{1}{|I_X|} \sum_{i \in I_X} p(x_{i+1})$, where $p(x_{i+1})$ denotes the probability assigned by the model to token x_{i+1}. We further compute aggregated statistics across multiple trajectories. Among all trajectories in a dataset $D = \{X_1, X_2, ...X_N\}$, we use two subsets: correct-answer $D_{true} = \{X_i : \text{Answer}(X_i) = true\}$ and incorrect-answer $D_{false} = \{X_i : \text{Answer}(X_i) = false\}$. Specifically, we define the mean token probability over correct- and incorrect-answer trajectories as

$$\bar{p}_* = \frac{1}{|D_*|} \sum_{X \in D_*} p_X, \quad (1)$$

where $* \in \{true, false\}$ for correct p_{true} or incorrect p_{false} mean probabilities, respectively. Figure 1 illustrates the overall process of how the token probability is computed by collecting the softmax probabilities: we use the token following "\n\n" as a high-precision discourse boundary to reduce contamination from problem-specific tokens. Note that we verify in Appendix B that alternatives yield similar trends, albeit with weaker signal strength.

We detail the benchmarks and models in Appendix A and B. For each tested model, t-test is performed on token probability distributions to assess whether they differ significantly between correct and incorrect reasoning. In addition, we prompt the models to generate confidence estimates, enabling us to examine the interactions among correctness, confidence, and token-level probabilities.

Identified discourse tokens and their relation to answer correctness. Tables 1 present representative examples of how token probabilities vary with reasoning correctness in R1-32B (*i.e.,* DeepSeek-R1-distill-Qwen-32B (Guo et al., 2025)) and QwQ-32B (Team, 2024), both of which were likely to be post-trained over Qwen2.5-32B (Yang et al., 2024). For both models, we report discourse tokens whose probabilities are significantly different when the answer is correct versus incorrect, as determined by a t-test ($p < 0.05$). As shown, some tokens exhibit large differences (*e.g.,* "wait" in R1-32B, "alternatively" in QwQ-32B), while others show relatively smaller gaps (*e.g.,* "so" in R1-32B, "let" in QwQ-32B). Moreover, the same token (*e.g.,* "the") can behave differently across two models, suggesting variability in token-level signals. We define discourse tokens with $\Delta(t)>0$ and $\Delta(t)<0$ as correct- and incorrect-associated tokens, respectively.

Importantly, tokens with $\Delta(t) < 0$, such as "wait" and "alternatively", commonly occur more frequently in incorrect trajectories, suggesting that they are employed to revise erroneous reasoning paths toward correct ones. This may reflect training data in which these discourse tokens appear

Model	Pearson	Spearman
R1-32B	0.9899**	1.0000***
QwQ-32B	0.8647*	0.8286*
s1.1-32B	0.9076*	1.0000***
s1-32B	0.7971	0.9000*

Table 2: **Correlation between correct–incorrect token probability gap and model confidence**. Both Pearson and Spearman correlations are reported (*: $p<0.05$, **: $p<0.01$, ***: $p<0.001$).

at key transition points that redirect trajectories toward correct outcomes. However, how many such transition-inducing examples are required during training remains underexplored.

Mean token probability correlates with confidence. To further justify the effectiveness of our metric, we examine how token-level probability correlates to model confidence. We compute the self-reported confidence per trajectory following the design proposed by Yoon et al. (2025) (see details in Appendix A). For each trajectory, the model confidence is compared with the trajectory-wise correct–incorrect token probability gap, computed using eq. (1) at the single-trajectory level as the difference between the summed probabilities of correct- and incorrect-associated tokens. This formulation allows for a more direct comparison between model confidence and token-level signals on each side.

As shown in Table 2, both Pearson and Spearman coefficients show strong positive correlations across all models, indicating that larger gaps are aligned with higher confidence. These findings indicate that token-level probability patterns not only reflect correctness but are also significantly associated with the self-reported confidence of the model, supporting the reliability of our token-level analyses as meaningful behavioral indicators. We argue that the correlation between higher token probability and greater confidence suggests that the model can assess, at the token level, whether a trajectory is progressing correctly. This, in turn, indicates that discourse tokens act as token-level signals.

3.2 Discourse tokens across setups and scales

Training recipes change discourse tokens. We compare correct- and incorrect-associated tokens across models under the shared architectural baseline (Qwen2.5). Table 3a presents the lists of correct-/incorrect-associated tokens across different models. First of all, tokens with negative or contrastive roles, such as "however", "but", "consider", and "alternatively", tend to appear as incorrect-associated tokens. In contrast, tokens that convey positive progression, such as "therefore", "so", and "let", are typically included among correct-associated tokens.

The associated tokens vary across the models: R1-32B, QwQ-32B, and s1.1-32B, which all share the same base architecture, Qwen2.5-32B, yet their signals diverge depending on the training setups. s1.1-32B, trained on DeepSeek-R1 trajectories, shows strong similarity to R1-32B. Similarly, QwQ-32B, though post-trained on the same backbone, shows further deviations due to its extensive combination of reinforcement learning and SFT.

Model scales do not change discourse tokens. We now control the fixed training strategy and compare three models of different sizes from the same series, R1-{7B, 14B, 32B}, as shown in Table 3b. Interestingly, the sets of correct- and incorrect- associated tokens remain consistent across scales, with only minor variations in their relative importance ($|\Delta(t)|$). This consistency suggests that token-level patterns are largely independent of model capacity and are instead governed by the shared training recipe, such as the training corpus. Summing up the results in Table 3, they further suggest that the post-training corpus can trigger discourse tokens that revise reasoning trajectories, largely independent of model size. Notably, even a small-scale corpus such as s1.1 seemingly exhibits this effect to some extent.

Summary of our findings. We investigate token probabilities in models, focusing on which tokens serve as indicators of reasoning success and failure. Our analysis shows that progression tokens (*e.g.,* "therefore", "so") are consistently associated with correctness, while contrastive tokens (*e.g.,* "wait", "alternatively") are associated with incorrectness (Table 1). Since token-level probability patterns involved in decision-making correlate strongly with model confidence (Table 2), this supports the reliability of the proposed metric and provides insight into the behavior of individual tokens with respect to trajectory success or failure. Additionally, the discourse tokens vary across training recipes but remain stable across model sizes (Table 3), suggesting that they are influenced more by training recipes than by capacity.

Interestingly, in both R1-32B and s1.1-32B, which are trained on DeepSeek-R1 trajectories with the same backbone, "wait" consistently emerges as an incorrect-associated token. As observed in the

Model	Type	Associated Tokens
R1-32B	Correct	I, Therefore, The, Let, First
	Incorrect	Wait, Alternatively, But, In, If
QwQ-32B	Correct	I, Therefore, So, First, Now
	Incorrect	This, If, In, The, Alternatively
s1.1-32B	Correct	Therefore, So, First, Now, Thus
	Incorrect	Alternatively, Wait, Given, If, In

(a) Models post-trained on Qwen2.5-32B

Model	Type	Associated Tokens
32B	Correct	I, Therefore, The, Let, First
	Incorrect	Wait, Alternatively, But, In, If
14B	Correct	Therefore, Let, Now, So, I
	Incorrect	Wait, Alternatively, But, In, If
7B	Correct	The, Now, Therefore, Let, I
	Incorrect	Wait, Alternatively, Hmm, But, In

(b) Varying scales of R1

Table 3: **Correct-/incorrect-associated tokens across models and scales.** Three models at the same scale: {R1-32B, QwQ-32B, s1.1-32B} and three scales of R1-{7B, 14B, 32B} under a consistent training method are employed.

probability of s1.1-32B in Table A, R1-32B enjoys a larger $\Delta(t)$, but is noticeably weaker in s1.1-32B. This observation motivates us to see 1) a closer examination of why such closely related models differ in their use (for revising incorrect trajectories) of "wait"; and 2) which signals are transferred through small-scale SFT concerning "wait", compared with larger datasets.

4 "Wait" in Reasoning Trajectories

The previous section highlighted that "wait" plays a key role in revising incorrect trajectories; we analyze it in depth and examine its correlation with reasoning success. We study how "wait" relates to reasoning progression and answer probability by truncating reasoning trajectories and prompting answers from incomplete reasoning. We conduct analyses of R1-32B and s1.1-32B to examine how data-efficient SFT can approximate the effects of large-scale post-training. Our goal is not to establish causation, but rather to provide empirical observations that help explain why just post-training with a few yet curated samples can nonetheless exhibit similar trends.

4.1 Probability jump and "wait" emergence

We compute the (next-token's) answer probability as the probability that the model would generate the correct final answer. We compute this probability at intermediate points in the thinking trajectory by truncating it at fixed intervals of 10 tokens and prompting the model to directly generate the final answer. We insert the model-specific answer delimiter token, "`<|im_start|>answer`" for s1.1 and "`</think>`" for R1, followed by the prefix "`Final answer: \boxed \{`". This setup ensures that the model produces the final answer explicitly. We use AIME24 (OpenAI, 2024) for efficiency. Since all the answers in AIME24 are numerical, the probability of a correct answer is computed from the predicted distribution over digits. If the correct answer consists of multiple digits, we condition the generation by fixing the preceding digits and compute the probability of each subsequent digit sequentially. Formally, for an answer represented as a sequence of digits $a = (d_1, d_2, \ldots, d_T)$, the probability is given by $P(a \mid \text{prompt}) = \prod_{t=1}^{T} P(d_t \mid d_{<t}, \text{prompt})$.

How does the answer probability go? The answer probability is expected to increase gradually as the reasoning trajectory unfolds. However, we observe that the probability often exhibits a sudden jump rather than a smooth trend (see Figure 2 illustrating an example trajectory with a highlighted jump area). We call this phenomenon a *probability jump* as the point in the trajectory where the increase in answer probability is maximized. We detect such a jump by sliding a window along each probability trajectory and computing, for each token position t, the difference between the average probability over the four preceding steps and the four following steps. The position t that maximizes this difference is designated as the *probability jump*. Notably, such sharp probability jumps are consistently observed whenever the model arrives at the correct answer. Concurrently, Tikhonov et al. (2026) identify a similar phenomenon and exploit it for early stopping, whereas we use the jump as an analytical anchor for discourse-token dynamics.

4.2 "wait" is multifaceted

Figure 2 further illustrates that many "wait" tokens usually appear near a probability jump. To systematically analyze this trend, we classify every occurrence of "wait" in the reasoning trajectory relative to the jump point. As discussed earlier, "wait" frequently appears in incorrect trajectories to revise them toward correct ones; moreover, it is also commonly observed in correct trajectories (Table 3). This trend suggests that "wait" plays a

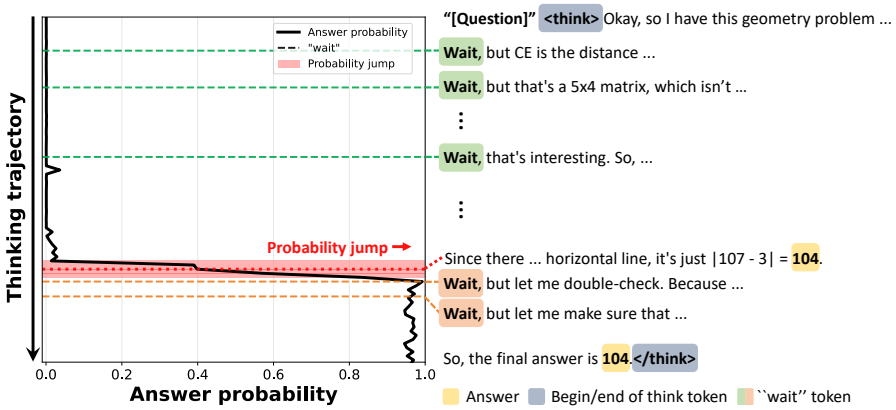


Figure 2: **Answer probability** changes and **emergence of "wait"** along the **reasoning trajectory**. Horizontal dashed lines indicate where "wait" is generated, and the **Probability Jump** with the red bar marks the point of maximum increase. Expressions following "wait" differ depending on whether they occur before or after the probability jump: earlier Wait (*rethink "wait"*) more often extend the reasoning by attempting to revise the reasoning trace, whereas later Wait (*recall "wait"*) re-check even after a decision has already been made.

characteristic role within each trajectory, particularly with respect to answer probability - *i.e.,* "wait" is multifaceted.

Rethink "wait"* and *recall "wait". We decompose the role of "wait" into two categories: *rethink "wait"* and *recall "wait"*. A *rethink "wait"* is defined as any "wait" token that appears before the probability jump, typically used to extend or reconsider the ongoing reasoning. Examples include phrases such as *"Wait, but in this case ..."* or *"Wait, but actually ..."*, where the token pushes the reasoning forward by exploring alternatives, and *"Wait, that's interesting. So ..."*, where the token extends the reasoning by building on the current line of thought. In contrast, a *recall "wait"* is defined as any occurrence of "wait" after the jump, generally produced when the model has already reached the solution and is double-checking or summarizing its result. For instance, it appears in forms like *"Wait, but let me double-check ..."* or *"Wait, but let me think again ..."*, signaling verification or restatement. In other words, we label each "wait" token as either preceding the confidence leap (rethink) or following it (recall). We hereafter use the terms *rethink* to denote the revision of incorrect trajectories, and *recall* to denote the re-evaluation of corrected tokens after the transition.

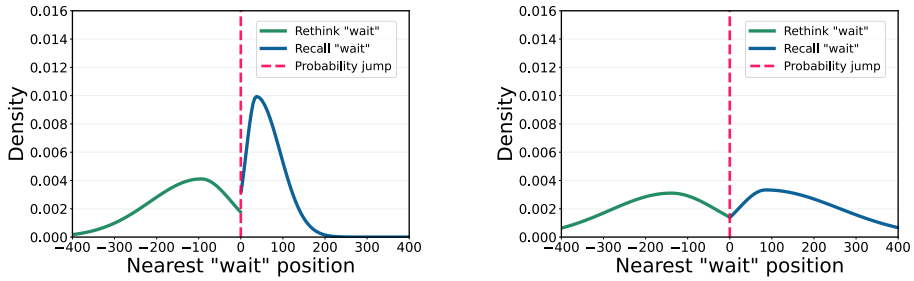
4.3 Can Data-Efficient SFT Rival Large-Scale Post-Training?

We analyze the patterns of *rethink "wait"* and *recall "wait"* tokens across two models: s1.1-32B and R1-32B to gain insights into data-efficient SFT relative to large-scale post-training.

Distribution of *rethink "wait"*s and *recall "wait"*s and their functionality. We compare *rethink "wait"* and *recall "wait"* around the probability jump for R1-32B and s1.1-32B. The distribution illustrated in Figure 3 shows that, in the case of *rethink "wait"s* (before the jump), they might trigger a reasoning step that makes the probability jump. After the jump, *recall "wait"*s follow to review its reasoning and confirm the answer. While the distributions of *rethink "wait"* are similar across the two models, the *recall "wait"*s show substantially different distributions between R1-32B and s1.1-32B. This implies that although the s1.1-32B learns the position of *rethink "wait"* from the small corpus enough to mimic R1-32B, yet the confirmation step via its revisiting process through *recall "wait"* still differs, which is related to the gap in reasoning patterns and performance.

Furthermore, for *rethink "wait"s*, we argue that budget forcing (Muennighoff et al., 2025) works by intentionally inserting more "wait" into the reasoning trace: when the number of wait tokens hits an internal threshold, a probability jump occurs. Although effective at test time, this may function as a temporary compensation by enriching many "wait" tokens.

R1-32B: fewer "wait" but reliable probability gains; s1.1-32B: many "wait" with low-probability exploration. Figure 4 shows the difference in answer probability increase distributions between R1-32B and s1.1-32B models. We report the maximum probability increases within a 384-token window after the "wait" token. For s1.1-32B, the answer probability increases are concentrated in the low-increase region (*i.e.,* $< 40\%$) compared to R1-32B. A few *rethink "wait"* make the proba-



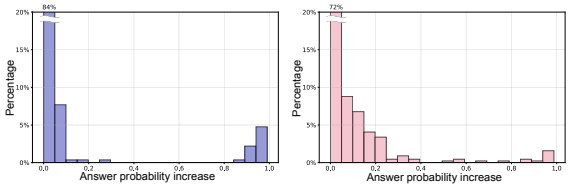
(a) R1-32B (b) s1.1-32B

Figure 3: Distribution of the **relative positions of the nearest rethink and *recall "wait"* token to the probability jump**. An asymmetric Gaussian curve is fitted to each distribution.

bility jump close to 100% (at the rightmost bin), but only in a small portion, 1.6%. Otherwise, R1-32B has a separate increase pattern: while it exhibits a reduced distribution in the low-increase region (*i.e.,* $< 40\%$), it demonstrates a stronger increase at higher probabilities ($>80\%$). Figure 5c reports the ratio of probability increase exceeds 80% after applying question-level normalization. R1-32B *rethink "wait"* is about four times more likely to make a probability jump compared to s1.1-32B. Overall, these analyses indicate that "wait" tokens in R1-32B have superior effects compared with those in s1.1-32B.

More specifically, we quantify "wait" in R1-32B and s1.1-32B across all questions in Figure 5. As shown in Figure 5a, s1.1-32B uses approximately 1.5× more "wait" tokens than R1-32B in rethink cases, and these tokens are more widely distributed prior to the probability jump (see Figure 3). This suggests that s1.1 performs more trials to induce a probability jump, albeit with a low success ratio (Figure 5c) as also indicated in Figure 4. In contrast, the number of *recall "wait"* is similar between the two models (Figure 5b); however, their behavior near the probability jump differs as shown in Figure 3, suggesting that R1 performs rethinking more promptly.

Finally, we visualize the number of "wait" in incorrect samples (*i.e.,* the reasoning trajectories where the final answer is incorrect) in Figure 5d. Intriguingly, R1-32B generates more "wait" in the incorrect samples than s1.1, in contrast to its usage in correct samples. R1-32B appears to use more "wait" than s1.1-32B when it cannot find a path to the answer. This is likely due to larger training data, which provides more opportunities to learn where to place "wait" tokens to revise incorrect trajectories into correct ones with higher confidence. While s1.1 exhibits similar effects, larger training corpora are evidently more beneficial for 'effective' reasoning success and overall success ratio.



(a) R1-32B (b) s1.1-32B

Figure 4: **Distribution of answer probability increases** within a fixed window following *rethink "wait"*.

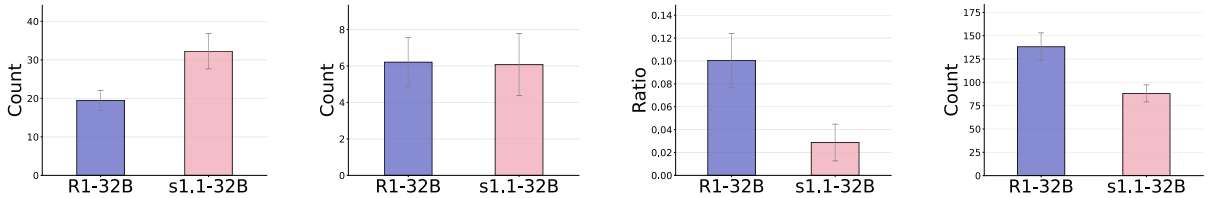
Finally, we have focused on for detailed analysis with "wait" due to its strong and consistent signal, similar (though weaker) trends would be observed for other discourse tokens such as "alternatively".

Summary of our findings. We investigate the difference in the distribution of "wait" between s1.1-32B and R1-32B. R1-32B enjoys a more appropriate occurrence of "wait" compared to s1.1-32B in the following aspects. "wait" emerges near the probability jump for R1-32B (Figure 3); efficient but high success ratio *rethink "wait"* tokens (Figure 5a and Figure 5c); high "wait" counts for incorrect samples Figure 5d. Although the strength of data-efficient SFT lies in its ability to mimic large-scale post-training even with limited data, as evidenced by similar wait distributions and overall behavioral patterns, the results suggest that it offers limited advantages beyond reasoning performance. We believe that models trained on small-scale data rely on discourse-token-enriched training data, leading to superficial reasoning patterns driven by the high frequency of such tokens, rather than learning when and where they should arise.

5 Exploratory Experiments

This section aims to reconfirm our claim and explore further applications.

Experiments on token suppression. Inspired by Section 3, we perform an exploratory experiment to determine whether manipulating discourse tokens actually affects model performance. We ex-



(a) Total *rethink "wait"* (b) Total *recall "wait"* (c) Success ratio (d) Incorrect samples

Figure 5: **Statistics of *rethink* and *recall "wait"* tokens** (more details beyond Figure 4). (a)–(d) show the number of *rethink "wait"* and *recall "wait"* tokens, the ratio of *rethink "wait"* tokens followed by a significant probability increase, and the total number of "wait" tokens in incorrect trajectories, respectively.

	Baseline	Correct	Incorrect	All
R1-32B	81.0%	80.2%	77.3%	78.5%
QwQ-32B	74.7%	73.2%	70.1%	68.0%
s1.1-32B	70.2%	74.9%	71.2%	70.9%

Table 4: **Effect of associated-token suppression.** Baseline denotes no suppression; Correct, Incorrect, and All denote suppression of correct-/incorrect-/all associated tokens, respectively. Note that s1.1-32B shows divergent trends, where suppressing correct-associated tokens improves over baseline (74.9% vs. 70.2%). It suggests that data-efficient SFT enforces discourse-token patterns without acquiring the underlying robustness.

	Baseline	Maj. V.	DeepConf	Ours
R1-32B	50.0%	**60.0%**	**60.0%**	**60.0%**
QwQ-32B	56.7%	63.3%	66.7%	**70.0%**
s1.1-32B	43.3%	53.3%	**56.7%**	**56.7%**

Table 5: **Ensemble using token-level signals.** Results are on the AIME24 benchmark. "Maj. V." denotes the majority voting strategy; ours denotes an ensemble strategy based on the correct–incorrect token probability gap proposed in this work, respectively.

amine this question by evaluating three models: R1-32B, QwQ-32B, and s1.1-32B across three settings: suppressing correct-associated tokens, incorrect-associated tokens (Table 1), and both groups simultaneously. We use AIME24, GPQA-Diamond, and MATH benchmarks and average the results.

We observe a consistent pattern in large-scale post-trained models: while both types of tokens influence performance, tokens associated with incorrect trajectories play a more crucial role in maintaining reasoning ability. However, data-efficiently trained model (s1.1-32B) does not follow this trend. We conjecture that limited data forces the model to mimic these behaviors without acquiring sufficient robustness, so that removing the enforced signal can sometimes be beneficial, revealing a side effect.

Introducing an ensembling method. Our analysis can also connect to select reliable samples across multiple trials to form an ensemble. Here, we instead utilize the correct–incorrect token probability gap, which exhibits a strong correlation with model confidence. We treat responses with low gap values as unreliable and exclude the bottom 20% of samples based on this metric during ensembling. All experiments are conducted on AIME24, where ensembling is performed over 32 sampled responses. We compare with: (1) majority voting, a general strategy for ensembling; (2) DeepConf (Fu et al., 2025), which recently introduced a group-level confidence for weighting or filtering answers from multiple trials.

Table 5 presents the experimental results. All ensembling methods outperform the baseline (pass@1, without ensembling). Notably, our approach using the correct-incorrect token probability gap achieves the best performance across all three models. This observation suggests that our analysis can provide insights for enhancing model performance beyond indicating simple correlation with confidence or internal model signals.

6 Conclusion

In this work, we have analyzed token-level signals to understand how large language models acquire and apply reasoning. Discourse tokens such as "therefore" and "so" are strongly linked to correct reasoning, whereas contrastive ones like "wait" and "alternatively" often accompany incorrect reasoning. These patterns remain stable across model sizes but vary with training recipes, suggesting that supervision plays a greater role than scale in shaping reasoning behavior. Further analysis of the "wait" token shows its dual role as both a trigger for probability shifts and a cue for self-checking. Our analysis also supports that data-efficient SFT captures token-level signals to approximate large-scale behavior, but only partially achieves the desired outcome. This, in turn, presumably explains why Muennighoff et al. (2025) benefits from inserting more discourse tokens like "wait" yet still exhibits performance gaps.

Limitations

While this study focuses on training strategies and model scales, our analyses primarily rely on open-source models from the Qwen series. Future work could extend this framework to other base architectures such as LLaMA (Touvron et al., 2023) or Mixtral (Jiang et al., 2024) to test its generality. Because our method requires access to softmax outputs and full reasoning trajectories, it is less applicable to closed-source reasoning models such as GPT (Achiam et al., 2023) or Gemini (Comanici et al., 2025). We also analyze three representative reasoning benchmarks that emphasize natural-language reasoning, where discourse markers naturally play a central role. Code-generation settings (Quan et al., 2025; Penedo et al., 2025; Jain et al., 2024) may exhibit different types of reasoning signals beyond discourse markers, which remain an interesting direction for future exploration. Finally, our analyses are correlational; targeted interventions on individual "wait" tokens would be needed to causally verify their functional roles.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Claudia Marcela Chapetón Castro. 2009. The use and functions of discourse markers in efl classroom interaction. *Profile: Issues in Teachers' Professional Development*, (11):57–77.

Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wanxiang Che. 2025. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *arXiv preprint arXiv:2503.09567*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Yichao Fu, Xuewei Wang, Yuandong Tian, and Jiawei Zhao. 2025. Deep think with confidence. *arXiv preprint arXiv:2508.15260*.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2024. The language model evaluation harness.

Etash Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, and 1 others. 2025. Openthoughts: Data recipes for reasoning models. *arXiv preprint arXiv:2506.04178*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Anas Huneety, Asim Alkhawaldeh, Bassil Mashaqba, Zainab Zaidan, and Abdallah Alshdaifat. 2023. The use of discourse markers in argumentative compositions by jordanian efl learners. *Humanities and Social Sciences Communications*, 10(1):1–8.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2024. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, and 1 others. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Jack Lindsey, Wes Gurnee, Emmanuel Ameisen, Brian Chen, Adam Pearce, Nicholas L. Turner, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, Jonathan Marcus, Michael Sklar, Adly Templeton, Trenton Bricken, Callum McDougall, Hoagy Cunningham, Thomas Henighan, Adam Jermyn, Andy Jones, and 8 others. 2025. On the biology of a large language model. *Transformer Circuits Thread*.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. 2025. Understanding r1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candes, and Tatsunori Hashimoto. 2025. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*.

OpenAI. 2024. Learning to reason with llms. OpenAI blog.

Guilherme Penedo, Anton Lozhkov, Hynek Kydlíček, Loubna Ben Allal, Edward Beeching, Agustín Piqueres Lajarín, Quentin Gallouédec, Nathan Habib, Lewis Tunstall, and Leandro von Werra. 2025. Codeforces. https://huggingface.co/datasets/open-r1/codeforces.

Chen Qian, Dongrui Liu, Haochen Wen, Zhen Bai, Yong Liu, and Jing Shao. 2025. Demystifying reasoning dynamics with mutual information: Thinking tokens are information peaks in llm reasoning. *arXiv preprint arXiv:2506.02867*.

Shanghaoran Quan, Jiaxi Yang, Bowen Yu, Bo Zheng, Dayiheng Liu, An Yang, Xuancheng Ren, Bofei Gao, Yibo Miao, Yunlong Feng, and 1 others. 2025. Codeelo: Benchmarking competition-level code generation of llms with human-comparable elo ratings. *arXiv preprint arXiv:2501.01257*.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2024. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*.

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3).

Wei Sun. 2013. The importance of discourse markers in english learning and teaching. *Theory & Practice in Language Studies (TPLS)*, 3(11).

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, and 1 others. 2025. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*.

Qwen Team. 2024. Qwq: Reflect deeply on the boundaries of the unknown.

Pavel Tikhonov, Ivan Oseledets, and Elena Tutubalina. 2026. Confidence leaps in llm reasoning: Early stopping and cross-model transfer. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 602–616.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Chenlong Wang, Yuanning Feng, Dongping Chen, Zhaoyang Chu, Ranjay Krishna, and Tianyi Zhou. 2025. Wait, we don't need to" wait"! removing thinking tokens improves reasoning efficiency. *arXiv preprint arXiv:2506.08343*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Shu Yang, Junchao Wu, Xin Chen, Yunze Xiao, Xinyi Yang, Derek F Wong, and Di Wang. 2025b. Understanding aha moments: from external observations to internal mechanisms. *arXiv preprint arXiv:2504.02956*.

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

Dongkeun Yoon, Seungone Kim, Sohee Yang, Sunkyoung Kim, Soyeon Kim, Yongil Kim, Eunbi Choi, Yireun Kim, and Minjoon Seo. 2025. Reasoning models better express their confidence. *arXiv preprint arXiv:2505.14489*.

Junyu Zhang, Runpei Dong, Han Wang, Xuying Ning, Haoran Geng, Peihao Li, Xialin He, Yutong Bai, Jitendra Malik, Saurabh Gupta, and 1 others. 2025. Alphaone: Reasoning models thinking slow and fast at test time. *arXiv preprint arXiv:2505.24863*.

Zekai Zhao, Qi Liu, Kun Zhou, Zihan Liu, Yifei Shao, Zhiting Hu, and Biwei Huang. 2025. Activation control for efficiently eliciting long chain-of-thought ability of language models. *arXiv preprint arXiv:2505.17697*.

Appendix

A Experimental Settings

Models. All models used in this study are publicly available open-source checkpoints released on HuggingFace under permissive licenses. We use six reasoning-oriented LLMs: DeepSeek-R1-distill-Qwen-32B[2], DeepSeek-R1-distill-Qwen-14B[3], DeepSeek-R1-distill-Qwen-7B[4], s1.1-32B[5], s1-32B[6], and QwQ-32B[7]. Licenses are MIT (DeepSeek-R1), Apache 2.0 (s1.1-32B and s1-32B), and Qwen Community License (QwQ-32B). Each model is used solely for research purposes.

Datasets and evaluation. We evaluate models on three reasoning-focused benchmarks: AIME24[8], GPQA-Diamond[9], and MATH-500[10]. Each dataset is used solely for evaluation purposes for scientific research. All evaluations are performed using the `lm-evaluation-harness` codebase (Gao et al., 2024)[11] and the `s1` codebase[12], both of which were adapted for our analysis.

All of these models and datasets are utilized for studying the reasoning of LLMs. The models (e.g., Qwen2.5, DeepSeek-R1, and QwQ series) were pretrained on multilingual corpora that include both English and Chinese data. However, all analyses and evaluations in this study were conducted exclusively on English-language benchmarks (AIME24, GPQA-Diamond, and MATH-500).

Chat template for self-reported confidence. To obtain self-reported confidence in Section 3, we use the chat template shown below, following the design proposed by Yoon et al. (2025).

Chat Template

First, solve the following math problem efficiently and clearly.
Then, thoroughly assess your confidence in that answer by evaluating your thinking process so far.
Finally, classify your confidence into one of the following classes based on how likely your answer is to be correct:
- "Almost no chance" (0.0–0.1)
- "Highly unlikely" (0.1–0.2)
- "Chances are slight" (0.2–0.3)
- "Unlikely" (0.3–0.4)
- "Less than even" (0.4–0.5)
- "Better than even" (0.5–0.6)
- "Likely" (0.6–0.7)
- "Very good chance" (0.7–0.8)
- "Highly likely" (0.8–0.9)
- "Almost certain" (0.9–1.0)

Each category reflects the probability that your answer is correct.
The last line of your response should be of the following format: Therefore, the final answer is: $\boxed {{ANSWER}}$, Confidence: $CLASS. I hope it is correct. (without quotes) where ANSWER is just the final number or expression that solves the problem and CLASS is one of the names (only the names without the probability ranges) of the classes above. Think step by step before answering.\n\n

B Details on computing token probabilities

In Section 3, we extract token-level signals through the average probability of tokens after "\n\n". In practice, we store the top-20 logits at each step and restrict our analysis to tokens whose average generation probability exceeds 0.02 and that appear on average more than 20 times per question, ensuring statistical reliability. We merge the probabilities of tokens that share the same semantic context but differ in surface form due to capitalization or leading whitespace (i.e., "Wait", "wait", " Wait", and " wait"). We conduct experiments on 30 questions from AIME24 (OpenAI, 2024), 100 questions from GPQA-D (Rein et al., 2024), and 100 questions from MATH-500 (Lightman et al., 2023).

[2] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-32B
[3] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-14B
[4] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-7B
[5] https://huggingface.co/simplescaling/s1.1-32B
[6] https://huggingface.co/simplescaling/s1-32B
[7] https://huggingface.co/Qwen/QwQ-32B
[8] https://huggingface.co/datasets/simplescaling/aime24_nofigures
[9] https://huggingface.co/datasets/Idavidrein/gpqa
[10] https://huggingface.co/datasets/simplescaling/openaimath
[11] https://github.com/EleutherAI/lm-evaluation-harness
[12] https://github.com/simplescaling/s1

We also consider calculating token probabilities at different token positions, not only after "\n\n". When token probabilities are computed after a dot or across all positions, the associated tokens for R1-32B in Table 1 remain consistent. However, the strength of the signals is weakened. For example, the average true-associated probability $\bar{p}_{\text{true}}(t)$ of "Wait" is 15.4% when considering positions after "\n\n", but it decreases to 2.15% when considering positions after a dot and to 0.03% when considering all positions. Furthermore, comparatively uninformative tokens, such as "$" or "=", appear to exhibit spurious signals due to problem-specific biases. Since our goal is to focus on discourse markers, we define token-level signals to be computed only at positions following "\n\n".

Token	$\bar{p}_{true}(t)$	$\bar{p}_{false}(t)$	$\Delta(t)$
Therefore	5.9%	2.8%	+3.1%
So	5.9%	3.2%	+2.7%
First	3.7%	1.7%	+2.0%
Now	1.9%	0.9%	+1.0%
Thus	1.4%	0.7%	+0.7%
In	0.6%	1.1%	-0.6%
If	0.9%	1.9%	-1.0%
Given	0.8%	2.2%	-1.3%
Wait	8.2%	10.5%	-2.3%
Alternatively	10.3%	18.4%	-8.1%

Table A: **Discourse tokens with statistically significant differences** (t-test, $p < 0.05$) between correct and incorrect trajectories for **S1.1-32B**. Tokens are sorted by $\Delta(t)$.

Model	**Per-trace**	**Group**
R1-32B	0.5802***	0.5993***
QwQ-32B	0.7200***	0.6585***
s1.1-32B	0.6896***	0.6092***
s1-32B	0.5312***	0.5019**

Table B: **Correlation between the correct-incorrect token probability gap and model confidence based on log probability**. "Per-trace" denotes the confidence averaged over a single trajectory, while "Group" refers to the log probability-based confidence following DeepConf (Fu et al., 2025). Pearson correlations are reported (*: $p < 0.05$, **: $p < 0.01$, ***: $p < 0.001$).

C Token-level signals and model confidence

In Section 3, we obtain model confidence using self-reported confidence. Model confidence can also be estimated by the average log probability over a single generated trace (per-trace confidence), as used in Farquhar et al. (2024), or by group confidence, as suggested in Fu et al. (2025). In this analysis, we follow DeepConf-low and compute group confidence using the bottom 10% local token groups within a trajectory. Table B presents the correlation results between our token-probability–based signals, derived from correct- and incorrect-associated tokens, and model confidence measured by log probability. Although the correlation is weaker than that observed with self-reported confidence, which exhibits clearer stratification due to its discrete confidence levels, the correlation remains statistically meaningful.

D Details on token suppression

For token suppression, we apply a masking strategy that prevents the generation of the correct- or incorrect-associated tokens listed in Table 3 during decoding, as a form of token-level steering. Each setting is run with a temperature of 0.6 for three trials, and we report the averaged results.

E The use of LLMs

We use large language models to construct a small-scale SFT dataset by paraphrasing an existing dataset. We also use them for minor language refinement, such as improving fluency and clarity. They were not involved in any aspect of the study's design, analysis, or interpretation, and all research findings are entirely our own.