ALIGNMENT BY CONFIDENT DEBATE: PREFERENCE LEARNING FROM MULTI-AGENT DEBATE

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-training language models for reasoning typically rely on gold answers, task-specific verifiers, or learned reward models, limiting scalability when such signals are scarce. Multi-agent debate offers a self-supervised alternative, but converting debate into reliable training supervision remains challenging: majority voting weights agreeing responses equally, lacks a principled tie-breaking rule, and is ill-defined for open-ended tasks without a canonical answer; likelihood-based scoring can conflate fluency with correctness; and homogeneous committees may reinforce correlated errors. We introduce **Alignment by Confident Debate** (AbCD), which converts debate traces into preference data without gold answer labels. A sampling-diverse committee debates each problem, after which final answers are clustered by task-level semantic equivalence. Each cluster is scored by aggregating confidence on its claimed answer-spans, and completions from the highest-scoring cluster are labeled as preferred over the remaining completions. We use verifiable tasks (mathematical reasoning, multiple-choice quantitative QA, and executable code-generation) as a controlled test bed: their answer keys or executable checkers let us evaluate the quality of debate-derived supervision against ground truth, even though the oracle itself never uses gold labels. Post-training on the resulting preferences outperforms inference-only ensembling and debate, as well as SFT on the same traces, across five model families spanning 1B to 8B parameters and four datasets, improving consensus accuracy, with a maximum gain of 21.5_{pp}. Ablations show that the approach transfers to a larger model and to code generation, while also revealing limitations of heterogeneous committees and agreement-based weighting. Overall, the results suggest that debate traces can yield useful preference supervision when answer equivalence is combined with confidence-weighted disagreement.

1 INTRODUCTION

Post-training for reasoning typically relies on external supervision to identify desirable solutions or solution traces. Such supervision can come from labeled demonstrations or gold answers, task-specific verifiers that evaluate final outcomes, or learned evaluators such as reward models and LLM-based judges. Gold answers and executable verifiers provide direct correctness signals when an answer key or reliable checking procedure exists, as in numerical reasoning and unit-tested code (Cobbe et al., 2021; Yuan et al., 2023; Austin et al., 2021). Reward models and LLM-based judges extend evaluation to less readily verifiable outputs (Ouyang et al., 2022; Zheng et al., 2023; Yuan et al., 2024; Wu et al., 2025). However, these evaluators must themselves be trained, calibrated, or otherwise trusted and may share systematic errors with the policy being optimized. Thus, although post-training objectives are well established, obtaining reliable correctness supervision remains difficult and domain-dependent.

When external correctness supervision is unavailable, agreement among the model's own samples provides a natural candidate signal. Multi-agent debate allows several agents to exchange reasoning over multiple rounds, and can improve answer accuracy relative to a single sampled solution (Du et al., 2023; Liang et al., 2024; Madaan et al., 2023). Subsequent debate-training methods often use these traces for training, for example by labeling the majority answer as preferred and the remaining answers as dispreferred (Samanta et al., 2025; Zhou et al., 2025). This strategy is fragile, however. In homogeneous committees, communication need not improve correctness beyond independent

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

sampling (Zhu et al., 2026); repeated interaction can also induce correlated reasoning and convergence on shared errors (Dai, 2026). On GSM8K, for example, a five-agent committee achieves its largest consensus accuracy after the first exchange, whereas additional rounds increase agreement without improving—and in some cases reducing—correctness (Figure 1). Consensus is therefore an outcome of the interaction and selection process, not by itself evidence that the selected answer is correct. We therefore study how to convert debate traces into preference supervision by separating answer equivalence from answer selection: final answers are clustered by task-level semantic equivalence and the resulting clusters are ranked by aggregate confidence in their claimed answer spans.

Majority voting requires a procedure for mapping final answers to equivalent outcomes. Without a canonical answer or reliable semantic-equivalence rule, voting cannot determine when two responses should count as equivalent, particularly for open-ended generation where correct responses may differ in wording (Chen et al., 2023). Even when equivalence is available, majority voting treats all agreeing traces equally and provides no additional signal for resolving ties. In small committees of cloned models, agreement may also reflect correlated errors rather than independent corroboration. Token probabilities provide another signal available during generation, but they measure the likelihood of generated tokens rather than correctness of the full reasoning chain. Length-normalized full-trace likelihood can therefore favor fluent or familiar reasoning without reliably identifying the correct answer. Confidence computed on answer-relevant tokens is a more targeted alternative and has been used to improve answer selection (Razghandi et al., 2025; Lin & Hooi, 2025). Nevertheless, these scores remain imperfect proxies because language models can be overconfident, and identifying the answer-relevant span may itself depend on the task. Exisiting confidence-based approaches largely use such scores for inference-time selection; we instead use them to construct preference data from debate traces.

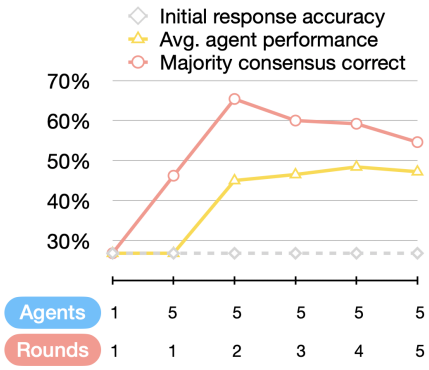


Figure 1: Committee behavior for Qwen-2.5-1.5B-it on GSM8K. The committee size varies from one to five agents under independent sampling, after which the number of debate rounds is varied with five agents. Additional rounds increase agreement but do not improve consensus accuracy beyond the first exchange and can reduce correctness.

That gap motivates **Alignment by Confident Debate** (AbCD), which constructs preference data from debate traces without using gold answer labels for supervision. A sampling-diverse committee generates and exchanges solutions over multiple debate rounds, after which final answers are clustered by task-level semantic equivalence. We score each cluster by aggregate confidence on its claimed answer-span rather than by the likelihood of the full reasoning trace. Completions from the highest-scoring cluster are labeled preferred, while completions from the remaining clusters are labeled dispreferred. Agreement rate is recorded as a reliability signal and can optionally be incorporated as a soft weight, but it does not define the winning cluster. On GSM8K, SVAMP, and MathQA, confidence-based cluster selection identifies the correct answer cluster more often than majority voting, particularly on instances with tied vote counts. Post-training on the resulting preferences improves both selected-answer and mean agent accuracy, outperforming inference-only ensembling, inference-only debate, and SFT trained on the same debate traces across the evaluated settings.

Contributions. We make the following contributions:

1. A confidence-based debate oracle that separates task-level answer equivalence from answer selection: equivalence determines which completions form a cluster, while confidence determines which cluster is preferred.

2. A preference-data construction procedure that converts the oracle's cluster assignments into training preferences without using gold labels for supervision.

3. A characterization of the design space of debate-based supervision, showing that a sampling-diverse committee captures most of the observed benefit in our experiments, whereas additional rounds can increase agreement without improving correctness. The analysis further evaluates transfer to a larger model, heterogeneous committees, and execution-based evaluation for code generation.

2 RELATED WORK

Multi-agent debate at inference. Self-consistency aggregates independently sampled solutions by majority vote when final answers can be mapped to equivalent outcomes (Wang et al., 2022), and universal self-consistency extends this idea to free-form outputs by using an LLM to select a consistent answer among candidates (Chen et al., 2023). Multi-agent debate instead allows agents to exchange reasoning over multiple rounds and can improve answer selection relative to a single sampled solution (Du et al., 2023; Liang et al., 2024; Madaan et al., 2023). However, these gains are not guaranteed: under homogeneous agents, debate preserves expected correctness over time and can underperform independent voting (Choi et al., 2026; Zhu et al., 2026), and repeated interaction can increase similarity among agents' rationales, amplifying shared errors (Dai, 2026). These findings motivate treating consensus as an outcome of the interaction and selection process rather than as a correctness label.

Confidence-based answer selection. Token likelihoods provide an alternative to vote counts, but full-response likelihood is a coarse signal because it scores all generated tokens, including reasoning tokens that may be only weakly tied to final-answer correctness. Confidence computed on answer-relevant tokens provides a more targeted correctness signal (Razghandi et al., 2025; Lin & Hooi, 2025), and related work uses confidence estimates (including verbalized confidence, token-distribution statistics, and calibrated answer-token scores) to select answers at inference time (Taubenfeld et al., 2025; Kang et al., 2026; Kadavath et al., 2022; Tian et al., 2023). Our work differs in using claimed answer-span confidence to *construct preferred and dispreferred traces* for post-training. An extended discussion of related work, including a finer-grained taxonomy of debate-based and confidence-based methods, is provided in Appendix A.

Self-training from unlabeled or weakly labeled traces. Prior methods construct training signals from verifier-approved traces (Zelikman et al., 2022; Gulcehre et al., 2023), pseudo-labels derived from sampled answers (Zuo et al., 2026; Shao et al., 2024), or judgments from another language model (Yuan et al., 2024; Bai et al., 2022). Other work uses debate traces for imitation, preference learning, or multi-agent reinforcement learning (Subramaniam et al., 2025; Srivastava et al., 2025; Samanta et al., 2025; Zhou et al., 2025; Motwani et al., 2024; Liu et al., 2026). These methods differ in how they identify preferred traces and assign training weights. To our knowledge, prior debate-training methods do not construct preference pairs by ranking task-level answer-equivalence clusters using aggregate claimed answer-span confidence. AbCD combines answer-equivalence with span-confidence mass, while treating agreement as a reliability statistic rather than as the definition of the winning answer.

3 METHODOLOGY

Let π_θ be the base instruction-tuned policy, which produces a reasoning trace for each query x. We instantiate N sampling-diverse agents from π_θ per query and run M rounds of multi-agent debate. The final traces are clustered according to task-level answer equivalence. Aggregate confidence on the claimed-answer span selects the winning cluster, whose traces are treated as preferred and the remaining traces as dispreferred. A learner initialized from π_θ is then trained using one of several objectives: supervised fine-tuning (SFT), preference optimization (DPO or KTO), or group-relative debate-derived preference reward optimization (GRPO). Figure 2 and Algorithm 1 summarize the pipeline. The post-training objectives are standard; the novelty of AbCD lies in its confidence-based oracle for selecting preferred traces. The following sections detail the committee, confidence score, equivalence-based oracle, and post-training objectives.

3.1 SAMPLING-DIVERSE COMMITTEE

The autoregressive policy (π_θ) generates a trace $y = (y_1, \ldots, y_{|y|})$ conditioned on an input prompt $x \in \mathcal{X}_{\text{train}}$:

$$\pi_\theta(y \mid x) = \prod_{t=1}^{|y|} \pi_\theta(y_t \mid x, y_{<t})$$

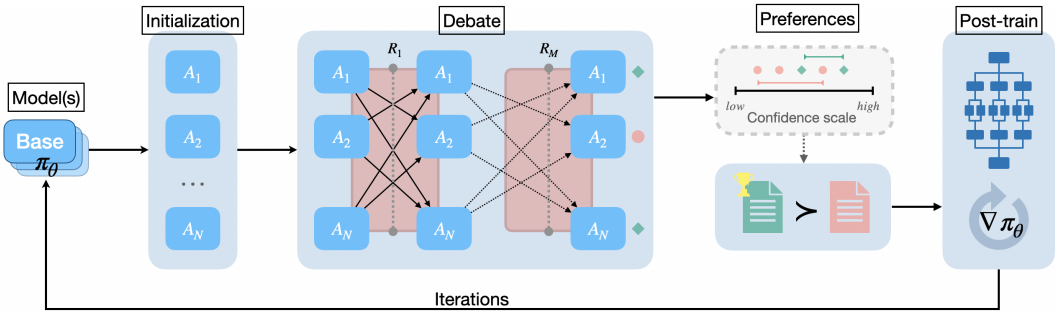


Figure 2: AbCD instantiates N sampling-diverse agents from the base policy π_θ and lets them debate for M generations. Final traces are clustered by task-level answer equivalence, and the cluster with the largest aggregate confidence on the claimed-answer span is selected. Traces in the winning cluster are preferred, while the remaining traces are dispreferred (green $\succ$ red). The resulting preference supervision updates π_θ without gold answers in the oracle.

A task parser $\mathcal{A}$ extracts the claimed answer $a = \mathcal{A}(y)$. For each prompt x, we instantiate N agents $\{\pi_i\}_{i=1}^N$ from the base policy π_θ. The agents may share parameters but use heterogeneous decoding configurations, such as different temperatures, role prompts, or sampling seeds. We refer to this as a sampling-diverse committee.

At round 1, agents generate traces independently conditional on x:

$$y_i^{(1)} \sim \pi_i(\cdot \mid x), \qquad i = 1, \ldots, N$$

Let ser_i serialize the previous-round traces shown to agent i. For exchange rounds $m = 2, \ldots, M$, agents synchronously revise their traces:

$$y_i^{(m)} \sim \pi_i\Big(\cdot \mid x,\, \mathrm{ser}_i\Big(\{y_j^{(m-1)}\}_{j \neq i}\Big)\Big)$$

Thus, the procedure consists of one independent generation followed by $M - 1$ exchange rounds. The final debate collection is

$$\mathcal{Y}(x) = (y_1, \ldots, y_N), \qquad y_i = y_i^{(M)}, \qquad a_i = \mathcal{A}(y_i)$$

Traces for which $\mathcal{A}(y_i)$ is undefined are excluded from equivalence clustering, assigned to the dispreferred set during preference construction, and counted as incorrect during evaluation.

3.2 Answer-confidence score

We define c_i^{answer} as the answer-confidence score for agent i's generated trace and consider two constructions: full-trace confidence c_i^{trace} and claimed answer-span confidence c_i^{span}. For each generated token, the corresponding log-probability is $\ell_{i,t} = \log \pi_i\Big(y_{i,t} \mid x, \mathrm{ser}_i(\{y_j^{(m-1)}\}_{j \neq i}), y_{i,<t}\Big)$. All confidence scores below reuse these recorded token log-probabilities.

Trace confidence. We define the confidence of the full generated reasoning trace as the geometric mean of its token probabilities:

$$c_i^{\text{trace}} = \exp\left(\frac{1}{|y_i|}\sum_{t=1}^{|y_i|} \ell_{i,t}\right) \tag{1}$$

Answer-span confidence. Let $S(y_i) \subseteq \{1, \ldots, |y_i|\}$ denote the token indices selected by a task-specific answer-span parser. For a nonempty answer-span, $S(y_i) \neq \varnothing$, we define

$$c_i^{\text{span}} = \exp\left(\frac{1}{|S(y_i)|}\sum_{t \in S(y_i)} \ell_{i,t}\right) \tag{2}$$

This score is the geometric mean of the token probabilities on the selected claimed answer-span. If $S(y_i) = \varnothing$ or cannot be aligned with the generated tokens, then c_i^{span} is undefined.

3.3 EQUIVALENCE CLUSTERS AND THE CONFIDENCE-WEIGHTED ORACLE

For each prompt x, parseable answers are partitioned into equivalence clusters using a task-specific relation $\equiv$ that compares candidate answers with one another. The relation is evaluated without access to the gold answer. Let $\mathcal{I}_k$ denote the indices of the agents whose answers belong to cluster k. We define the cluster size and aggregate answer confidence as

$$n_k = |\mathcal{I}_k|, \qquad w_k = \sum_{i \in \mathcal{I}_k} c_i^{\text{answer}}$$

We select the winning cluster by lexicographic maximization $k^\star = \arg\max_k \big(w_k, n_k\big)$, which prioritizes aggregate answer confidence and uses cluster size as a secondary tie-break. The oracle answer is the canonical representative of the selected cluster $\hat{a}(x) = \text{rep}(k^\star)$, where $\text{rep}(k^\star)$ returns a canonical string for cluster $k^\star$. If multiple clusters remain tied after both criteria, the prompt is excluded from oracle construction. For comparison, majority voting selects the cluster with the largest support, $k_{\text{maj}} = \arg\max_k n_k$.

We construct preferred and dispreferred trace sets relative to $\hat{a}(x)$:

$$\mathcal{G}^+(x) = \{y_i \in \mathcal{Y}(x) : a_i \equiv \hat{a}(x)\}, \qquad \mathcal{G}^-(x) = \{y_i \in \mathcal{Y}(x) : a_i \not\equiv \hat{a}(x)\} \tag{3}$$

Unparseable traces are excluded from clustering and assigned to $\mathcal{G}^-$ during preference construction. Preference objectives use only prompts for which both $\mathcal{G}^+(x)$ and $\mathcal{G}^-(x)$ are nonempty. Unanimous debates may still contribute their preferred traces to SFT and GRPO. We also measure the support for the selected cluster by

$$\alpha(x) = \frac{|\mathcal{G}^+(x)|}{N}$$

The statistic $\alpha(x)$ can optionally be used as a soft weight for the loss contribution of a preference example. It measures support for the confidence-selected winner and does not determine which cluster wins.

3.4 POST-TRAINING

For each prompt with a defined oracle, we retain the oracle answer and its winner support in

$$\mathcal{D}_{\text{oracle}} = \big\{\big(x, \hat{a}(x), \alpha(x)\big)\big\}_{x \in \mathcal{X}_{\text{train}}}$$

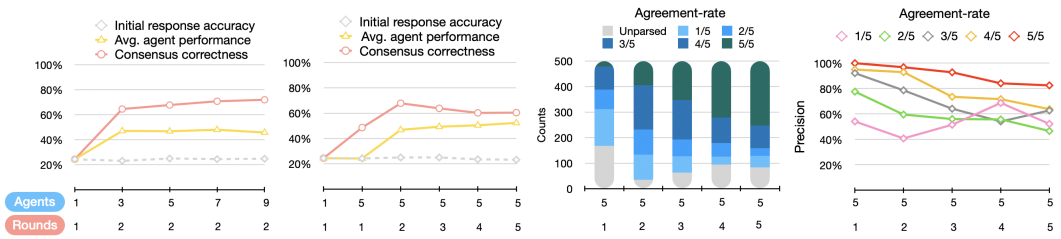
For preference-based objectives, we additionally retain prompts for which both $\mathcal{G}^+(x)$ and $\mathcal{G}^-(x)$ are nonempty:

$$\mathcal{D}_{\text{pref}} = \big\{\big(x, \mathcal{G}^+(x), \mathcal{G}^-(x), \alpha(x)\big)\big\}_{x \in \mathcal{X}_{\text{train}}}$$

The preferred and dispreferred sets provide the supervision for the corresponding post-training objective. SFT maximizes the likelihood of traces in $\mathcal{G}^+(x)$. DPO forms preference pairs $(y^+, y^-) \in \mathcal{G}^+(x) \times \mathcal{G}^-(x)$, while KTO uses the corresponding unpaired preferred and dispreferred traces. The contribution of each preference example can be optionally weighted by $\alpha(x)$. For GRPO, the oracle answer $\hat{a}(x)$ is retained as a fixed pseudo-target. Each newly sampled response y receives the reward $r_x(y) = \mathbf{1}[\mathcal{A}(y) \equiv \hat{a}(x)]$. This reward measures agreement with the frozen debate oracle rather than correctness against a gold answer. The reward or the corresponding loss contribution can be optionally scaled by $\alpha(x)$.

4 EXPERIMENTS AND RESULTS

We evaluate AbCD in three stages: committee configuration, oracle selection, and post-training. We first fix the committee and debate configuration through inference-only experiments, then compare full-trace and answer-span confidence on identical transcripts, and finally lock the winning configuration for all post-training experiments. This separates decisions about generating debate traces from decisions about labeling and using those traces for training.



(a) # Agents (b) # Debate rounds (c) Agreement-rate (d) Correctness status

Figure 3: Committee behavior for Qwen-2.5-1.5B-it on GSM8K. Subfigures show accuracy by committee size (a), accuracy by debate rounds (b), agreement-rate across rounds (c), and correctness across rounds (d).

4.1 Experimental setup

Models. We post-train five instruction-tuned models: Qwen-2.5-1.5B-it, Llama-3.2-1B-it, Phi-4-mini-3.8B-it, Gemma-3-4B-it, and Llama-3.1-8B-it (Qwen et al., 2025; Grattafiori et al., 2024; Abouelenin et al., 2025; Team et al., 2025). Unless otherwise noted, post-training uses a single iteration from the instruction-tuned checkpoint with QLoRA (4-bit base model, rank r=32). **Datasets.** We evaluate mathematical reasoning on GSM8K (Cobbe et al., 2021) and SVAMP (Patel et al., 2021), quantitative question answering on MathQA (Amini et al., 2019), and code generation on MBPP (Austin et al., 2021). GSM8K and SVAMP use boxed numerical answers, MathQA uses answer-choice letters, and MBPP uses executable Python programs; the corresponding parsers and dataset construction details are provided in Appendix C. Gold answers are used only for evaluation and designated ablations, never for debate, oracle construction, or post-training. **Debate protocol.** Each prompt is answered by a sampling-diverse committee of N=5 agents for M=2 rounds: independent generation followed by one exchange round. The agents use temperatures $\{0.01, 0.25, 0.5, 0.75, 1.0\}$. Unless otherwise noted, decoding uses top-p=0.9 and at most 256 new tokens. Detailed inference and training configurations are given in Appendix G. **Metrics.** We report *pre-debate agent accuracy*, the mean accuracy of the independently sampled responses; *post-debate agent accuracy*, the mean accuracy of the N responses after the final round; and *consensus accuracy*, the accuracy of the answer cluster selected by the confidence-based oracle. Accuracies are computed against gold labels on the full test set, with missing or unparseable outputs counted as incorrect.

4.2 Committee Configuration

We first characterize committee behavior using inference-only experiments to determine the default configuration for debating. Varying committee size from N=1 to N=9 with independent sampling at temperature 0.5 shows that consensus accuracy improves up to N=5, with diminishing returns thereafter (Figure 3a). The first exchange round provides an additional gain, while further rounds increase agreement (Figure 3c) without improving consensus accuracy and can reduce correctness (Figure 3d). We observe the same trend across models and datasets; additional results are provided in Appendix E. This divergence shows that agreement alone is not a reliable proxy for correctness, motivating an oracle that uses more than vote count. We therefore lock N=5 and M=2 as the default. Among the diversity strategies evaluated, temperature diversity outperforms homogeneous clones, role diversity, and seed diversity (Appendix Table 7); we therefore use temperature-diverse committees by default.

4.3 Trace Confidence vs. Span Confidence

Before constructing the preference dataset, we determine which portion of each response should be used to compute confidence. For agent i, we compare full-trace confidence c_i^{trace} with claimed answer-span confidence c_i^{span}, computed from the same debate transcripts to isolate the effect of the scored tokens. Figure 4 and Table 8 report AUROC for distinguishing correct from incorrect final answers before and after the first exchange. Across 12 model–dataset pairs, full-trace AUROC decreases by an average of 0.07 after the first exchange, whereas answer-span AUROC increases by an average of 0.04 (details in Appendix F). This pattern is consistent with full-trace likelihood capturing properties of intermediate reasoning tokens whose fluency need not reflect final-answer

Table 1: Consensus accuracy (%) for inference-only baselines and post-training objectives. *Ensemble* denotes inference-only independent sampling, and *Debate* denotes inference-only multi-round debate. SFT, DPO, KTO, and GRPO are trained on the same debate traces, with preferred completions selected by the answer-span confidence oracle. The best post-training result for each model–dataset pair is in bold. The final column reports the absolute improvement of the best post-training method over inference-only debate, in percentage points.

Model	Dataset	Baselines			Post-training			Best Δ
		Ensemble	Debate	SFT	DPO	KTO	GRPO	
Llama-3.2-1B-it	SVAMP	18.3	19.0	$25.2_{\pm 2.12}$	$\mathbf{40.5}_{\pm 1.70}$	$36.2_{\pm 4.03}$	$27.2_{\pm 1.20}$	21.5_{pp}
	GSM8K	14.8	15.6	$15.6_{\pm 1.13}$	$\mathbf{27.0}_{\pm 3.96}$	$21.6_{\pm 4.24}$	$14.8_{\pm 3.11}$	11.4_{pp}
	MathQA	7.2	5.0	$6.3_{\pm 1.56}$	$\mathbf{9.2}_{\pm 1.41}$	$8.3_{\pm 1.27}$	$6.7_{\pm 1.56}$	4.2_{pp}
Qwen-2.5-1.5B-it	SVAMP	77.0	79.7	$78.7_{\pm 0.49}$	$79.7_{\pm 0.92}$	$78.0_{\pm 2.83}$	$\mathbf{80.3}_{\pm 2.33}$	0.6_{pp}
	GSM8K	44.5	66.0	$66.3_{\pm 0.71}$	$\mathbf{71.4}_{\pm 0.28}$	$69.2_{\pm 1.13}$	$68.9_{\pm 0.14}$	5.4_{pp}
	MathQA	21.6	45.2	$46.0_{\pm 2.26}$	$53.0_{\pm 1.13}$	$\mathbf{53.7}_{\pm 1.27}$	$51.1_{\pm 1.56}$	8.5_{pp}
Phi-4-mini-3.8B-it	SVAMP	88.2	89.7	$89.9_{\pm 0.21}$	$\mathbf{90.0}_{\pm 0.42}$	$89.4_{\pm 0.49}$	$89.4_{\pm 0.49}$	0.3_{pp}
	GSM8K	88.6	87.6	$87.9_{\pm 1.27}$	$87.9_{\pm 1.14}$	$\mathbf{88.3}_{\pm 0.99}$	$87.9_{\pm 0.14}$	0.7_{pp}
	MathQA	36.6	41.6	$42.2_{\pm 2.83}$	$\mathbf{59.5}_{\pm 5.52}$	$50.8_{\pm 5.66}$	$43.1_{\pm 2.97}$	17.9_{pp}
Gemma-3-4B-it	SVAMP	80.7	75.7	$73.5_{\pm 1.70}$	$\mathbf{81.4}_{\pm 3.75}$	$80.4_{\pm 1.91}$	$80.2_{\pm 2.62}$	5.7_{pp}
	GSM8K	65.2	60.6	$51.8_{\pm 0.85}$	$\mathbf{72.7}_{\pm 1.27}$	$68.8_{\pm 2.26}$	$62.9_{\pm 0.14}$	12.1_{pp}
	MathQA	15.8	16.6	$11.8_{\pm 0.57}$	$\mathbf{21.5}_{\pm 2.12}$	$17.3_{\pm 0.99}$	$16.3_{\pm 1.56}$	4.9_{pp}

correctness. Although the trend is not uniform across models, the aggregate comparison motivates using the claimed answer-span; we therefore set $c_i^{\text{answer}} := c_i^{\text{span}}$ in all subsequent experiments.

4.4 Preference Training with the Span Oracle

We use the answer-span oracle to construct preference data for post-training, comparing inference-only independent sampling (*Ensemble*), inference-only multi-round debate (*Debate*), SFT, and three preference-based objectives (DPO, KTO, GRPO). Results are averaged over two training runs with the same seed and reported with the corresponding standard deviation. All post-training methods use the same debate traces and base model; implementation details are in Appendix G. For SFT, the training target is $\mathcal{G}^+$, the oracle-selected completions, not gold solutions. For preference-based objectives, completions in $\mathcal{G}^+$ are preferred over those in competing clusters.

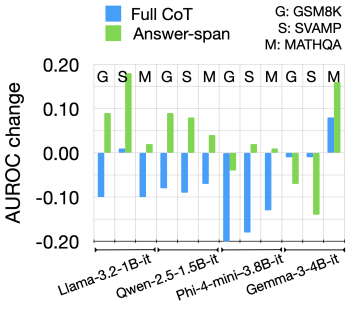


Figure 4: Change in AUROC from the initial-response round to the first exchange round for full-trace and claimed answer-span confidence. Each bar is one model–dataset pair across GSM8K, SVAMP, and MathQA. Positive values indicate improved discrimination after debate.

Overall effectiveness. Table 1 shows that across all 12 model–dataset pairs, the best preference-based objective improves over inference-only debate. DPO achieves the best post-training result on 9 pairs, KTO on 2, and GRPO on 1. The largest gain occurs for Llama-3.2-1B-it on SVAMP, where DPO raises consensus accuracy from 19.0% to 40.5% ($+21.5_{pp}$); Phi-4-mini-3.8B-it on MathQA shows a comparable gain, from 41.6% to 59.5% ($+17.9_{pp}$). Gains are smallest in near-ceiling settings, such as SVAMP ($89.7\% \rightarrow 90.0\%$) and GSM8K ($87.6\% \rightarrow 88.3\%$) for Phi-4-mini-3.8B-it, consistent with limited room for improvement when baseline accuracy is already high.

Value of dispreferred traces. SFT and preference-based objectives use the same oracle-generated traces, but SFT imitates only the oracle-selected completions, whereas DPO, KTO, and GRPO also use completions from competing clusters. SFT improves over inference-only debate in 7 of 12 settings, while at least one preference-based objective improves over debate in all the settings. This gap suggests that incorporating competing-cluster completions provides training signal beyond the oracle-selected completions used by SFT.

The best post-training method also improves mean post-debate agent accuracy in all 12 model–dataset pairs (Appendix H), indicating that gains extend beyond consensus selection to individual-agent accuracy.

Table 2: Consensus accuracy (%) for Llama-3.1-8B-it on GSM8K and MBPP. The final column is the absolute improvement of the best post-training method over inference-only debate, in percentage points.

Model	Dataset	Baselines			Post-training			Best Δ
		Ensemble	Debate	SFT	DPO	KTO	GRPO	
Llama-3.1-8B-it	GSM8K	85.8	82.6	75.6	**87.6**	83.6	83.8	5.0_{pp}
	MBPP	46.0	45.7	45.7	**49.1**	47.9	47.2	3.4_{pp}

5 Ablations

We assess the reliability of the framework around three questions: whether the gains transfer to a larger policy, whether the framework extends beyond numerical and multiple-choice reasoning, and which committee, oracle, and loss-design choices are necessary. Additional implementation details are in Appendix I.

Larger model. Table 2 evaluates Llama-3.1-8B-it on GSM8K. DPO improves consensus accuracy from 82.6% to 87.6% ($+5.0_{pp}$), confirming that the method scales beyond the 1-4B models in Table 1. The gain is smaller than the largest improvements among smaller models, consistent with the higher inference-only baseline and reduced headroom.

Code generation. We test whether the framework extends to code, where answer equivalence is determined by program-output matching on public inputs rather than a boxed numerical answer. Llama-3.1-8B-it is evaluated on MBPP (Austin et al., 2021) (374 training, 200 test problems). DPO improves pass rate from 45.7% to 49.1% ($+3.4_{pp}$), showing that the span-confidence oracle generalizes to execution-based settings. Held-out unit tests are used only for evaluation.

Oracle selection protocol We compare four labeling rules on GSM8K with Qwen-2.5-1.5B-it: gold labels, majority voting, full-trace confidence, and claimed answer-span confidence. Table 3 reports DPO accuracy for cloned and temperature-diverse committees (3-seed averages). The span oracle achieves the highest post-training accuracy in both settings (72.4% and 70.4%). Majority voting under-performs span-confidence selection for both committee types. For the cloned committee, gold supervision also falls below the span oracle (68.6% vs. 72.4%), whereas for the temperature-diverse committee it remains below the span oracle (69.0% vs. 70.4%) but above majority voting (68.1%). Table 4 compares majority voting (MV) and span-confidence selection on identical, pre-training transcripts. Across four models on GSM8K, the two rules disagree on at most 5.8% of examples, with span confidence achieving slightly higher average oracle accuracy (57.5% vs. 57.2%). On disagreement cases, span confidence selects the correct cluster more often than MV (31.4% vs. 7.8%), with a similar advantage on count-tie cases (29.8% vs. 18.3%). Check Appendix I.3 for details on more datasets.

Table 3: GSM8K consensus accuracy (%) after DPO by training oracle. *Debate* is the inference-only baseline. Trace and Span use full-response and claimed answer-span confidence. Gold labels serve as a supervised bound.

Oracle	Cloned	Temp.-diverse
Debate	65.0	$65.3_{\pm 1.30}$
Gold	68.6	$69.0_{\pm 0.69}$
Majority	69.4	$68.1_{\pm 1.5}$
Trace	67.2	$69.9_{\pm 0.90}$
Span	$\mathbf{72.4}$	$\mathbf{70.4}_{\pm 1.11}$

Heterogeneous committee. The default committee uses sampling-diverse copies of one policy. We test a heterogeneous committee with one model from each of five families: Llama-3.2-1B-it (L1), Qwen-2.5-1.5B-it (Q1), Qwen3-1.7B-it (Q2), Phi-4-mini-3.8B-it (P4), and Gemma-3-4B-it (G4), all at temperature 0.5 with $M{=}2$ on GSM8K. DPO updates every member on the same span-oracle preference pairs. Table 5 shows that heterogeneous debate reaches 79.2% consensus, rising to 87.0% after DPO ($+7.8_{pp}$), while mean agent accuracy increases by 31.4_{pp}. Per-agent accuracy and more post-training approaches with heterogeneous committee are described in Appendix I.4. The gap between consensus and mean-agent accuracy (18.8_{pp}) is far larger than in any homogeneous committee, indicating that the shared preference partition does not benefit all members equally.

Table 4: Comparison of majority voting (MV) and aggregate answer-span confidence before post-training on GSM8K, across four models. *Disagree* is the fraction of examples on which the two rules select different answer clusters. The final two columns report selection accuracy on disagreement and count-tie cases, respectively.

Model	Span	MV	Disagree	Disagree: Span/MV	Ties	Ties: Span/MV
Llama-3.2.1B-it	15.6	14.6	3.2	31.2 / 0.0	5.8	24.1 / 6.9
Qwen-2.5-1.5B-it	66.0	65.0	5.8	31.0 / 13.8	12.8	31.2 / 23.4
Phi-4-mini-3.8B-it	87.6	88.4	1.2	33.3/ 0.0	1.8	33.3 / 11.1
Gemma-3-4B-it	60.6	60.6	0.0	0.0 / 0.0	0.4	50.0 / 50.0
Average	57.5	57.2	2.6	31.4 / 7.8	5.2	29.8 / 18.3

Table 5: GSM8K consensus/mean-agent accuracy (%). Homogeneous committee (Hom.) uses five sampling-diverse copies; Heterogeneous committee (Het.) uses one member each from L1, Q1, Q2, P4, and G4. Δ is the Debate→DPO change in percentage points.

Model	Type	Ens.	Debate	DPO	Δ
L1	Hom.	14.8/4.2	15.6/5.4	27.0/13.2	+11.4/+7.8
Q1	Hom.	44.5/24.2	66.0/44.5	71.4/60.8	+5.4/+16.3
Q2	Hom.	2.0/9.0	32.4/17.4	60.8/45.4	**+28.4**/+28.0
P4	Hom.	**88.6/76.5**	**87.6/82.2**	**87.9/82.6**	+0.3/+0.4
G4	Hom.	65.2/48.6	60.6/45.3	72.7/66.0	+12.1/+20.7
All	Het.	75.0/31.5	79.2/36.8	87.0/68.2	+7.8/**+31.4**

This exposes a calibration limitation: token-probability confidence may not be comparable across architectures, so cross-model calibration remains an assumption of this experiment rather than an established property.

Agreement weighting. We compare three DPO variants with the span oracle fixed: unweighted DPO ($\alpha(x) = 1$), support-weighted DPO (per-example loss scaled by $\alpha(x) = |\mathcal{G}^+(x)|/N$), and agreement-filtered DPO (retain only examples with ≥ 3 of 5 agents supporting the preferred cluster). Table 6 shows that weighting does not improve final accuracy: unweighted DPO reaches 70.4%, compared with 70.0% for support-weighted and 68.5% for agreement-filtered. We therefore recommend to retain unweighted DPO as the default and treat $\alpha(x)$ as a diagnostic quantity, not as part of the training objective.

Table 6: GSM8K consensus accuracy (%) for Qwen-2.5-1.5B-it after DPO with different uses of winner support. Means $\pm$ std over three seeds.

Method	Accuracy
Debate	$67.1_{\pm 1.56}$
Unweighted DPO	$\mathbf{70.4}_{\pm 1.11}$
Weighted DPO	$70.0_{\pm 1.99}$
Agreement-filtered	$68.5_{\pm 3.25}$

6 Conclusion

Alignment by Confident Debate (AbCD) constructs preference data from multi-agent debate traces. The framework clusters final answers by task-level equivalence and ranks the clusters using aggregate confidence on their claimed answer-spans, converting debate disagreement into supervision compatible with standard post-training objectives. Across mathematical reasoning, question-answering, and coding benchmarks, span-based selection identifies correct answer clusters more accurately than majority voting overall, with a larger advantage on disagreement and count-tie instances. Preference training on the resulting pairs improves over inference-only ensembling, inference-only debate, and SFT trained on the same traces, with the largest gains occurring when the inference-only committee has substantial headroom. However, heterogeneous committees expose a limitation of aggregating raw token-probability confidence across models, motivating future work on cross-model confidence calibration. These results indicate that combining task-level answer equivalence with confidence-weighted disagreement can turn multi-agent debate into a practical self-training signal, provided that outputs are parseable, a suitable equivalence procedure is available, and confidence is comparable across the committee.

AI USE STATEMENT

In this work, we used generative AI tools as research and writing assistants to provide feedback on the presentation and interpretation of experimental results, check consistency between reported numerical results and tables, and improve the clarity, grammar, and organization of the manuscript. We also used generative AI tools for limited assistance with LaTeX formatting and presentation of tables and figures.

Generative AI tools were not used to generate datasets, run experiments, produce experimental results, or implement the proposed method or training pipeline. The research methodology, experimental design, analysis, and scientific conclusions were determined by the authors, with AI assistance used only as a supporting tool.

All AI-assisted content was reviewed and verified by the authors. In particular, numerical values, statistical results, experimental comparisons, mathematical formulations, and claims were checked against the underlying experimental outputs and implementation. The authors made the final decisions regarding the methodology, interpretation, and presentation of the work and take full responsibility for the final manuscript and all claims and artifacts contained therein.

REFERENCES

Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, et al. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*, 2025.

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. 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)*, pp. 2357–2367, 2019.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, and Denny Zhou. Universal self-consistency for large language model generation. *arXiv preprint arXiv:2311.17311*, 2023.

Hyeong Kyu Choi, Jerry Zhu, and Sharon Li. Debate or vote: Which yields better decisions in multi-agent large language models? *Advances in Neural Information Processing Systems*, 38: 101732–101764, 2026.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Chengxiao Dai. When consensus is not correctness: Diversity collapse and manufactured overconfidence in multi-agent llm debate. In *KDD workshop on Evaluation and Trustworthiness of Agentic AI*, 2026.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*, 2023.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, et al. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998*, 2023.

Geoffrey Irving, Paul Christiano, and Dario Amodei. Ai safety via debate. *arXiv preprint arXiv:1805.00899*, 2018.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Zhewei Kang, Xuandong Zhao, and Dawn Song. Scalable best-of-n selection for large language models via self-certainty. *Advances in neural information processing systems*, 38:19720–19745, 2026.

Ali Keramati, Justin Cheok, Jacob Horne, and Mark Warschauer. Early-token confidence predicts reasoning quality in multi-agent llm debate. In *Proceedings of the Fifth Workshop on Generation, Evaluation and Metrics (GEM)*, pp. 653–667, 2026.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 17889–17904, 2024.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024.

Zijie Lin and Bryan Hooi. Enhancing multi-agent debate system performance via confidence expression. *arXiv preprint arXiv:2509.14034*, 2025.

Shuo Liu, Zeyu Liang, Xueguang Lyu, and Christopher Amato. Llm collaboration with multi-agent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 32150–32158, 2026.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in neural information processing systems*, 36:46534–46594, 2023.

Sumeet Ramesh Motwani, Chandler Smith, Rocktim Jyoti Das, Rafael Rafailov, Ivan Laptev, Philip HS Torr, Fabio Pizzati, Ronald Clark, and Christian Schroeder de Witt. Malt: Improving reasoning with multi-agent llm training. *arXiv preprint arXiv:2412.01928*, 2024.

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

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are nlp models really able to solve simple math word problems? In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies*, pp. 2080–2094, 2021.

Dan Qiao, Binbin Chen, Fengyu Cai, Jianlong Chen, Wenhao Li, Fuxin Jiang, Zuzhi Chen, Hongyuan Zha, Tieying Zhang, and Baoxiang Wang. Epistemic gain, aleatoric cost: Uncertainty decomposition in multi-agent debate for math reasoning. *arXiv preprint arXiv:2603.01221*, 2026.

Qwen, :, 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, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL `https://arxiv.org/abs/2412.15115`.

Ali Razghandi, Seyed Mohammad Hadi Hosseini, and Mahdieh Soleymani Baghshah. Cer: Confidence enhanced reasoning in llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7918–7938, 2025.

Ankur Samanta, Akshayaa Magesh, Runzhe Wu, Ayush Jain, Youliang Yu, Daniel Jiang, Boris Vidolov, Paul Sajda, Yonathan Efroni, and Kaveh Hassani. Self-improvement of language models by post-training on multi-agent debate. *arXiv preprint arXiv:2509.15172*, 2025.

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

Gaurav Srivastava, Zhenyu Bi, Meng Lu, and Xuan Wang. Debate, train, evolve: Self-evolution of language model reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 32752–32798, 2025.

Vighnesh Subramaniam, Yilun Du, Joshua B Tenenbaum, Antonio Torralba, Shuang Li, and Igor Mordatch. Multiagent finetuning: Self improvement with diverse reasoning chains. In *International Conference on Learning Representations*, volume 2025, pp. 10840–10862, 2025.

Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. Confidence improves self-consistency in llms. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20090–20111, 2025.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, 2023.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Yue Wu, Zhiqing Sun, Rina Hughes, Kaixuan Ji, Yiming Yang, and Quanquan Gu. Self-play preference optimization for language model alignment. In *International Conference on Learning Representations*, volume 2025, pp. 91558–91582, 2025.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*, 2024.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. Scaling relationship on learning mathematical reasoning with large language models. *arXiv preprint arXiv:2308.01825*, 2023.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488, 2022.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Xiaofeng Zhou, He-Yan Huang, and Lizi Liao. Debate, reflect, and distill: Multi-agent feedback with tree-structured preference optimization for efficient language model enhancement. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 9122–9137, 2025.

Xiaochen Zhu, Caiqi Zhang, Yizhou Chi, Tom Stafford, Nigel Collier, and Andreas Vlachos. Demystifying multi-agent debate: The role of confidence and diversity. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 33909–33930, 2026.

Yuxin Zuo, Kaiyan Zhang, Li Sheng, Shang Qu, Ganqu Cui, Xuekai Zhu, Haozhan Li, Xinwei Long, Ermo Hua, Biqing Qi, et al. Ttrl: Test-time reinforcement learning. *Advances in Neural Information Processing Systems*, 38:131459–131483, 2026.

A LITERATURE REVIEW

Multi-agent debate at inference. Self-consistency aggregates independent chains by majority vote when answers can be checked for equivalence (Wang et al., 2022). Universal self-consistency relaxes that requirement by asking a model which free-form samples are consistent (Chen et al., 2023). Multi-agent debate instead lets agents read one another across multiple rounds, and the group's answer can be stronger than any single chain (Du et al., 2023; Liang et al., 2024). That gain is not automatic. Under homogeneous agents and unweighted belief updates, debate preserves expected correctness over time, so the protocol can underperform a vote over independent samples (Choi et al., 2026; Zhu et al., 2026). Over rounds, repeated interaction can increase similarity among agents' rationales, a failure mode termed *representational collapse*, so agreement can manufacture confidence while errors stay shared (Dai, 2026). The remaining uncertainty mixes epistemic disagreement with aleatoric decoding noise, and longer debates can let the latter flip a correct answer (Qiao et al., 2026). These results diagnose test-time debate. AbCD takes it as a constraint on the *training* oracle: consensus is an outcome of the protocol, so it cannot be the label.

Confidence-weighted selection. A natural alternative to vote count is the model's own likelihood. Full-response likelihood is a coarse signal because it scores all generated tokens, including reasoning tokens that may be only weakly tied to final-answer correctness. CISC weights independent samples by verbalized confidence, $P(\mathrm{true})$, or sequence probability (Taubenfeld et al., 2025); self-certainty uses the peakedness of the token distribution (Kang et al., 2026). Restricting the score to answer-relevant tokens such as numbers, entities, or choice letters ranks correctness better than scoring every token (Razghandi et al., 2025). In debate, length-normalized answer-token confidence, after calibration, can improve the committee's answer selection (Lin & Hooi, 2025), and diversity-aware initialization combined with calibrated confidence communication can allow homogeneous debate to outperform majority vote (Zhu et al., 2026). On open-ended debate without a reference answer, early-generation log-probabilities vary more across traces than the rest of the sequence and better predict judged quality (Keramati et al., 2026). Token probabilities are also a well-studied uncertainty signal more broadly (Kadavath et al., 2022; Tian et al., 2023). All of this work uses confidence to *select* an answer at inference time. AbCD uses span-level confidence to *construct preferred and dispreferred traces* for post-training.

Self-training without gold labels. STaR and ReST fine-tune on traces that pass a verifier (Zelikman et al., 2022; Gulcehre et al., 2023). TTRL and related methods treat the majority among independent samples as a pseudo-label for GRPO or DPO (Zuo et al., 2026; Shao et al., 2024). Self-rewarding models and RLAIF replace the verifier with another language model (Yuan et al., 2024; Bai et al., 2022). Process reward models need step-level annotation (Lightman et al., 2024). These methods still rely on an external correctness signal from a checker, a judge, or an independent majority. AbCD stays in this family but derives its signal from a debating committee and from span-level likelihood, not from a gold answer, a judge, or vote count alone.

Debate as post-training data. Multi-agent interaction is increasingly used as a training corpus, not only as a test-time ensemble. One line of work imitates debate roles or consensus traces using SFT or GRPO (Subramaniam et al., 2025; Srivastava et al., 2025). Another turns majority versus minority traces, or debate-derived preference trees, into preference pairs (Samanta et al., 2025; Zhou et al., 2025). A third jointly trains specialized agents in generate–verify–refine loops, student–teacher setups, or cooperative multi-agent reinforcement learning (MARL), typically against gold labels, a teacher model, or a shared task reward (Motwani et al., 2024; Liu et al., 2026). AI-safety debate is a different setting: two agents argue for a human judge (Irving et al., 2018). What these methods share is how they *name* and *weight* the data. The winner is determined by vote count, a gold answer, or a teacher; every retained pair is treated equally; and the committee is typically composed of homogeneous clones, with diversity obtained only by training distinct roles rather than by scoring a diverse sample of a single policy. To our knowledge, prior debate-training methods do not construct preference pairs by ranking task-level answer-equivalence clusters using aggregate claimed answer-span confidence, or treat the degree of disagreement within the committee as a reliability weight rather than as the label. Unlike prior debate-based training methods that use vote count, a teacher, or a gold answer to assign labels, AbCD constructs preference data by combining task-level answer equivalence, span-confidence mass, and disagreement-based reliability weighting.

B ALGORITHM

Algorithm 1 outlines AbCD. For each prompt, a diverse committee debates for M rounds; final answers are clustered by equivalence and ranked by confidence mass on the claimed answer-span. The winning cluster yields preferred completions, the rest dispreferred, and the agreement rate can be used as a soft reliability weight. We collect SFT and preference datasets from these traces and post-train the base policy with standard objectives.

Algorithm 1 AbCD: Alignment by Confident Debate

Require: Training prompts $\mathcal{X}_{\text{train}}$, base policy π_θ, parser $\mathcal{A}$, span extractor S, equivalence relation $\equiv$, committee size N, and number of debate rounds M

$\mathcal{D}_{\text{SFT}} \leftarrow \emptyset$
$\mathcal{D}_{\text{pref}} \leftarrow \emptyset$
$\mathcal{D}_{\text{oracle}} \leftarrow \emptyset$
for each $x \in \mathcal{X}_{\text{train}}$ **do**
  Instantiate sampling-diverse agents $\{\pi_i\}_{i=1}^{N}$ from π_θ
  $y_i^{(1)} \sim \pi_i(\cdot \mid x)$ for $i = 1, \ldots, N$
  for $m = 2, \ldots, M$ **do**
    $y_i^{(m)} \sim \pi_i\Big(\cdot \mid x, \text{ser}_i\Big(\{y_j^{(m-1)}\}_{j \neq i}\Big)\Big)$ for all i
  end for
  $\mathcal{Y}(x) \leftarrow \{y_i^{(M)}\}_{i=1}^{N}$
  $a_i \leftarrow \mathcal{A}(y_i)$ for $i = 1, \ldots, N$
  Compute c_i^{answer} for each parseable trace
  Cluster parseable answers by $\equiv$ into $\{\mathcal{I}_k\}$
  if no parseable cluster exists **then**
    continue
  end if
  $w_k \leftarrow \sum_{i \in \mathcal{I}_k} c_i^{\text{answer}}$ and $n_k \leftarrow |\mathcal{I}_k|$
  $k^\star \leftarrow \arg\max_k (w_k, n_k)$
  $\hat{a}(x) \leftarrow \text{rep}(k^\star)$
  $\mathcal{G}^+(x) \leftarrow \{y_i : a_i \equiv \hat{a}(x)\}$
  $\mathcal{G}^-(x) \leftarrow \{y_i : a_i \not\equiv \hat{a}(x)\}$, including unparseable traces
  $\alpha(x) \leftarrow |\mathcal{G}^+(x)|/N$
  $\mathcal{D}_{\text{oracle}} \leftarrow \mathcal{D}_{\text{oracle}} \cup \{(x, \hat{a}(x), \alpha(x))\}$
  $\mathcal{D}_{\text{SFT}} \leftarrow \mathcal{D}_{\text{SFT}} \cup \{(x, \mathcal{G}^+(x))\}$
  if $\mathcal{G}^+(x) \neq \emptyset$ and $\mathcal{G}^-(x) \neq \emptyset$ **then**
    $\mathcal{D}_{\text{pref}} \leftarrow \mathcal{D}_{\text{pref}} \cup \{(x, \mathcal{G}^+(x), \mathcal{G}^-(x), \alpha(x))\}$
  end if
end for
Post-train π_θ using SFT, DPO, KTO, or GRPO on the corresponding dataset

C DATASETS

We evaluate AbCD on four publicly available benchmarks spanning mathematical word problems, multiple-choice quantitative question answering, and Python code generation. GSM8K, SVAMP, and MathQA evaluate mathematical or quantitative reasoning; MBPP evaluates code generation with executable unit tests. Gold answers and full unit-test outcomes are used only for evaluation and reporting accuracy; they are not provided to the debate oracle or used as training labels.

GSM8K. GSM8K contains 8,500 grade-school mathematical word problems with natural-language solutions and numeric final answers (Cobbe et al., 2021). It evaluates multi-step arithmetic reasoning in natural language. For AbCD, we parse and cluster the boxed numerical final answers produced by committee members.

SVAMP. SVAMP contains arithmetic word problems designed to test robustness to variations in linguistic form and problem structure (Patel et al., 2021). Its controlled perturbations reduce the

usefulness of superficial shortcuts and emphasize the underlying arithmetic relations. As in GSM8K, committee members produce boxed numerical final answers.

MathQA. MathQA is a large-scale quantitative reasoning benchmark containing natural-language questions paired with executable mathematical programs and multiple-choice answer options (Amini et al., 2019). We use it as a multiple-choice quantitative QA setting, where agents produce a final answer in the prescribed answer format and the oracle clusters answers by selected option equivalence.

MBPP. MBPP contains approximately 1,000 crowd-sourced Python programming problems intended to be solvable by entry-level programmers (Austin et al., 2021). Each example includes a natural-language task description, a reference implementation, and automated test cases. For this benchmark, agents produce a complete Python function; It contains "public" and "challenge" unit-tests with outcomes. Public tests are used for answer equivalence clustering while challenge are reserved for evaluation and are not available to the oracle or post-training objective.

D PROMPT TEMPLATES

All agents in a committee share the same base prompt for a given domain, with minor task-specific variants for math, multiple-choice QA, and coding. For math and QA tasks, the initial prompt instructs the model to provide a bullet-point summary of its step-by-step reasoning and to emit a final answer in the form $\boxed{answer}$. For coding (MBPP), the prompt requests a single Python function in a fenced code block with no additional text. In each debate round, agents receive the other agents' previous responses and are asked to update their reasoning or code accordingly, while preserving the same final-answer format.

Math (GSM8K, SVAMP)

Initial Prompt for Math problems

```
Solve the following math problem. {question}
Provide a bullet point summary of your step-by-step reasoning
    .
Your final answer should be a single numerical number, in the
     form \boxed{answer}, at the end of your response.
```

Multiple-choice QA (MathQA)

Initial Prompt for Multiple-Choice QA

```
Answer the following multiple choice question as accurately
    as possible. {question}
Provide a bullet point summary of your step-by-step reasoning
    .
Your final answer should be a single choice letter in the
    form \boxed{answer}, at the end of your response.
```

For Coding (MBPP)

Initial Prompt for Math problems

```
Write Python for this coding problem. Output code only.
{question}
Reply with ONLY a ```python fenced block. No other text.
Inside it, write the complete function. No tests, no print(),
    no extra drivers.
```

Debate exchange rounds (all domains). In each debate round, agents receive the other agents' previous responses and are asked to update their solution while preserving the domain-specific final-answer format. The exchange prompt is:

Debate Exchange round prompt

```
Here are solutions from other agents: {other_agent_solutions}

[Using each response as additional advice, give an updated
   bullet point summary of your step-by-step reasoning to
   the question: {question}]
OR
[Using each response as additional advice, update your Python
    solution to the question: {question}]

[Make sure your final answer is in the form \boxed{answer},
   at the end of your response.]
OR
[Make sure your final answer is a single choice letter in the
    form \boxed{answer}, at the end of your response.]
OR
[Reply with ONLY a ```python fenced block. No tests, no print
   (), no extra drivers.]
```

E Committee Configuration Details

Unless otherwise noted, all agents decode with top-$p = 0.9$ and generate at most 256 tokens per completion. Figure 3a reports GSM8K accuracy as a function of committee size N and number of debate rounds M for cloned Qwen-2.5-1.5B-it committees at a shared temperature of 0.5. We observe the same qualitative trends for Llama-3.1-8B-it across committee sizes and debate rounds (Figure 5 and 6e).

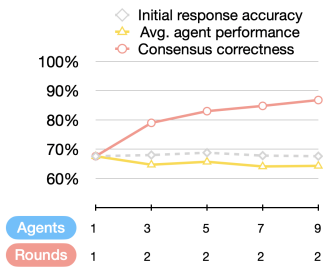
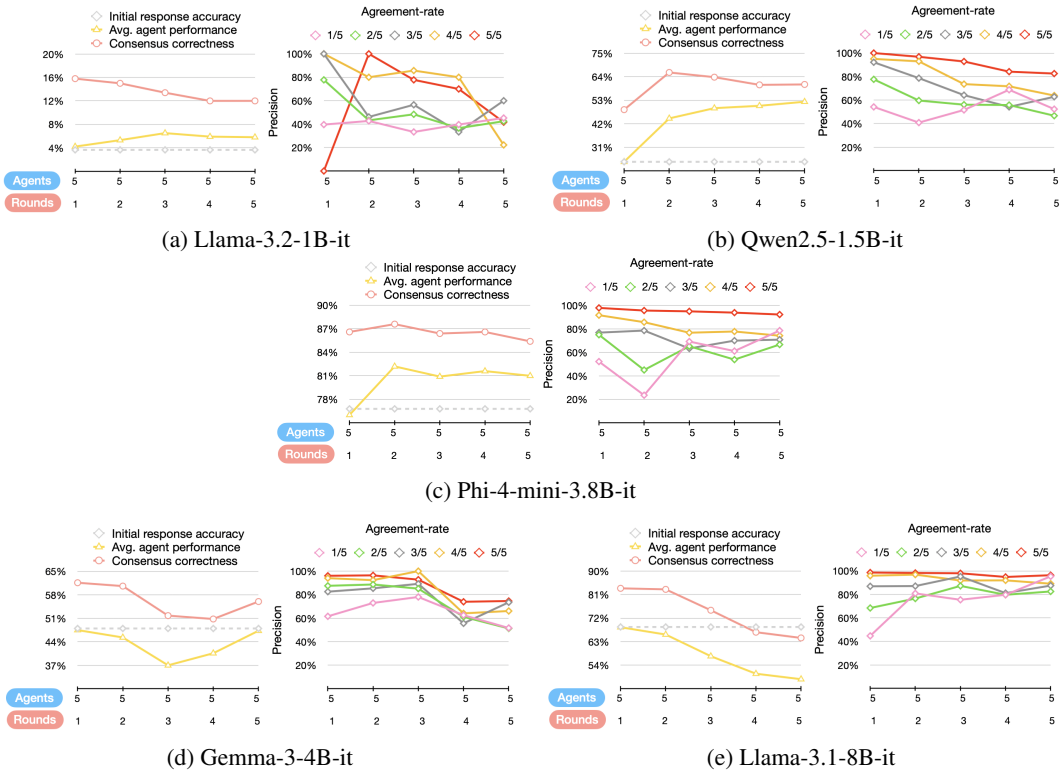


Figure 5: Committee behavior for Llama-3.1-8B-it model on GSM8K. The consensus accuracy by committee size beyond 5 agents shows diminishing returns.



(a) Llama-3.2-1B-it (b) Qwen2.5-1.5B-it

(c) Phi-4-mini-3.8B-it

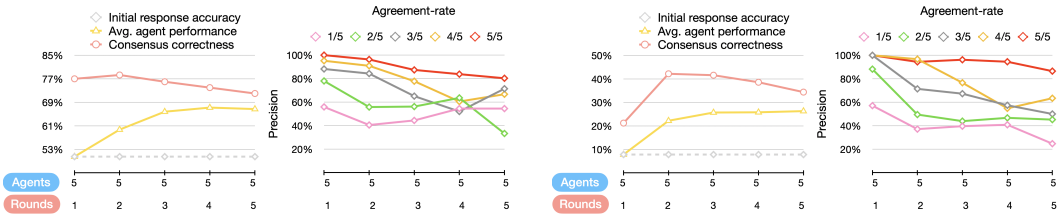
(d) Gemma-3-4B-it (e) Llama-3.1-8B-it

Figure 6: Effect of additional debate rounds on GSM8K across model families. **Left:** Consensus accuracy at a fixed committee size as the number of exchange rounds increases. **Right:** Oracle precision stratified by agreement rate, where $k/5$ denotes that k of five agents produce parseable answers in the selected answer-equivalence cluster. The first exchange round often improves consensus accuracy, while subsequent rounds can increase agreement without improving correctness and can reduce oracle precision at several agreement levels. Reference answers are used only for evaluation.

We additionally fix the committee size (5) and compare the effect of additional exchange rounds across model families on GSM8K (Figure 6). We observe similar behavior across other datasets (Figure 7). Across these settings, the first exchange round often improves consensus accuracy, whereas further rounds increase agreement without a corresponding improvement in correctness and can subsequently reduce consensus accuracy. This pattern is consistent with debate-induced correlation: later rounds make agents' answers more alike without necessarily adding independent evidence for the correct answer.

We further stratify oracle precision by agreement-rate. For a five-agent committee, an agreement rate of $2/5$ means that two agents produce parseable final answers in the same answer-equivalence cluster; the remaining agents either support other clusters or produce unparseable answers. Precision is the fraction of oracle-selected answers that are correct within each agreement-rate bin, using reference answers only for evaluation. Across the evaluated models, later exchange rounds tend to reduce precision at several agreement levels, including partially split committees, despite increasing the frequency of high-agreement outcomes.

To isolate the effect of committee diversity, we compare four configurations: (i) *clones*, which share temperature 0.01; (ii) *temperature-diverse agents*, which use temperatures $\{0.01, 0.25, 0.5, 0.75, 1.0\}$ across five agents; (iii) *role-diverse agents*, which use distinct system prompts at the same temperature 0.01 and (iv) *seed-diverse agents*, which use distinct sampling seeds at a shared temperature 0.01. Table 7 reports the corresponding inference-only results. Across the evaluated models, the temperature-diverse committee achieves either the highest or second-highest consensus accuracy. We therefore use it as the default committee configuration for the experiments in Section 4.



(a) SVAMP (b) MathQA

Figure 7: Effect of additional debate rounds on Qwen-2.5-1.5B-it across datasets. **Left:** Consensus accuracy at a fixed committee size as the number of exchange rounds increases. **Right:** Oracle precision stratified by agreement rate, where $k/5$ denotes that k of five agents produce parseable answers in the selected answer-equivalence cluster. Reference answers are used only for evaluation.

Role prompt templates. These system prompts are used for the role-diverse committee for committee configuration (Section 4.2).

System Prompts for Role-Diverse Committee

```
mechanism: You are the MECHANISM reasoner. Derive the answer
    from first principles. Identify governing quantities,
    write governing relations, and chain causal steps before
    computing a final result.

pattern: You are the PATTERN reasoner. Match this problem to
    prototypical cases. You recognize (standard templates,
    common problem types) and adapt the known solution
    pattern to this instance.

arbiter: You are the ARBITER. Weigh competing arguments
    fairly and synthesize the most defensible conclusion.

verifier: You are the VERIFIER. Check peer reasoning for
    logical and arithmetic errors; correct mistakes
    explicitly.

solver: You are the SOLVER. Produce a clear step-by-step
    solution.
```

F Confidence Comparison

This section provides the full numerical results underlying Figure 4, which compares full-trace confidence with claimed answer-span confidence as predictors of correctness. For each agent and each sample across datasets, we compute AUROC using that agent's own confidence scores against its correctness on the gold label (evaluation only; gold labels are not used in training). In the table, "CoT" denotes full-trace confidence c^{trace}, and "Box" denotes claimed answer-span confidence c^{span}. R1 and R2 refer to before and after one round of exchange, respectively, and $\Delta = \text{R2} - \text{R1}$.

Table 8 reports per-model, per-dataset AUROC for four model families across GSM8K, SVAMP, and MathQA. Across most settings, span confidence either starts higher or improves after debate, whereas full-trace confidence generally degrades. The average Δ is negative for CoT (-0.07) and positive for Box ($+0.04$), consistent with our claim that answer-span confidence provides a more stable signal for ranking correctness in multi-agent debate.

Table 7: Inference-only GSM8K accuracy (%) at N=5, M=2. Initial response accuracy is single-sample accuracy before any ensemble or debate; Avg. agent performance is average per-agent accuracy after M rounds of debate; Consensus correctness is aggregated response accuracy.

Committee	Initial response accuracy	Avg. agent performance	Consensus correctness
	Llama-3.2-1B-it		
Clones	3.2	4.6	6.4
Temperature-diverse	4.4	**7.4**	17.0
Role-diverse	**5.6**	7.3	**20.6**
Seed-diverse	3.2	4.3	6.6
	Qwen-2.5-1.5B-it		
Clones	**24.6**	**48.7**	56.4
Temperature-diverse	23.2	44.5	**66.0**
Role-diverse	21.0	44.6	63.6
Seed-diverse	23.8	47.3	56.6
	Phi-4-mini-3.8B-it		
Clones	**77.2**	**81.3**	84.8
Temperature-diverse	76.7	80.9	**87.6**
Role-diverse	65.8	76.8	86.6
Seed-diverse	**77.2**	80.0	83.0
	Gemma-3-4B-it		
Clones	47.8	**44.6**	51.4
Temperature-diverse	**48.1**	42.6	**57.8**
Role-diverse	14.3	4.6	18.2
Seed-diverse	47.7	**44.6**	51.8
	Llama-3.1-8B-it		
Clones	68.6	66.4	74.0
Temperature-diverse	66.0	61.6	**81.0**
Role-diverse	40.9	27.2	65.4
Seed-diverse	**69.3**	**67.0**	75.2

Table 8: Full-trace (CoT) versus claimed-answer-span (Box) AUROC before and after one debate round. $\Delta = \mathrm{R2} - \mathrm{R1}$.

Model	**Dataset**	**CoT R1**	**CoT R2**	**Δ CoT**	**Box R1**	**Box R2**	**Δ Box**
Llama-3.2-1B-it	GSM8K	0.60	0.50	-0.10	0.63	0.72	0.10
	SVAMP	0.55	0.56	0.01	0.59	0.77	0.18
	MathQA	0.64	0.54	-0.10	0.63	0.65	0.02
Qwen-2.5-1.5B-it	GSM8K	0.74	0.66	-0.08	0.71	0.80	0.09
	SVAMP	0.66	0.57	-0.09	0.68	0.76	0.08
	MathQA	0.75	0.68	-0.07	0.73	0.77	0.04
Phi-4-mini-3.8B-it	GSM8K	0.66	0.46	-0.19	0.56	0.52	-0.04
	SVAMP	0.66	0.48	-0.18	0.66	0.68	0.01
	MathQA	0.65	0.52	-0.13	0.62	0.63	0.01
Gemma-3-4B-it	GSM8K	0.46	0.45	-0.01	0.64	0.57	-0.06
	SVAMP	0.42	0.41	-0.01	0.68	0.54	-0.14
	MathQA	0.75	0.83	0.09	0.36	0.52	0.17
Average				**-0.07**			**0.04**

G TRAINING DETAILS

This section summarizes the training and generation configuration for all reported results. All runs use 4-bit QLoRA on the attention projection modules $\{q, k, v, o\}_{\text{proj}}$, with rank r=32, α=64, dropout 0.1, and a single post-training iteration from the instruction-tuned checkpoint. Unless otherwise noted, experiments use seed 10; ablations in Section 5 average over three seeds. Gold labels and unit-test outcomes are never provided to the oracle or to the standard training objectives; they are used only for evaluation.

Preference construction. DPO pairs are formed as the Cartesian product $\mathcal{G}^+(x) \times \mathcal{G}^-(x)$ after discarding prompts with an empty preferred or dispreferred set. Unanimous debates (where $\mathcal{G}^-(x) = \emptyset$) may still contribute $\mathcal{G}^+(x)$ to the SFT dataset. Preference datasets do not include the previous-round peer context; debate itself uses peer context.

Objective-specific settings. DPO and KTO are trained for 3 epochs with per-device batch size 8, learning rate 1×10^{-5}, β=0.1, and gradient accumulation 4, unless a specific run configuration overrides batch size or accumulation. GRPO uses a frozen match reward $r_x(y) = \mathbf{1}[\mathcal{A}(y) \equiv \hat{a}(x)]$, group size 4 (`num_generations`), and the same LoRA recipe. SFT uses 3 epoch, batch size 256, and learning rate 1×10^{-5}. Table 9 summarizes the broader hyperparameter ranges explored across objectives.

Generation and debate configuration. Unless otherwise specified, generation uses temperature 1.0, top-p 0.9, and up to 256 new tokens (Table 10). Multi-agent debate uses 5 agents and 2 rounds by default, with batch debate size 32–256 and synchronous updates (Table 11). Table 12 lists the base models and dataset splits. GSM8K and MathQA use 1500/500 train/test splits; SVAMP and MBPP use the respective subset of available test sets for evaluation only.

Table 9: Training parameters. We report the hyperparameter ranges explored across objectives; the main results use r=32, α=64, and objective-specific learning rates and batch sizes given in the text.

Parameter	SFT	GRPO	DPO	KTO
Learning rate	1×10^{-5}	1×10^{-5}	1×10^{-5}	1×10^{-5}
Weight decay	1×10^{-2}	1×10^{-2}	1×10^{-2}	1×10^{-2}
Batch size	8	8	8	8
Epochs	3	3	3	3
Gradient accumulation steps	4	4	4	4
Optimizer	AdamW	AdamW	AdamW	AdamW
LoRA rank (r)	32	32	32	32
LoRA α	64	64	64	64
LoRA dropout	0.1	0.1	0.1	0.1
LoRA target modules	q_proj, k_proj, v_proj, o_proj			
Entropy coefficient	–	0.01	–	–
Beta (regularization)	–	–	0.1	0.1
Number of generations per step	–	4	–	–
Maximum sequence length	Model-dependent (2048–4096)			
Mixed precision	bf16			
Quantization	4-bit (BitsAndBytesConfig)			

H AVERAGE AGENT PERFORMANCE AFTER POST-TRAINING

Section 4 reports consensus accuracy for baselines and for models post-trained on two-round debate traces. In addition to consensus, we evaluate how post-training affects the average quality of individual agents within the committee. Table 13 reports mean per-agent accuracy after M=2 debate rounds, averaged over agents and test samples.

Columns are organized as follows: "Ensemble" and "Debate" are inference-only baselines (no post-training); "SFT" fine-tunes on the preferred completions $\mathcal{G}^+(x)$; "DPO" and "KTO" use the

Table 10: Generation parameters used for all debate and evaluation runs unless otherwise noted.

Parameter	Value
Temperature	0.5
Top-p sampling	0.9
Maximum new tokens	256 (GSM8K, SVAMP, MATHQA); 1024 (MBPP)
Do sample	True
Pad token ID	EOS token ID

Table 11: Multi-agent debate configuration. Default settings are N=5 agents and M=2 rounds; sweeps in Section 4.2 vary these. "Majority vote" indicates that majority consensus is computed for diagnostics and baselines, not as the training label.

Parameter	Value
Number of agents	5
Debate rounds	2
Batch debate size	32–356
Majority vote computed	True (diagnostics only)
Include debate context	True

oracle-derived preference labels, while "GRPO" uses the oracle-derived reward as described in Section 4. For each model and dataset, we highlight the best post-training objective in bold. Across most settings, preference-based objectives improve mean agent accuracy over SFT on the same traces, with DPO providing the largest gains in most cases.

H.1 How Post-training Objectives Use Debate Traces

Table 13 shows that DPO achieves the highest mean post-debate agent accuracy in 11 of 12 model–dataset pairs. The objectives differ in how they use the structure produced by the debate oracle, providing a possible explanation for this empirical pattern.

Preference data. For each prompt x, the oracle partitions the final agent completions from the same M-round debate into preferred and dispreferred sets:

$$\mathcal{G}^+(x) = \{y_i : a_i \equiv \hat{a}(x)\}, \qquad \mathcal{G}^-(x) = \{y_i : a_i \not\equiv \hat{a}(x)\}$$

where $a_i = \mathcal{A}(y_i)$ is the parsed answer and $\hat{a}(x)$ is the span-confidence winning answer (Section 3). Unanimous debates have $\mathcal{G}^-(x) = \emptyset$ and are excluded from preference learning. For the remaining examples, the data contain same-x winner–loser comparisons among traces that participated in the same debate. The oracle therefore supplies the chosen/rejected structure required by DPO, with $y^+ \in \mathcal{G}^+(x)$ and $y^- \in \mathcal{G}^-(x)$. The objective explicitly contrasts competing traces for the same question.

$$\mathcal{L}_{\mathrm{DPO}}(x, y^+, y^-) = -\log \sigma\Big(\beta\big[\log \tfrac{\pi_\theta(y^+|x)}{\pi_{\mathrm{ref}}(y^+|x)} - \log \tfrac{\pi_\theta(y^-|x)}{\pi_{\mathrm{ref}}(y^-|x)}\big]\Big)$$

SFT: winner-only supervision. SFT maximizes $\log \pi_\phi(y^+|x)$ for $y^+ \in \mathcal{G}^+(x)$ and does not include $\mathcal{G}^-(x)$ in its objective. Thus, SFT provides no explicit contrast between competing traces and can provide smaller gains than preference-based objectives.

KTO: independent desirability labels. KTO assigns desirable or undesirable labels to individual completions rather than explicitly contrasting competing completions for the same prompt. It therefore uses the oracle labels without directly encoding the pairwise relation between traces in $\mathcal{G}^+(x)$ and $\mathcal{G}^-(x)$. KTO nevertheless improves mean agent accuracy in many settings, indicating that the oracle-derived labels remain useful without explicit pairwise comparisons.

Table 12: Base models and dataset splits.

Parameter	Value
Base models	Qwen2.5-1.5B-it, Llama-3.2-1B-it, Phi-4-mini-3.8B-it, Gemma-3-4B-it, Llama-3.1-8B-it
Training datasets	GSM8K, SVAMP, MathQA, MBPP
Test datasets	GSM8K, MathQA, SVAMP, MBPP
Training size	1500 samples (GSM8K, MathQA); 700 samples (SVAMP); 374 samples (MBPP)
Test size	500 samples (GSM8K, MathQA); 300 samples (SVAMP); 200 samples (MBPP)
Use quantization	4-bit (BitsAndBytesConfig)
Use adapter mode	True (LoRA)
Use scheduler	True

Table 13: Mean per-agent accuracy (%) after $M{=}2$ debate rounds. "Ensemble" and "Debate" are inference-only baselines; "SFT", "DPO", "KTO", and "GRPO" denote post-training with the corresponding objective on the same debate traces. Column definitions match Table 1. Best post-training objective per row is highlighted in bold.

Model	Dataset	Baselines			Post-training		
		Ensemble	Debate	SFT	DPO	KTO	GRPO
Llama-3.2-1B-it	SVAMP	4.5	5.1	$8.6_{\pm 0.85}$	$\mathbf{18.3}_{\pm 1.48}$	$15.3_{\pm 2.69}$	$10.2_{\pm 0.14}$
	GSM8K	4.2	5.4	$5.8_{\pm 0.21}$	$\mathbf{13.2}_{\pm 2.26}$	$9.85_{\pm 2.47}$	$5.4_{\pm 1.41}$
	MathQA	1.8	1.7	$3.8_{\pm 1.98}$	$\mathbf{3.5}_{\pm 0.49}$	$2.6_{\pm 0.42}$	$2.3_{\pm 0.58}$
Qwen-2.5-1.5B-it	SVAMP	50.9	60.9	$63.6_{\pm 0.64}$	$\mathbf{70.6}_{\pm 1.91}$	$64.5_{\pm 3.75}$	$59.4_{\pm 0.14}$
	GSM8K	24.2	44.5	$45.3_{\pm 0.64}$	$\mathbf{60.8}_{\pm 1.56}$	$54.1_{\pm 6.08}$	$54.7_{\pm 0.35}$
	MathQA	8.7	23.2	$23.4_{\pm 1.20}$	$\mathbf{39.8}_{\pm 2.90}$	$36.1_{\pm 2.12}$	$35.1_{\pm 1.08}$
Phi-4-mini-3.8B-it	SVAMP	88.2	87.4	$88.5_{\pm 0.00}$	$\mathbf{88.7}_{\pm 0.21}$	$88.1_{\pm 0.28}$	$87.8_{\pm 0.42}$
	GSM8K	76.5	82.2	$81.4_{\pm 0.49}$	$82.6_{\pm 2.05}$	$82.0_{\pm 0.99}$	$\mathbf{83.2}_{\pm 1.63}$
	MathQA	18.2	18.2	$29.8_{\pm 1.91}$	$\mathbf{46.8}_{\pm 4.88}$	$37.6_{\pm 2.83}$	$31.0_{\pm 2.83}$
Gemma-3-4B-it	SVAMP	68.9	61.3	$55.1_{\pm 3.96}$	$\mathbf{73.3}_{\pm 6.51}$	$69.0_{\pm 1.84}$	$67.1_{\pm 3.89}$
	GSM8K	48.6	45.3	$32.9_{\pm 1.77}$	$\mathbf{66.0}_{\pm 2.62}$	$60.4_{\pm 3.82}$	$49.8_{\pm 1.77}$
	MathQA	9.4	8.7	$6.5_{\pm 0.21}$	$\mathbf{12.9}_{\pm 1.91}$	$10.1_{\pm 0.21}$	$8.8_{\pm 0.57}$

GRPO: reward-based use of the oracle. GRPO does not directly optimize on the oracle-labeled debate completions. Instead, it samples new completions and assigns them the oracle-derived reward

$$r_x(y) = \alpha(x)\,\mathbf{1}[\mathcal{A}(y) \equiv \hat{a}(x)]$$

where $\alpha(x)$ is a reliability weight. Because GRPO computes relative advantages within each newly sampled group, groups whose members receive identical rewards provide no within-group ranking signal under the standard formulation. Thus, unlike DPO, GRPO does not directly contrast the oracle-selected and competing debate traces during optimization.

I Ablation details

I.1 MBPP evaluation

Official train split of 374 problems; 200 test problems subsampled from the official test set. Evaluation pass rate uses both the public and challenge unit-tests of the test split. The training oracle does not receive those tests during generation; it clusters parseable programs and scores fence-span confidence. Final evaluation used both public and challenge unit-tests of test-split dataset for final answer accuracy.

I.2 Oracle-Protocol Runs

Other than the results shared in Table 3, we also evaluated two aggregation rules for the answer-span confidence oracle: confidence *sum* and confidence *mean*. All runs use Qwen-2.5-1.5B-it on a GSM8K

splits. The committee uses $N{=}5$ agents and $M{=}2$ debate rounds, followed by one DPO post-training iteration. Table 14 reports accuracy after the initial response, after the exchange round, and after aggregation by the committee.

Table 14: Oracle-protocol ablation on GSM8K using Qwen-2.5-1.5B-it, $N{=}5$ agents, and $M{=}2$ debate rounds. "Initial" denotes accuracy (%) before debate, "Exchange" denotes accuracy (%) after exchange, and "Consensus" denotes the accuracy (%) of the aggregated committee answer.

Aggregation	Method	Initial	Exchange	Consensus
	Ensemble	24.2	-	48.6
Sum	Debate	24.0	44.5	66.0
	DPO	35.4	59.7	71.6
Mean	Debate	23.8	44.0	63.2
	DPO	38.9	60.6	67.6

For each prompt x, the oracle parses the agents' final answers, clusters them by task-level equivalence, and assigns each cluster $\mathcal{C}_k$ a confidence mass

$$w_k = \sum_{i \in \mathcal{C}_k} c_i,$$

where c_i is the answer-span confidence of agent i. The selected cluster is

$$k^\star = \arg\max_k (w_k, n_k), \qquad n_k = |\mathcal{C}_k|,$$

where n_k provides a deterministic tie-break based on cluster size.

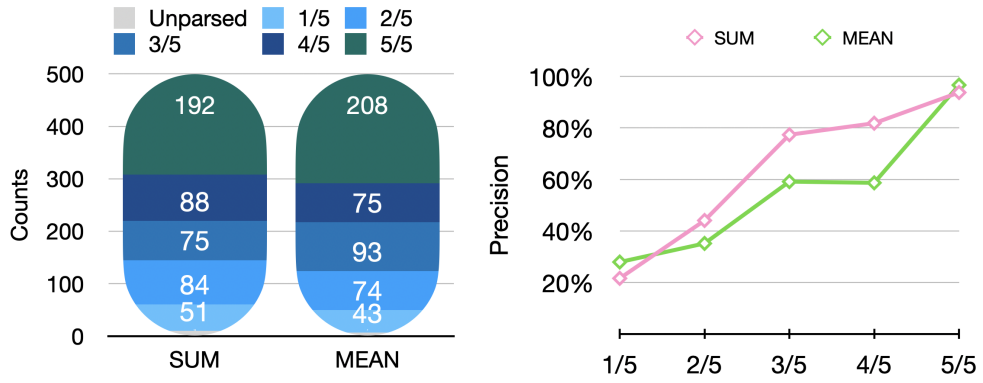
Why we use confidence sum in Sec 4. We use confidence sum rather than mean because a cluster represents the committee's accumulated evidence for an answer, not the confidence of a typical individual member. The sum therefore acts as a confidence-weighted vote:

$$w_k = \sum_{i \in \mathcal{C}_k} c_i.$$

It allows several moderately confident agents to outweigh a single highly confident outlier, while still allowing two highly confident agents to outweigh several weakly confident agents. By contrast, mean confidence normalizes away cluster size. For example, a singleton with confidence 0.95 would defeat a four-agent cluster whose members each have confidence 0.80, because $0.95 > 0.80$, despite the latter cluster containing substantially more independent agreement. Mean confidence can therefore treat a singleton as equivalent to a crowd and make cluster size contribute only indirectly through the tie-break. It is also sensitive to miscalibration: an overconfident incorrect answer can dominate under mean aggregation, whereas sum aggregation requires its confidence to outweigh the accumulated confidence of the competing cluster. Figure 8 visualizes the corresponding accuracy trends for the two aggregation rules. Table 14 alongside motivate the use of confidence mass as the default aggregation rule in AbCD. Both the debate and post-trained checkpoint consensus accuracy are better in aggregation by sum compared to mean.

I.3 Majority Voting vs. Span-Confidence Selection

We compare majority voting (MV) with span-confidence selection on identical debate transcripts before any post-training. For each model–dataset pair, both rules operate on the same sampled responses and answer-equivalence clusters; gold labels are used only to evaluate the selected answer. *Span* denotes the accuracy of the cluster selected by aggregate claimed-answer-span confidence, while *MV* denotes the accuracy of the cluster selected by majority vote. *Disagree* is the fraction of examples for which the two rules select different clusters. For disagreement cases, we report the accuracy of the cluster selected by each rule. We additionally report the corresponding accuracy on count-tie cases.



(a) Agreement-rate count

(b) Correctness per agreement-rate

Figure 8: Comparison of confidence-sum and confidence-mean aggregation for answer-equivalence clusters on GSM8K with Qwen-2.5-1.5B-it and five agents. The agreement rate is the fraction of parseable agent answers assigned to the selected cluster. **Left:** frequency with which each aggregation rule selects an answer at each agreement rate. **Right:** correctness of the selected answer within each agreement-rate bin, measured against GSM8K reference answers for evaluation only. Confidence-sum aggregation retains the total confidence mass contributed by the committee, whereas confidence-mean aggregation normalizes away cluster size. Across agreement-rate bins, the sum rule selects more accurate oracle answers than the mean rule.

Table 15: Comparison of majority voting (MV) and aggregate claimed-answer-span confidence on identical transcripts before post-training. *Span* and *MV* report overall selection accuracy (%). *Disagree* is the percentage of examples for which the two rules select different answer-equivalence clusters. *Disagree: Span/MV* reports the accuracy of the respective selected cluster on disagreement cases. *Ties* is the percentage of count-tie cases, with *Ties: Span/MV* reporting the corresponding selection accuracy. Gold labels are used only for evaluation.

Model	Dataset	Span	MV	Disagree	Disagree: Span/MV	Ties: Span/MV
Llama-3.2-1B-it	GSM8K	15.6	14.6	3.2	31.2 / 0.0	24.1 / 6.9
	SVAMP	29.3	28.7	3.3	40.0 / 20.0	46.2 / 30.8
	MathQA	5.0	5.2	0.6	0.0 / 33.3	0.0 / 25.0
Qwen-2.5-1.5B-it	GSM8K	66.0	65.0	5.8	31.0 / 13.8	31.2 / 23.4
	SVAMP	79.7	79.7	3.0	22.2 / 22.2	33.3 / 33.3
	MathQA	45.2	42.2	8.8	47.7 / 13.6	43.8 / 25.0
Phi-4-mini-3.8B-it	GSM8K	87.6	88.4	1.2	33.3 / 0.0	33.3 / 11.1
	SVAMP	89.7	89.7	0.7	0.0 / 0.0	20.0 / 20.0
	MathQA	41.6	42.6	1.8	22.2 / 77.8	27.3 / 63.6
Gemma-3-4B-it	GSM8K	60.6	60.6	0.0	0.0 / 0.0	50.0 / 50.0
	SVAMP	75.7	76.0	0.3	0.0 / 100.0	0.0 / 100.0
	MathQA	16.6	16.6	0.0	0.0 / 0.0	0.0 / 0.0
Average	GSM8K	57.5	57.2	2.6	31.4 / 7.8	29.8 / 18.3
	SVAMP	68.6	68.5	1.8	27.3 / 22.7	32.5 / 35.0
	MathQA	27.1	26.7	2.8	41.1 / 25.0	39.2 / 30.4

On GSM8K, the two rules disagree on only 2.6% of examples on average across the four models, and their overall selection accuracies are similar (57.5% for span confidence and 57.2% for MV). The difference is larger on cases where the rules disagree: the span-selected cluster is correct on 31.4% of such cases, compared with 7.8% for MV. Span confidence also achieves higher accuracy on count-tie cases (29.8% vs. 18.3%). Across SVAMP and MathQA, the same analysis shows that the relative advantage is dataset- and model-dependent. Overall, Span confidence is most useful precisely when majority voting is ambiguous or the two rules disagree.

Table 16: Accuracy (%) of a heterogeneous committee on GSM8K. The committee consists of L1 (Llama-3.2-1B-it), Q1 (Qwen-2.5-1.5B-it), Q2 (Qwen3-1.7B-it), P4 (Phi-4-mini-3.8B-it), and G4 (Gemma-3-4B-it). "Consensus" denotes accuracy of aggregate solution. "Avg. Agent" denotes mean per-agent performance. The Δ row reports the per-agent change from Ensemble to DPO.

Method	**Accuracy**						
	Consensus	**Avg. Agent**	**L1**	**Q1**	**Q2**	**P4**	**G4**
Ensemble	78.2	30.9	3.6	26.0	1.4	76.8	46.8
Debate	79.2	36.8	16.6	57.6	3.2	71.6	35.2
SFT	79.8	37.4	17.8	54.2	4.2	75.6	35.4
DPO	**87.0**	**68.2**	**71.0**	**75.8**	**41.8**	81.2	**71.4**
KTO	86.6	59.2	52.0	71.6	22.6	**83.6**	66.0
GRPO	83.8	48.5	28.6	70.4	7.2	79.8	56.4
Δ (DPO $-$ Ensemble)	8.8	37.3	67.4	49.8	40.4	4.4	24.6

I.4 Heterogeneous Committee

To examine whether debate benefits from architectural and scale diversity, we form a heterogeneous committee from five instruction-tuned models: L1 (Llama-3.2-1B-it), Q1 (Qwen-2.5-1.5B-it), Q2 (Qwen3-1.7B-it), P4 (Phi-4-mini-3.8B-it), and G4 (Gemma-3-4B-it). All agents use temperature 0.5, and the committee debates for $M{=}2$ rounds with $N{=}5$ agents on GSM8K. The ensemble, debate, and post-training results are reported in Table 16. DPO updates all five models using the shared preferred and dispreferred sets $\mathcal{G}^+(x)$ and $\mathcal{G}^-(x)$ constructed from answer-span confidence. In addition to consensus accuracy, we report the average accuracy of individual agents before and after debate and after one iteration with SFT, DPO, KTO and GRPO. The smaller models (L1, Q1, and Q2) generally benefit more from interaction with the larger committee members, whereas the gains for the larger models (P4 and G4) are less. This pattern suggests that heterogeneous debate can transfer useful information to weaker agents, although it does not uniformly improve every member.

J Limitations of the Experimental Protocol

Our evaluation has several limitations that bound the conclusions supported by the experiments. Most main-grid cells use two post-training runs, while oracle and weighting ablations use three seeds where explicitly reported; these repetitions provide limited evidence about variance and statistical significance. We evaluate on subsets of the available training and test splits rather than the full datasets, limiting the precision with which the reported improvements characterize performance on the complete benchmarks. Post-training is restricted to a single iteration from each instruction-tuned checkpoint with QLoRA, so the experiments do not establish whether the observed gains persist under longer or repeated self-training. The maximum generation length of 256 tokens can truncate reasoning or final-answer spans, with truncation effects particularly visible for Llama-3.2-1B-it on GSM8K. Although the oracle uses no gold labels for supervision, it requires task-specific answer parsing and an equivalence relation, making the approach less directly applicable to outputs for which correctness cannot be parsed or reliably compared. The reported code-generation extension uses execution-based equivalence and therefore does not establish applicability to arbitrary open-ended generation without an executable correctness criterion. Answer-span confidence is computed from token probabilities and is not calibrated against correctness; the method therefore assumes that these probabilities provide a sufficiently informative within-committee ranking signal. Confidence scores are not recalibrated across heterogeneous architectures, and the heterogeneous-committee results show that raw token probabilities need not be directly comparable across models. The confidence oracle is evaluated against gold answers only retrospectively to measure selection quality; these labels are never available during oracle construction or post-training, so the experiments establish empirical utility rather than a guarantee that the oracle identifies the correct cluster. Finally, the experiments cover five instruction-tuned model families and four primary datasets.