

000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 OVERCONFIDENCE IN LLM-AS-A-JUDGE: DIAGNOSIS AND CONFIDENCE-DRIVEN SOLUTION

Anonymous authors

Paper under double-blind review

ABSTRACT

Large Language Models (LLMs) are widely used as automated judges, where practical value depends on both accuracy and trustworthy, risk-aware judgments. Existing approaches predominantly focus on accuracy, overlooking the necessity of well-calibrated confidence, which is vital for adaptive and reliable evaluation pipelines. In this work, we advocate a shift from accuracy-centric evaluation to confidence-driven, risk-aware LLM-as-a-Judge systems, emphasizing the necessity of well-calibrated confidence for trustworthy and adaptive evaluation. We systematically identify the **Overconfidence Phenomenon** in current LLM-as-a-Judges, where predicted confidence significantly overstates actual correctness, undermining reliability in practical deployment. To quantify this phenomenon, we introduce **TH-Score**, a novel metric measuring confidence-accuracy alignment. Furthermore, we propose **LLM-as-a-Fuser**, an ensemble framework that transforms LLMs into reliable, risk-aware evaluators. Extensive experiments demonstrate that our approach substantially improves calibration and enables adaptive, confidence-driven evaluation pipelines, achieving superior reliability and accuracy compared to existing baselines.

1 INTRODUCTION

The widespread adoption of large language models (LLMs) as automated judges—termed the LLM-as-a-Judge paradigm—has revolutionized the evaluation of AI-generated content by offering scalability and efficiency over traditional human annotation (Zheng et al., 2023). In this paradigm, LLMs act as evaluators, with one common application being pairwise comparisons where the model decides which of two text segments is better based on criteria like quality, relevance, or coherence. However, the practical value of these systems depends not only on accuracy but also on trustworthy, risk-aware judgments that can adapt to real-world deployment scenarios. Existing approaches, such as FairEval (Wang et al., 2023a) and JudgeBench (Tan et al., 2024), predominantly emphasize accuracy, often overlooking the critical role of well-calibrated confidence. This calibration, defined as the alignment between a model’s predicted confidence and its actual correctness, is essential for building adaptive evaluation pipelines. For instance, well-calibrated confidence allows high-confidence outputs to be automatically accepted, minimizing manual intervention, while low-confidence cases can be flagged for human review (Li et al., 2024). In this work, we advocate a fundamental shift from accuracy-centric evaluations to confidence-driven, risk-aware LLM-as-a-Judge systems, prioritizing calibration to ensure reliable and trustworthy assessments.

Despite these potential benefits, current LLM-as-a-Judge systems suffer from a pervasive Overconfidence Phenomenon, where predicted confidence levels significantly overstate actual correctness (Mielke et al., 2022; Zhou et al., 2023), thereby undermining reliability in practical applications. Through systematic analysis, we observe that state-of-the-art LLMs exhibit this issue prominently, leading to inflated confidence scores that do not reflect true performance (Zhao et al., 2021). This misalignment results in substantial risks: overconfident models may propagate erroneous judgments without detection, eroding the efficiency gains of automated evaluation, while also complicating downstream decision-making in pipelines (Gu et al., 2024). Furthermore, existing benchmarks and metrics exacerbate the problem by focusing on aggregate accuracy without addressing confidence alignment, introducing biases such as response length or model familiarity that distort calibration assessments (Chen et al., 2024; Zheng et al., 2023; Wang et al., 2023a). Consequently, the lack

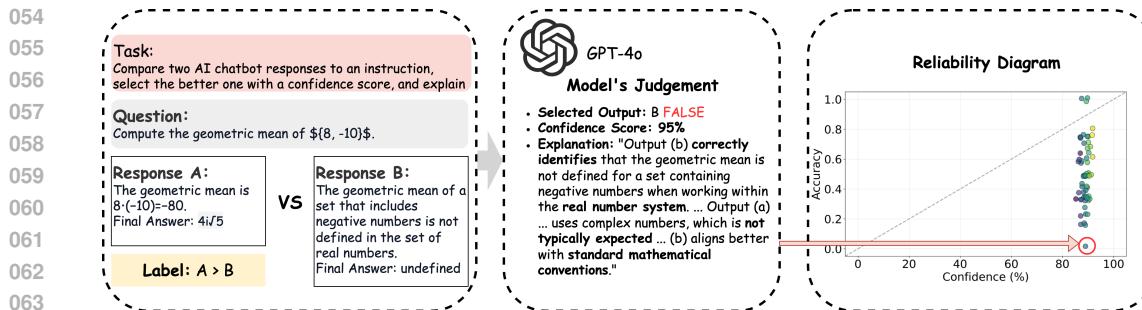


Figure 1: Overconfidence phenomenon of GPT-4o as a Judge on JudgeBench, illustrated with a specific example (content omitted for demonstration purposes) and a Reliability Diagram. The Reliability Diagram plots the model’s confidence scores against actual accuracy, revealing calibration gaps where overconfidence occurs.

of calibration-aware tools limits the deployment of LLMs as dependable evaluators in high-stakes environments.

To address these challenges, we introduce TH-Score, a novel metric that quantifies confidence-accuracy alignment by focusing on critical high- and low-confidence intervals, where practical decisions hinge. Unlike traditional metrics like accuracy or Expected Calibration Error (ECE)—which ignore confidence or overlook key thresholds—TH-Score balances accuracy within these intervals against their coverage, rewarding aligned successes (e.g., high-confidence correct predictions) while penalizing mismatches like overconfident errors. This makes TH-Score a principled tool for detecting the Overconfidence Phenomenon under LLM-as-a-Judge scenario, highlighting cases where high confidence fails to match actual correctness.

Furthermore, we propose LLM-as-a-Fuser, an ensemble framework that leverages a dedicated “fuser” LLM to synthesize judgments and critiques from multiple models, transforming LLMs into reliable, risk-aware evaluators. By integrating diverse perspectives, LLM-as-a-Fuser significantly enhances calibration. Extensive experiments on a widely-used benchmark demonstrate that our approach achieves superior calibration, reliability, and overall accuracy compared to existing baselines, paving the way for more trustworthy LLM-as-a-Judge systems in practical settings.

In a nutshell, our contributions are threefold:

- **Overconfidence Phenomenon:** We identify and characterize the overconfidence in LLM-as-a-Judge, where confidence overstates correctness, limiting risk-aware evaluation.
- **Metric Innovation:** We introduce TH-Score, a novel metric quantifying confidence-accuracy alignment over specified intervals for trustworthy LLM-as-a-Judge.
- **Framework Advancement:** We propose LLM-as-a-Fuser, an ensemble method boosting calibration for adaptive, confidence-driven pipelines with higher reliability and accuracy.

2 RELATED WORK

2.1 LLM-AS-A-JUDGE

LLMs are increasingly used as automated evaluators for text quality. Zheng et al. (2023) showed GPT-4 aligns with human judgments in pairwise comparisons, but proprietary APIs limit reproducibility. PandaLM (Wang et al., 2023b) introduced a 7B-parameter local evaluator with 94% agreement with ChatGPT, supporting offline use. JudgeLM (Zhu et al., 2025) and Agent-as-a-Judge (Zhuge et al., 2024) use modular frameworks with memory and planning, cutting DevAI evaluation costs by 97%. However, alignment between model confidence and accuracy is often ignored, causing inconsistent judgments. Meta’s self-rewarding models (Wu et al., 2024) generate and evaluate outputs iteratively, but calibration needs further study.

As LLM-based judges gain traction for evaluating and enhancing LLMs, various benchmarks have emerged to gauge their effectiveness. Prior works like LLMEval (Lin & Chen, 2023), MTBench,

108 and FairEval primarily assess how well LLM-based judges align with subjective human preferences,
 109 often emphasizing stylistic differences over factual and logical accuracy. Similarly, LLMBBar (Zeng
 110 et al., 2024) evaluates judges based on their ability to follow instructions, using response pairs
 111 with explicit ground truth labels tied to instruction adherence. In contrast, JudgeBench offers a
 112 novel benchmark specifically designed to test LLM-based judges’ reasoning capabilities. It features
 113 350 carefully curated challenging response pairs across knowledge, reasoning, math, and coding
 114 domains, each containing one objectively correct and one subtly incorrect response, prioritizing
 115 factual and logical correctness over subjective or stylistic factors.

116 2.2 CALIBRATION IN LLMs

117 Accurate calibration, aligning a model’s confidence with its accuracy, is crucial for reliable LLM
 118 applications. Traditional methods like temperature scaling (Guo et al., 2017) adjust confidence with
 119 a single scalar but are less effective for large models, while Bayesian methods are computationally
 120 infeasible. Recent approaches, such as the Thermometer method (Shen et al., 2024), train auxiliary
 121 models for recalibration, achieving top uncertainty quantification across 12 benchmarks, and
 122 SPACE (Yi et al., 2024) uses lightweight linear layers for dynamic confidence adjustment. However,
 123 most techniques focus on single models, missing multi-model aggregation benefits, and Collaborative
 124 Calibration (Yang et al., 2024) reduces overconfidence via multi-agent deliberation but requires
 125 significant resources. Current research lacks focus on calibration’s impact on downstream tasks like
 126 data generation, where confidence filtering affects output quality, warranting further exploration.

127 2.3 UNCERTAINTY QUANTIFICATION AND REWARD MODELING

128 Uncertainty-aware frameworks bridge calibration and practical applications. Generating with Con-
 129 fidence (Lin et al., 2023) combines Monte Carlo dropout and response length analysis to filter
 130 low-confidence outputs, demonstrating that well-calibrated models yield higher-quality synthetic
 131 data. Inference-Time Scaling (Liu et al., 2025) dynamically aligns reward models with human
 132 preferences, indirectly improving calibration through gradient-free optimization. However, these
 133 approaches often assume static datasets, failing to address the iterative nature of LLM-as-a-Judge
 134 workflows. Benchmarking LLMs via Uncertainty Quantification (Ye et al., 2024) reveals that cali-
 135 bration degrades under distribution shifts, underscoring the need for adaptive methods.

136 3 OVERCONFIDENCE IN LLM-AS-A-JUDGE

137 In the LLM-as-a-Judge paradigm, models are typically required to select the superior option from
 138 pairwise samples. However, the reliability of model predictions warrants careful examination, par-
 139 ticularly regarding the Overconfidence Phenomenon—a tendency for language models to display
 140 predicted confidence levels that significantly exceed their actual accuracy, resulting in calibration
 141 gaps that undermine reliability. Underconfident models tend to underestimate their own accuracy,
 142 while overconfident ones overestimate their judgment correctness. Such biases introduce noisy sig-
 143 nals that can adversely affect the performance of downstream tasks (e.g., reward modeling). Partic-
 144 ularly in unsupervised or weakly-supervised scenarios, developing a well-calibrated model where
 145 judgment capability aligns with confidence becomes crucial. By acquiring confidence of model
 146 judgments, we can not only filter out low-accuracy predictions but also effectively identify high-
 147 accuracy decisions, thereby enhancing the overall system reliability.

148 3.1 HOW TO MEASURE CONFIDENCE IN LLMs?

149 We employed three methods for calculating confidence: Self-Confidence (SC), Multiple-Prompting
 150 (MP) confidence, and Logprob (LogP) confidence.

151 **SC setting:** We prompt the model to output both the result and its confidence. Model’s temperature
 152 is set to 0 to ensure the reproducibility of the setting.

153 **MP setting:** We adopt a method similar to SimpleQA (Wei et al., 2024), but reduce the number
 154 of requests from 100 to 10 for efficiency, while keeping the temperature at 0.7. The final reply is
 155 determined by majority voting, and the confidence is the count of the chosen response over 10.

162 **LogP Setting.** In this setting, confidence scores are derived from softmax-normalized logits for the
 163 final output tokens (e.g., ‘A’ or ‘B’). For a binary choice task with options A and B , and correspond-
 164 ing logits l_A and l_B , we first compute the softmax probabilities:
 165

$$166 \quad p(A) = \frac{e^{l_A}}{e^{l_A} + e^{l_B}}, \quad p(B) = \frac{e^{l_B}}{e^{l_A} + e^{l_B}}.$$

168 Confidence is then defined as the maximum probability:
 169

$$170 \quad \text{Confidence}_{\text{logp}} = \max(p(A), p(B)).$$

171 Temperature is set to 0 to ensure deterministic outputs.
 172

173 **Question:**

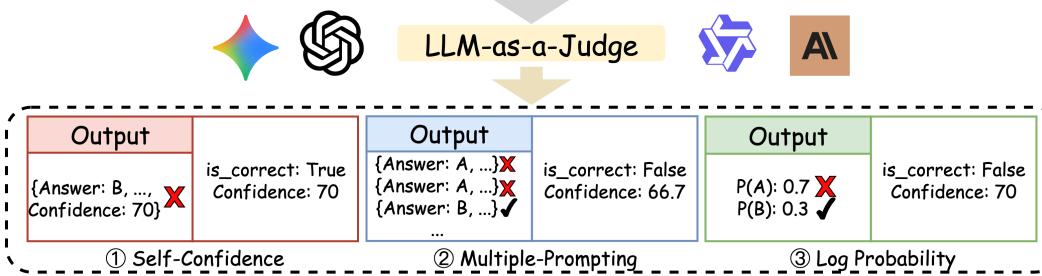
174 The complementary color of blue is\n(A) green\n(B) black\n(C) white\n(D) purple\n(E) red\n(F) orange\n(G) magenta\n(H)
 175 yellow\n(I) brown\n(J) cyan\nIf you cannot determine the correct multiple-choice answer, take your best guess. Once you have your
 176 answer, please duplicate that letter five times in a single string. For example, if the answer is K, then write KKKKK.\nLet's think step
 177 by step.

178 **Text A:**

179 To determine the complementary color of blue, we need to
 180 understand the concept of color theory, specifically the color
 181 wheel Let's duplicate the letter corresponding to the correct
 182 answer:\nFFFFFF

183 **Text B:**

184 To find the complementary color of blue, we need to use color
 185 theory, specifically the color wheel Following the instructions
 186 to duplicate the letter five times, the final answer is:\nHHHHH



190 Figure 2: Visualization of the three confidence calculation settings: Self-Confidence (SC), Multiple-
 191 Prompting (MP), and Logprob (Logp), using data with ID 122 from JudgeBench as an example.
 192

193 **3.2 EXISTING CALIBRATION EVALUATION METRICS**

195 To conduct an objective and comprehensive evaluation of the calibration of LLM-as-a-Judge, we
 196 applied five existing metrics—Expected Calibration Error (ECE), Adaptive Calibration Error (ACE),
 197 Maximum Calibration Error (MCE), Brier Score, and Negative Log Likelihood (NLL)—to the three
 198 confidence calculation methods described earlier in this section. Table 3 provides a brief introduction
 199 to the calculation methods and characteristics of these metrics.
 200

201 **3.3 INITIAL RESULTS**

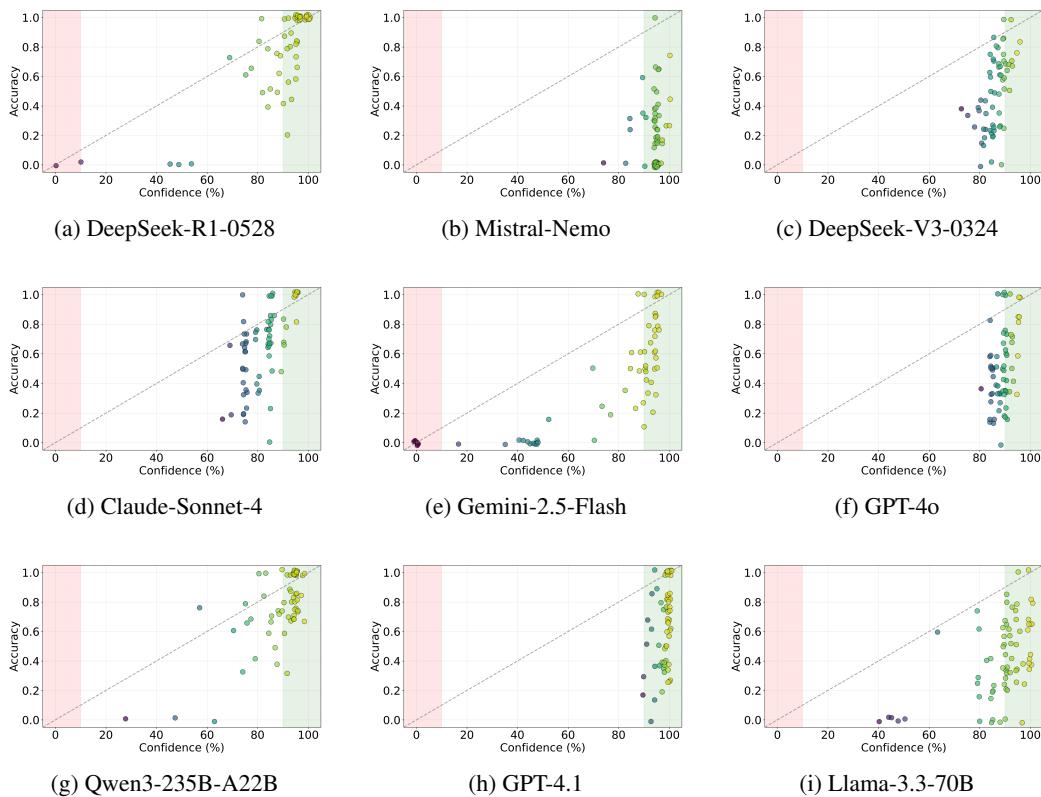
203 We systematically evaluate 14 cutting-edge models on the JudgeBench benchmark, with com-
 204 plete results presented in Table 1 and 2. These include open-source models such as Qwen3-235B-
 205 A22B (Qwen Team, 2025), DeepSeek-R1-0528 (DeepSeek-AI et al., 2025), R1-Distill-Qwen, R1-
 206 Distill-Llama, DeepSeek-V3-0324 (DeepSeek-AI et al., 2024), Llama-3.3-70B (Dubey et al., 2024),
 207 and Mistral-Nemo (Team, 2024), as well as proprietary models like OpenAI-o3-mini (OpenAI,
 208 2025b), Claude-Sonnet-4 (Anthropic, 2025), GPT-4.1 (OpenAI, 2025a), GPT-4.1-mini, Gemini-2.5-
 209 Flash (DeepMind, 2025), GPT-4o (Ahmad et al., 2024), and GPT-4.1-nano. Our analysis focuses
 210 on the impact of model scales on accuracy and confidence calibration (ECE/ACE), visually further
 211 illustrated by reliability plots in Figure 3 showcasing calibration gaps in high-confidence (red) and
 212 low-confidence (green) regions for selected models.
 213

214 **3.4 EMPIRICAL OBSERVATIONS OF OVERCONFIDENCE**

215 Figure 3 reveals significant calibration gaps across the evaluated models, with most exhibiting the
 216 Overconfidence Phenomenon in high-confidence regions (green). This pattern undermines the reli-
 217

216
 217 Table 1: Performance metrics of various models under the Self-Confidence (SC) setting, categorized
 218 into Open Source and Proprietary models. Arrows indicate optimization direction: \uparrow higher is better,
 219 \downarrow lower is better. Best results for each metric are bolded.

Model	Acc \uparrow	ECE \downarrow	ACE \downarrow	Brier Score \downarrow	MCE \downarrow	NLL \downarrow	TH \uparrow
Open Source Models							
Qwen3-235B-A22B	77.43	11.78	12.16	0.16	63.50	0.52	17.52
DeepSeek-R1-0528	76.86	12.07	11.39	0.13	40.00	0.42	14.59
R1-Distill-Qwen	65.71	27.26	27.10	0.29	69.00	0.91	8.16
R1-Distill-Llama	59.71	31.02	30.89	0.31	65.00	1.31	7.01
DeepSeek-V3-0324	49.71	36.21	36.35	0.37	50.24	1.03	2.46
Llama-3.3-70B	42.00	47.37	46.78	0.45	63.78	2.75	0.80
Mistral-Nemo	20.29	74.22	74.21	0.71	80.00	3.01	-11.64
Proprietary Models							
OpenAI-o3-mini	74.29	15.97	17.20	0.20	60.00	0.62	12.83
Claude-Sonnet-4	64.29	17.98	18.00	0.24	45.00	0.69	9.89
GPT-4.1	63.14	26.39	26.86	0.29	55.00	0.85	7.55
GPT-4.1-mini	55.71	32.70	32.79	0.35	44.21	1.00	3.29
Gemini-2.5-Flash	39.43	30.49	30.41	0.26	56.11	0.78	2.71
GPT-4o	49.71	39.25	39.28	0.40	57.50	1.15	1.57
GPT-4.1-nano	26.86	57.03	57.08	0.52	72.50	1.38	-0.07



261
 262 Figure 3: Illustration of calibration gaps in low-confidence regions (red) and high-confidence regions
 263 (green) where models show significant accuracy-confidence discrepancy.

264
 265 ability of the LLM-as-a-Judge, as models like DeepSeek-R1-0528 and GPT-4o cluster predictions at
 266 high confidence levels (90-100%) but achieve accuracies well below the ideal calibration line.

267
 268 This overconfidence impacts downstream tasks, such as data filtering, by retaining flawed outputs
 269 (false positives) or discarding valuable ones (false negatives), thereby degrading overall perfor-
 270 mance. For instance, high ECE values in GPT-4o (39.25 in SC, 47.09 in MP, 45.05 in LogP),

Mistral-Nemo (74.22 in SC, 68.89 in MP, 64.63 in LogP), and GPT-4.1-nano (57.03 in SC, 67.43 in MP, 66.05 in LogP) necessitate increased human oversight to mitigate risks, diminishing the efficiency of automated judging processes (see Table 1 and Appendix A.5).

Table 2: Performance comparison of different LLMs under logP setting.

Model	Acc \uparrow	ECE \downarrow	ACE \downarrow	Brier Score \downarrow	MCE \downarrow	NLL \downarrow	TH Score \uparrow
DeepSeek-R1-0528	78.29	6.84	6.62	0.1298	46.15	0.4211	2.96
GPT-4.1	63.43	34.46	34.43	0.3462	63.80	1.7287	7.36
GPT-4.1-mini	55.14	42.56	42.53	0.4253	58.32	1.7946	2.56
GPT-4o	50.86	45.05	45.06	0.4493	61.19	1.6238	0.79
DeepSeek-V3-0324	48.29	50.76	50.68	0.5044	50.76	2.4714	-0.85
Llama-3.3-70B	43.43	53.53	53.55	0.5318	54.04	2.3400	-3.25
GPT-4.1-nano	28.00	66.05	66.16	0.6349	82.98	2.1206	-8.70
Mistral-Nemo	23.43	64.63	64.60	0.6051	79.59	1.7887	-5.89

4 TH-SCORE: A NEW METRIC FOR LLM-AS-A-JUDGE CALIBRATION EVALUATION

While existing calibration metrics such as ECE, Brier Score and NLL offer valuable insights into model reliability, they often overlook practical aspects like high-confidence regions essential for real-world applications in LLM-as-a-Judge scenarios. To address these limitations and better align confidence with accuracy in targeted intervals, we introduce TH-Score, a novel metric designed to improve evaluation for data filtering and quality assessment tasks.

4.1 DEFINITION

The TH-Score focuses on two key confidence intervals relevant to practical applications:

- **High-Confidence Data** ($100 - \epsilon, 100$): These predictions are generally considered highly reliable, and selecting them can significantly enhance the overall dataset quality. ϵ is a hyperparameter defining the high-confidence threshold.
- **Low-Confidence Data** ($0, \epsilon$): These predictions are inherently uncertain, and discarding them can effectively reduce noise and enhance overall data quality. ϵ is also a crucial hyperparameter that determines a threshold for what constitutes low confidence.

This metric quantifies model performance by jointly considering the accuracy of predictions within specified confidence intervals and the coverage of these intervals, facilitating effective data filtering and quality evaluation. The TH-Score is formally defined as:

$$\text{TH-Score} = (e^{(\text{accuracy}-0.5)} - 1) \times \text{percentage},$$

where e denotes the base of the natural logarithm, serving as a scaling hyperparameter; *accuracy* represents the prediction accuracy specifically for samples falling within the target confidence intervals; *percentage* indicates the proportion of total samples that fall within these intervals.

This formulation ensures that the TH-Score increases with both higher accuracy and a larger proportion of high-confidence or low-confidence data, providing a balanced measure of model reliability in practical usage scenarios.

4.2 IMPACT OF ϵ ON TH-SCORE PERFORMANCE

Table 4 presents the TH-Score results for various models under different values of ϵ . The table also includes accuracy rates within specified intervals and the proportion of interval data relative to the total dataset. When $\epsilon = 0.05$, most models, except the most powerful ones, exhibit limited calibration capability. Most models either have minimal data within this interval or demonstrate significantly reduced accuracy, highlighting the stringent calibration demands of such a small ϵ and underscoring the challenges in achieving reliable confidence alignment at fine-grained thresholds. At $\epsilon = 0.1$,

the value used in our primary experiments, most models align well with this calibration threshold, resulting in strong discriminative power. With the exception of weaker models like Mistral-Nemo, the majority of models have substantial data within this interval, enabling effective comparison of their calibration performance. This observation suggests that an effective approach for selecting ϵ is to choose a value where most models contribute significant data to the interval.

However, when ϵ is increased to 0.15, while data coverage improves, the discriminative power diminishes. The advantages of high-performing models, such as DeepSeek-R1-0528, become less pronounced due to the relaxed performance requirements associated with a larger ϵ . Thus, selecting an appropriate ϵ requires balancing data coverage with discriminative power, avoiding excessively large values that dilute model differentiation.

Table 3: A comparison of calibration metrics. Our proposed TH-Score is designed to evaluate practical reliability in high-confidence regions, addressing the limitations of standard approaches. Notation: $\%$ = interval coverage; ϵ = adjustable threshold (default=0.1); acc = accuracy within ϵ ranges; o_i = ground truth; p_i = predicted probability.

Metric	Formula	Key Characteristics
ECE	$\sum_{i=1}^M \frac{n_i}{N} \text{acc}(i) - \text{conf}(i) $	✗ Fixed-width bins. ✗ Ignores high-confidence regions.
ACE	Variant of ECE with adaptive binning	✗ Computationally expensive. ✗ Lacks focus on confidence intervals.
Brier Score	$\frac{1}{N} \sum_{i=1}^N (p_i - o_i)^2$	✗ Less interpretable. ✗ Fails to isolate miscalibration.
MCE	$\max_{i \in \{1, \dots, M\}} \text{acc}(i) - \text{conf}(i) $	✗ Sensitive to outliers. ✗ Not reflective of overall calibration.
NLL	$-\frac{1}{N} \sum [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)]$	✗ Unbounded, hard to compare. ✗ Sensitive to overconfident errors.
TH-Score	$(e^{(\text{acc}-0.5)} - 1) \times \%$	✓ Targets high-confidence regions. ✓ Uses adaptive threshold ϵ . ✓ Balances accuracy and coverage. ✓ Interpretable, bounded score.

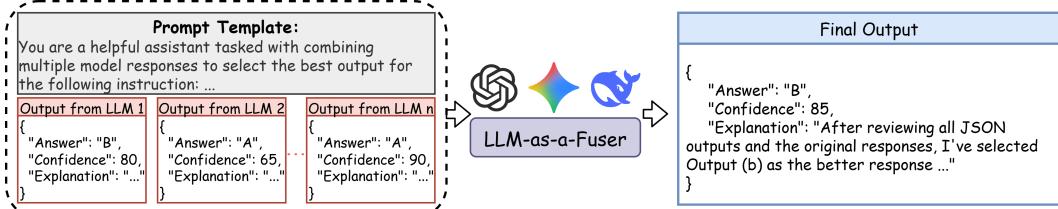
Table 4: Model performance under different ϵ values in SC setting.

Model	$\epsilon = 0.05$			$\epsilon = 0.10$			$\epsilon = 0.15$		
	Acc \uparrow	% \uparrow	TH \uparrow	Acc \uparrow	% \uparrow	TH \uparrow	Acc \uparrow	% \uparrow	TH \uparrow
DeepSeek-R1-0528	1.00	37.4	12.14	0.91	68.6	17.52	0.87	81.1	18.38
Qwen-235B-A22B	0.88	8.0	1.93	0.88	63.4	14.59	0.84	78.0	15.73
GPT-4.1	1.00	1.1	0.37	0.83	38.0	7.55	0.69	74.3	7.88
R1-Distill-Llama	0.86	24.9	5.42	0.71	59.4	7.01	0.65	81.4	6.72
GPT-4.1-mini	0.00	0.0	0.00	0.83	13.7	2.71	0.63	68.6	4.57
DeepSeek-V3-0324	1.00	0.6	0.19	0.79	9.4	1.57	0.65	44.3	3.63
GPT-4o	0.00	0.0	0.00	0.71	21.4	2.46	0.54	72.3	1.38
Llama-3.3-70B	0.52	22.0	0.22	0.54	43.1	0.80	0.48	74.3	-0.57
GPT-4.1-nano	0.00	0.0	0.00	0.43	2.0	-0.07	0.46	6.9	-0.14
Mistral-Nemo	0.36	12.9	-0.86	0.20	88.9	-11.64	0.20	94.9	-12.12

5 LLM-AS-A-FUSER

As shown in the section on overconfidence in LLM-as-a-judge, while LLM-as-a-judge offers a promising and increasingly practical approach to evaluating diverse model outputs, its calibration issues—such as overconfidence in unreliable judgments—limit its overall reliability. Traditional aggregation methods (e.g., majority voting) compound this problem by ignoring nuanced critiques from individual models and focusing only on final decisions. To address these limitations, we propose **LLM-as-a-Fuser** framework, which redefines the LLM’s role from a passive judge to an active

378 *fuser*. By synthesizing model decisions and their rationales, the fuser enables evidence-aware aggregation, ultimately improving both calibration and robustness. As shown in Figure 4, the fuser ingests decisions and critiques from an ensemble of models, analyzing reasoning. Unlike traditional 380 methods, this approach grounds the final decision in evidence. 381



382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
Figure 4: Illustration of the LLM-as-a-Fuser framework, where the fuser model aggregates decisions and their corresponding critiques from an ensemble of models.

5.1 BASELINE METHODS

To evaluate the performance of LLM-as-a-Fuser, we compare it against several baseline aggregation methods that combine predictions from multiple models. These methods vary in how they weight or process model predictions and confidences but do not incorporate model critiques, relying solely on final decisions and, where applicable, associated confidence scores. The baseline methods are:

- **Majority Voting:** Selects the most frequent label across models, with equal votes counted. Ties are broken fairly by the highest confidence score.
- **Confidence-Weighted Voting:** Weights votes by model confidence scores, selecting the label with the highest total. Ties use the maximum confidence.
- **Square-Root Confidence-Weighted Voting:** Applies square-root transformation to confidences, summing them to select the label with the highest total.
- **Entropy-Weighted Voting:** Weights model confidences by inverse entropy, ultimately selecting the label with the highest weighted confidence sum.

These baseline methods serve as standard approaches for aggregating model predictions and provide a robust comparison for evaluating the effectiveness of LLM-as-a-Fuser, which leverages model critiques in addition to final decisions. Each method was implemented with careful consideration of model calibration and tie-breaking mechanisms to ensure fair and consistent comparisons.

5.2 EXPERIMENTAL RESULTS

Table 5 presents the performance of LLM-as-a-Fuser and baseline on JudgeBench, compared to individual model results under the Self-Confidence (SC) setting in Table 1. LLM-as-a-Fuser with Qwen3-235B-A22B achieves the highest accuracy (86.29%) and best calibration (ECE of 6.42%), outperforming baselines like Entropy Weighted Voting (81.71% Acc, 8.48% ECE) and showing substantial gains over SC models (e.g., +8.86% Acc and -5.36% ECE relative to its SC counterpart at 77.43% Acc, 11.78% ECE). Notably, models like Mistral-Nemo exhibit dramatic improvements (+47.14% Acc, -53.73% ECE), followed by Gemini-2.5-Flash (+38.57% Acc) and GPT-4.1-nano (+30.85% Acc), indicating weaker SC performers benefit significantly from critique integration in the fuser framework. Baseline aggregation methods also surpass individual SC performances; for instance, Entropy Weighted Voting exceeds the top SC model by 4.28% in accuracy and 3.3% in ECE, while other baselines (80% Acc) outperform most SC models. Other fusers vary, with GPT-4.0 weakest (49.71% Acc, 44.07% ECE). Critique integration drives LLM-as-a-Fuser’s accuracy and calibration, and ensemble methods generally yield better results than isolated SC evaluations.

5.3 DISAGREEMENT WITH MAJORITY VOTING

We analyzed cases where LLM-as-a-Fuser’s decisions diverged from majority voting, as visualized in Figure 5. Qwen3-235B-A22B, the top-performing fuser (Table 5), has the most correct disagreements (34) and few incorrect ones (12), reflecting its effective use of model critiques. In contrast,

432
 433 Table 5: Performance comparison between baseline aggregation methods and LLM-as-a-Fuser mod-
 434
 435
 436
 437
 438
 439
 440
 441
 442
 443
 444
 445
 446
 447
 448
 449
 450
 451
 452
 453
 454
 455
 456
 457
 458
 459
 460
 461
 462
 463
 464
 465
 466
 467
 468
 469
 470
 471
 472
 473
 474
 475
 476
 477
 478
 479
 480
 481
 482
 483
 484
 485
 486
 487
 488
 489
 490
 491
 492
 493
 494
 495
 496
 497
 498
 499
 500
 501
 502
 503
 504
 505
 506
 507
 508
 509
 510
 511
 512
 513
 514
 515
 516
 517
 518
 519
 520
 521
 522
 523
 524
 525
 526
 527
 528
 529
 530
 531
 532
 533
 534
 535
 536
 537
 538
 539
 540
 541
 542
 543
 544
 545
 546
 547
 548
 549
 550
 551
 552
 553
 554
 555
 556
 557
 558
 559
 560
 561
 562
 563
 564
 565
 566
 567
 568
 569
 570
 571
 572
 573
 574
 575
 576
 577
 578
 579
 580
 581
 582
 583
 584
 585
 586
 587
 588
 589
 590
 591
 592
 593
 594
 595
 596
 597
 598
 599
 600
 601
 602
 603
 604
 605
 606
 607
 608
 609
 610
 611
 612
 613
 614
 615
 616
 617
 618
 619
 620
 621
 622
 623
 624
 625
 626
 627
 628
 629
 630
 631
 632
 633
 634
 635
 636
 637
 638
 639
 640
 641
 642
 643
 644
 645
 646
 647
 648
 649
 650
 651
 652
 653
 654
 655
 656
 657
 658
 659
 660
 661
 662
 663
 664
 665
 666
 667
 668
 669
 670
 671
 672
 673
 674
 675
 676
 677
 678
 679
 680
 681
 682
 683
 684
 685
 686
 687
 688
 689
 690
 691
 692
 693
 694
 695
 696
 697
 698
 699
 700
 701
 702
 703
 704
 705
 706
 707
 708
 709
 710
 711
 712
 713
 714
 715
 716
 717
 718
 719
 720
 721
 722
 723
 724
 725
 726
 727
 728
 729
 730
 731
 732
 733
 734
 735
 736
 737
 738
 739
 740
 741
 742
 743
 744
 745
 746
 747
 748
 749
 750
 751
 752
 753
 754
 755
 756
 757
 758
 759
 760
 761
 762
 763
 764
 765
 766
 767
 768
 769
 770
 771
 772
 773
 774
 775
 776
 777
 778
 779
 780
 781
 782
 783
 784
 785
 786
 787
 788
 789
 790
 791
 792
 793
 794
 795
 796
 797
 798
 799
 800
 801
 802
 803
 804
 805
 806
 807
 808
 809
 810
 811
 812
 813
 814
 815
 816
 817
 818
 819
 820
 821
 822
 823
 824
 825
 826
 827
 828
 829
 830
 831
 832
 833
 834
 835
 836
 837
 838
 839
 840
 841
 842
 843
 844
 845
 846
 847
 848
 849
 850
 851
 852
 853
 854
 855
 856
 857
 858
 859
 860
 861
 862
 863
 864
 865
 866
 867
 868
 869
 870
 871
 872
 873
 874
 875
 876
 877
 878
 879
 880
 881
 882
 883
 884
 885
 886
 887
 888
 889
 890
 891
 892
 893
 894
 895
 896
 897
 898
 899
 900
 901
 902
 903
 904
 905
 906
 907
 908
 909
 910
 911
 912
 913
 914
 915
 916
 917
 918
 919
 920
 921
 922
 923
 924
 925
 926
 927
 928
 929
 930
 931
 932
 933
 934
 935
 936
 937
 938
 939
 940
 941
 942
 943
 944
 945
 946
 947
 948
 949
 950
 951
 952
 953
 954
 955
 956
 957
 958
 959
 960
 961
 962
 963
 964
 965
 966
 967
 968
 969
 970
 971
 972
 973
 974
 975
 976
 977
 978
 979
 980
 981
 982
 983
 984
 985
 986
 987
 988
 989
 990
 991
 992
 993
 994
 995
 996
 997
 998
 999
 1000
 1001
 1002
 1003
 1004
 1005
 1006
 1007
 1008
 1009
 1010
 1011
 1012
 1013
 1014
 1015
 1016
 1017
 1018
 1019
 1020
 1021
 1022
 1023
 1024
 1025
 1026
 1027
 1028
 1029
 1030
 1031
 1032
 1033
 1034
 1035
 1036
 1037
 1038
 1039
 1040
 1041
 1042
 1043
 1044
 1045
 1046
 1047
 1048
 1049
 1050
 1051
 1052
 1053
 1054
 1055
 1056
 1057
 1058
 1059
 1060
 1061
 1062
 1063
 1064
 1065
 1066
 1067
 1068
 1069
 1070
 1071
 1072
 1073
 1074
 1075
 1076
 1077
 1078
 1079
 1080
 1081
 1082
 1083
 1084
 1085
 1086
 1087
 1088
 1089
 1090
 1091
 1092
 1093
 1094
 1095
 1096
 1097
 1098
 1099
 1100
 1101
 1102
 1103
 1104
 1105
 1106
 1107
 1108
 1109
 1110
 1111
 1112
 1113
 1114
 1115
 1116
 1117
 1118
 1119
 1120
 1121
 1122
 1123
 1124
 1125
 1126
 1127
 1128
 1129
 1130
 1131
 1132
 1133
 1134
 1135
 1136
 1137
 1138
 1139
 1140
 1141
 1142
 1143
 1144
 1145
 1146
 1147
 1148
 1149
 1150
 1151
 1152
 1153
 1154
 1155
 1156
 1157
 1158
 1159
 1160
 1161
 1162
 1163
 1164
 1165
 1166
 1167
 1168
 1169
 1170
 1171
 1172
 1173
 1174
 1175
 1176
 1177
 1178
 1179
 1180
 1181
 1182
 1183
 1184
 1185
 1186
 1187
 1188
 1189
 1190
 1191
 1192
 1193
 1194
 1195
 1196
 1197
 1198
 1199
 1200
 1201
 1202
 1203
 1204
 1205
 1206
 1207
 1208
 1209
 1210
 1211
 1212
 1213
 1214
 1215
 1216
 1217
 1218
 1219
 1220
 1221
 1222
 1223
 1224
 1225
 1226
 1227
 1228
 1229
 1230
 1231
 1232
 1233
 1234
 1235
 1236
 1237
 1238
 1239
 1240
 1241
 1242
 1243
 1244
 1245
 1246
 1247
 1248
 1249
 1250
 1251
 1252
 1253
 1254
 1255
 1256
 1257
 1258
 1259
 1260
 1261
 1262
 1263
 1264
 1265
 1266
 1267
 1268
 1269
 1270
 1271
 1272
 1273
 1274
 1275
 1276
 1277
 1278
 1279
 1280
 1281
 1282
 1283
 1284
 1285
 1286
 1287
 1288
 1289
 1290
 1291
 1292
 1293
 1294
 1295
 1296
 1297
 1298
 1299
 1300
 1301
 1302
 1303
 1304
 1305
 1306
 1307
 1308
 1309
 1310
 1311
 1312
 1313
 1314
 1315
 1316
 1317
 1318
 1319
 1320
 1321
 1322
 1323
 1324
 1325
 1326
 1327
 1328
 1329
 1330
 1331
 1332
 1333
 1334
 1335
 1336
 1337
 1338
 1339
 1340
 1341
 1342
 1343
 1344
 1345
 1346
 1347
 1348
 1349
 1350
 1351
 1352
 1353
 1354
 1355
 1356
 1357
 1358
 1359
 1360
 1361
 1362
 1363
 1364
 1365
 1366
 1367
 1368
 1369
 1370
 1371
 1372
 1373
 1374
 1375
 1376
 1377
 1378
 1379
 1380
 1381
 1382
 1383
 1384
 1385
 1386
 1387
 1388
 1389
 1390
 1391
 1392
 1393
 1394
 1395
 1396
 1397
 1398
 1399
 1400
 1401
 1402
 1403
 1404
 1405
 1406
 1407
 1408
 1409
 1410
 1411
 1412
 1413
 1414
 1415
 1416
 1417
 1418
 1419
 1420
 1421
 1422
 1423
 1424
 1425
 1426
 1427
 1428
 1429
 1430
 1431
 1432
 1433
 1434
 1435
 1436
 1437
 1438
 1439
 1440
 1441
 1442
 1443
 1444
 1445
 1446
 1447
 1448
 1449
 1450
 1451
 1452
 1453
 1454
 1455
 1456
 1457
 1458
 1459
 1460
 1461
 1462
 1463
 1464
 1465
 1466
 1467
 1468
 1469
 1470
 1471
 1472
 1473
 1474
 1475
 1476
 1477
 1478
 1479
 1480
 1481
 1482
 1483
 1484
 1485
 1486
 1487
 1488
 1489
 1490
 1491
 1492
 1493
 1494
 1495
 1496
 1497
 1498
 1499
 1500
 1501
 1502
 1503
 1504
 1505
 1506
 1507
 1508
 1509
 1510
 1511
 1512
 1513
 1514
 1515
 1516
 1517
 1518
 1519
 1520
 1521
 1522
 1523
 1524
 1525
 1526
 1527
 1528
 1529
 1530
 1531
 1532
 1533
 1534
 1535
 1536
 1537
 1538
 1539
 1540
 1541
 1542
 1543
 1544
 1545
 1546
 1547
 1548
 1549
 1550
 1551
 1552
 1553
 1554
 1555
 1556
 1557
 1558
 1559
 1560
 1561
 1562
 1563
 1564
 1565
 1566
 1567
 1568
 1569
 1570
 1571
 1572
 1573
 1574
 1575
 1576
 1577
 1578
 1579
 1580
 1581
 1582
 1583
 1584
 1585
 1586
 1587
 1588
 1589
 1590
 1591
 1592
 1593
 1594
 1595
 1596
 1597
 1598
 1599
 1600
 1601
 1602
 1603
 1604
 1605
 1606
 1607
 1608
 1609
 1610
 1611
 1612
 1613
 1614
 1615
 1616
 1617
 1618
 1619
 1620
 1621
 1622
 1623
 1624
 1625
 1626
 1627
 1628
 1629
 1630
 1631
 1632
 1633
 1634
 1635
 1636
 1637
 1638
 1639
 1640
 1641
 1642
 1643
 1644
 1645
 1646
 1647
 1648
 1649
 1650
 1651
 1652
 1653
 1654
 1655
 1656
 1657
 1658
 1659
 1660
 1661
 1662
 1663
 1664
 1665
 1666
 1667
 1668
 1669
 1670
 1671
 1672
 1673
 1674
 1675
 1676
 1677
 1678
 1679
 1680
 1681
 1682
 1683
 1684
 1685
 1686
 1687
 1688
 1689
 1690
 1691
 1692
 1693
 1694
 1695
 1696
 1697
 1698
 1699
 1700
 1701
 1702
 1703
 1704
 1705
 1706
 1707
 1708
 1709
 1710
 1711
 1712
 1713
 1714
 1715
 1716
 1717
 1718
 1719
 1720
 1721
 1722
 1723
 1724
 1725
 1726
 1727
 1728
 1729
 1730
 1731
 1732
 1733
 1734
 1735
 1736
 1737
 1738
 1739
 1740
 1741
 1742
 1743
 1744
 1745
 1746
 1747
 1748
 1749
 1750
 1751
 1752
 1753
 1754
 1755
 1756
 1757
 1758
 1759
 1760
 1761
 1762
 1763
 1764
 1765
 1766
 1767
 1768
 1769
 1770
 1771
 1772
 1773
 1774
 1775
 1776
 1777
 1778
 1779
 1780
 1781
 1782
 1783
 1784
 1785
 1786
 1787
 1788
 1789
 1790
 1791
 1792
 1793
 1794
 1795
 1796
 1797
 1798
 1799
 1800
 1801
 1802
 1803
 1804
 1805
 1806
 1807
 1808
 1809
 1810
 1811
 1812
 1813
 1814
 1815
 1816
 1817
 1818
 1819
 1820
 1821
 1822
 1823
 1824
 1825
 1826
 1827
 1828
 1829
 1830
 1831
 1832
 1833
 1834
 1835
 1836
 1837
 1838
 1839
 1840
 1841
 1842
 1843
 1844
 1845
 1846
 1847
 1848
 1849
 1850
 1851
 1852
 1853
 1854
 1855
 1856
 1857
 1858
 1859
 1860
 1861
 1862
 1863
 1864
 1865
 1866
 1867
 1868
 1869
 1870
 1871
 1872
 1873
 1874
 1875
 1876
 1877
 1878
 1879
 1880
 1881
 1882
 1883
 1884
 1885
 1886
 1887
 1888
 1889
 1890
 1891
 1892
 1893
 1894
 1895
 1896
 1897
 1898
 1899
 1900
 1901
 1902
 1903
 1904
 1905
 1906
 1907
 1908
 1909
 1910
 1911
 1912
 1913
 1914
 1915<br

486 REFERENCES
487

488 Lama Ahmad et al. Gpt-4o system card, 2024.

489 Anthropic. Claude sonnet 4, 2025. URL <https://www.anthropic.com/clause/sonnet>.

490 Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. Humans or LLMs
491 as the judge? a study on judgement bias. In *Proceedings of the 2024 Conference on Empirical
492 Methods in Natural Language Processing*, pp. 8301–8327, 2024.

493

494 Google DeepMind. Gemini flash, 2025. URL <https://deepmind.google/models/gemini/flash/>.

495

496 DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Cheng-
497 gang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang,
498 Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting
499 Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui
500 Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J.L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni,
501 Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao
502 Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong
503 Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang,
504 Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang,
505 Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R.J. Chen,
506 R.L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi
507 Chen, S.S. Li, Shanghao Lu, Shangyan Zhou, Shanhua Chen, Shaoqing Wu, Shengfeng Ye,
508 Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting
509 Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W.L. Xiao, Wangding Zeng, and et al. Deepseek-v3
510 technical report, 2024.

511 DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu,
512 Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z.F. Wu,
513 Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao
514 Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan,
515 Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao,
516 Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding,
517 Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang
518 Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J.L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong,
519 Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao,
520 Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang,
521 Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng
522 Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R.J. Chen,
523 R.L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhua Chen, Shengfeng Ye, Shiyu
524 Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S.S. Li, and et al. Deepseek-r1: Incentivizing
525 reasoning capability in llms via reinforcement learning, 2025.

526 Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha
527 Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models.
528 *arXiv e-prints*, pp. arXiv–2407, 2024.

529 Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan
530 Shen, Shengjie Ma, Honghao Liu, et al. A survey on llm-as-a-judge. *arXiv e-prints*, pp. arXiv–
531 2411, 2024.

532 Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural
533 networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

534 Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun
535 Liu. Llms-as-judges: a comprehensive survey on llm-based evaluation methods. *arXiv preprint
536 arXiv:2412.05579*, 2024.

537 Yen-Ting Lin and Yun-Nung Chen. Llm-eval: Unified multi-dimensional automatic evaluation for
538 open-domain conversations with large language models, 2023. URL <https://arxiv.org/abs/2305.13711>.

540 Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with confidence: Uncertainty quantifi-
 541 cation for black-box large language models. *arXiv preprint arXiv:2305.19187*, 2023.

542

543 Zijun Liu, Peiyi Wang, Runxin Xu, Shirong Ma, Chong Ruan, Peng Li, Yang Liu, and Yu Wu.
 544 Inference-time scaling for generalist reward modeling. *arXiv preprint arXiv:2504.02495*, 2025.

545 Sabrina J Mielke, Arthur Szlam, Emily Dinan, and Y-Lan Boureau. Reducing conversational agents'
 546 overconfidence through linguistic calibration. *Transactions of the Association for Computational
 547 Linguistics*, 10:857–872, 2022. doi: 10.1162/tacl_a_00494.

548

549 OpenAI. Introducing gpt-4.1 in the api, 2025a. URL <https://openai.com/index/gpt-4-1/>.

550

551 OpenAI. Introducing openai o3 and o4-mini, 2025b. URL <https://openai.com/index/introducing-o3-and-o4-mini/>.

552

553 Alibaba Qwen Team. Qwen3: Think deeper, act faster. <https://qwenlm.github.io/blog/qwen3/>, 2025.

554

555 Maohao Shen, Subhro Das, Kristjan Greenewald, Prasanna Sattigeri, Gregory Wornell, and Soumya
 556 Ghosh. Thermometer: Towards universal calibration for large language models. *arXiv preprint
 557 arXiv:2403.08819*, 2024.

558

559 Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Y Tang, Alejandro Cuadron, Chenguang
 560 Wang, Raluca Ada Popa, and Ion Stoica. Judgebench: A benchmark for evaluating llm-based
 561 judges. *arXiv preprint arXiv:2410.12784*, 2024.

562

563 Mistral AI Team. Mistral nemo, 2024. URL <https://mistral.ai/news/mistral-nemo>.

564

565 Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu,
 566 Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. *arXiv preprint
 567 arXiv:2305.17926*, 2023a.

568

569 Yidong Wang, Zhuohao Yu, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang,
 570 Rui Xie, Jindong Wang, Xing Xie, et al. Pandalm: An automatic evaluation benchmark for llm
 571 instruction tuning optimization. *arXiv preprint arXiv:2306.05087*, 2023b.

572

573 Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese,
 574 John Schulman, and William Fedus. Measuring short-form factuality in large language models.
 575 *arXiv preprint arXiv:2411.04368*, 2024.

576

577 Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason Weston,
 578 and Sainbayar Sukhbaatar. Meta-rewarding language models: Self-improving alignment with
 579 llm-as-a-meta-judge. *arXiv preprint arXiv:2407.19594*, 2024.

580

581 Ruixin Yang, Dheeraj Rajagopal, Shirley Anugrah Hayati, Bin Hu, and Dongyeop Kang. Con-
 582 fidence calibration and rationalization for llms via multi-agent deliberation. *arXiv preprint
 583 arXiv:2404.09127*, 2024.

584

585 Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek Wong, Emine Yilmaz, Shuming
 586 Shi, and Zhaopeng Tu. Benchmarking llms via uncertainty quantification. *Advances in Neural
 587 Information Processing Systems*, 37:15356–15385, 2024.

588

589 Hanling Yi, Feng Lin, Hongbin Li, Peiyang Ning, Xiaotian Yu, and Rong Xiao. Generation meets
 590 verification: Accelerating large language model inference with smart parallel auto-correct decod-
 591 ing. *arXiv preprint arXiv:2402.11809*, 2024.

592

593 Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. Evaluating
 594 large language models at evaluating instruction following, 2024. URL <https://arxiv.org/abs/2310.07641>.

595

596 Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving
 597 few-shot performance of language models. In *International conference on machine learning*, pp.
 598 12697–12706. PMLR, 2021.

594 Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang,
595 Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and
596 chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.
597

598 Kaitlyn Zhou, Dan Jurafsky, and Tatsunori Hashimoto. Navigating the grey area: How expressions
599 of uncertainty and overconfidence affect language models, 2023.

600 Lianghui Zhu, Xinggang Wang, and Xinlong Wang. JudgeLM: Fine-tuned large language models
601 are scalable judges. In *The Thirteenth International Conference on Learning Representations*,
602 2025. URL <https://openreview.net/forum?id=x5ELpEPn4A>.

603

604 Mingchen Zhuge, Changsheng Zhao, Dylan Ashley, Wenyi Wang, Dmitrii Khizbulin, Yunyang
605 Xiong, Zechun Liu, Ernie Chang, Raghuraman Krishnamoorthi, Yuandong Tian, et al. Agent-as-
606 a-judge: Evaluate agents with agents. *arXiv preprint arXiv:2410.10934*, 2024.

607

608

609

610

611

612

613

614

615

616

617

618

619

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648 A APPENDICES

649

650 A.1 LLMs USAGE IN OUR WORK

651

652 We hereby pledge that, regarding the use of LLMs in this work, we only used them for minor text
653 refinement, apart from their evaluation purposes in the experiments.

654

655 A.2 SELF-CONFIDENCE (SC) SETTING PROMPT

656

657 The following is the prompt template used in the Self-Confidence (SC) setting to elicit both
658 the judgment result and confidence score from the LLM. Placeholders such as {{question}},
659 {{answer_a}}, and {{answer_b}} are replaced with the actual instruction and output pairs dur-
660 ing evaluation.661 **SELF-CONFIDENCE (SC) SETTING PROMPT**

662

663

664 You are a helpful assistant in evaluating the quality of the
 665 outputs for a given instruction. Your goal is to select the best
 666 output for the given instruction and provide a confidence score
 667 (0-100) for your selection.
 668 Select the Output (a) or Output (b) that is better for the given
 669 instruction. The two outputs are generated by two different AI
 670 chatbots respectively.
 671 Evaluate the following outputs, and provide your best guess along
 672 with a confidence score in the following JSON format:
 673 {
 674 "selected_output": "Output (a)" or "Output (b)",
 675 "confidence_score": number,
 676 "explanation": "Your detailed explanation here"
 677 }
 678 # Instruction:
 679 {{question}}
 680 # Output (a):
 681 {{answer_a}}
 682 # Output (b):
 683 {{answer_b}}
 684 Your response must be in the JSON format as shown above. Do not
 685 output ANYTHING else. Do not provide the % symbol.

686

687

688 A.3 MULTIPLE-PROMPTING (MP) AND LOGP SETTINGS PROMPT

689

690 The following is the prompt template used in the Multiple-Prompting (MP) and LogP settings to
 691 elicit the judgment result from the LLM. Placeholders such as {{question}}, {{answer_a}},
 692 and {{answer_b}} are replaced with the actual instruction and output pairs during evaluation.

693

694 MULTIPLE-PROMPTING (MP) AND LOGP SETTINGS PROMPT

695 You are a helpful assistant in evaluating the quality of the
 696 outputs for a given instruction. Your goal is to select the best
 697 output for the given instruction.
 698 Select the Output (a) or Output (b) that is better for the given
 699 instruction. The two outputs are generated by two different AI
 700 chatbots respectively.
 701 Evaluate the following outputs, and provide your best guess in the
 702 following JSON format:
 703 {
 704 "selected_output": "Output (a)" or "Output (b)",
 705 "explanation": "Your detailed explanation here"

```

702
703     }
704     # Instruction:
705     {{question}}
706     # Output (a):
707     {{answer_a}}
708     # Output (b):
709     {{answer_b}}
710     Your response must be in the JSON format as shown above. Do not
711         output ANYTHING else.
712
713
714
```

A.4 LLM-AS-A-FUSER PROMPT

The following is the prompt template used in the LLM-as-a-Fuser framework to synthesize judgments from multiple models. Placeholders such as `{{question}}` , `{{answer_a}}` , `{{answer_b}}` , and the Jinja loop for JSON outputs are replaced with actual data during evaluation.

```

715     LLM-AS-A-FUSER PROMPT
716
717
718
719
720
721     You are a helpful assistant tasked with combining multiple model
722         responses to select the best output for the following
723         instruction: Evaluate the quality of multiple outputs for a
724         given instruction and select the best one based on specific
725         rules.
726
727     **Task:**
728     You will receive:
729     1. The instruction describing the task.
730     2. Multiple outputs (e.g., Output (a), Output (b)) generated by
731         different models.
732     3. A list of JSON outputs, each containing:
733         - selected_output: The chosen output (e.g., "Output (a)").
734         - confidence_score: A score showing the model's confidence (e.g
735             ., 85).
736         - explanation: Why the model chose that output.
737
738     Your goal is to:
739     - Review the JSON outputs and evaluate the original outputs (Output
740         (a), Output (b), etc.) using the evaluation rules.
741     - Pick the best output or create a new one by combining the best
742         parts of multiple outputs.
743     - Return a JSON response with the selected output, confidence_score
744         , and an explanation.
745
746     **Input:**
747     - **Instruction**: {{ question }}
748     - **Outputs**:
749         - Output (a): {{ answer_a }}
750         - Output (b): {{ answer_b }}
751     - **JSON Outputs**:
752         {% for output in json_outputs %}
753             - JSON Output {{ loop.index }}: {{ output }}
754         {% endfor %}
755
756     **Steps:**
757     1. **Check JSON Outputs**:
758         - Look at each selected_output, confidence_score, and
759             explanation.
760         - Use the explanation to understand why the model picked that
761             output.
762
```

```

756
757     - Note the confidence_score, but focus on explanation quality
758     and rule compliance.
759 2. **Evaluate Original Outputs**:
760     - Judge Output (a), Output (b), etc., against the evaluation
761     rules.
762     - Use JSON explanations to guide your evaluation.
763 3. **Pick or Combine**:
764     - Choose the best output if one clearly meets the rules.
765     - If no output is perfect, combine the best parts of multiple
766     outputs to create a better response.
767 4. **Explain Your Choice**:
768     - Say why you picked the output or created a new one.
769     - Mention the JSON outputs' explanations and scores, noting
770     agreements or differences.
771     - Show how your choice follows the rules better than others.

772     **Output Format:***
773     ```json
774     {
775         "selected_output": "Output (a)" or "Output (b)",
776         "confidence_score": number(0-100),
777         "explanation": "Why you chose this output or how you combined
778             outputs, referencing JSON explanations, confidence scores, and
779             evaluation rules."
780     }
781     ```

```

A.5 SUPPLEMENTARY TABLES

Table 6: Performance Comparison of Different LLMs under MP Setting

Model	Acc ↑	ECE ↓	ACE ↓	Brier Score ↓	MCE ↓	NLL ↓	TH Score ↑
DeepSeek-R1-0528	85.43	7.17	6.69	0.108	70.00	0.83	17.90
Qwen3-235B-A22B	78.86	13.00	12.04	0.151	70.00	0.98	16.85
OpenAI-o3-mini	76.00	18.49	18.77	0.184	43.91	1.84	17.08
R1-Distill-Llama	71.71	15.14	15.83	0.206	60.00	1.46	7.84
R1-Distill-Qwen	67.71	18.06	17.72	0.215	37.42	1.17	8.10
Claude-Sonnet-4	64.29	34.51	34.48	0.340	65.00	6.12	9.17
GPT-4.1	63.14	34.91	34.96	0.346	70.00	6.04	8.27
Gemini-2.5-Flash	52.57	14.43	14.77	0.220	43.33	1.44	7.40
DeepSeek-V3-0324	50.57	47.89	47.96	0.479	70.00	9.02	0.79
GPT-4o	49.71	47.09	46.97	0.463	58.13	7.64	1.68
GPT-4.1-mini	56.00	42.31	42.18	0.419	65.00	7.52	4.35
Llama-3.3-70B	42.86	54.31	54.28	0.537	70.00	9.49	-1.82
GPT-4.1-nano	28.29	67.43	67.41	0.663	71.26	10.86	-6.76
Mistral-Nemo	19.43	68.89	68.93	0.643	78.69	6.69	-4.35

A.6 SUPPLEMENTARY FIGURES

810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826

827 Table 7: Complete performance comparison of baseline aggregation methods and LLM-as-a-Fuser
 828 models. Values in parentheses represent changes compared to the original Self-Confidence (SC)
 829 setting (Table 1). In the LLM-as-a-Fuser section, changes are colored: **green** indicates improvements
 830 (better performance relative to SC baseline, considering metric directions: higher for \uparrow , lower for
 831 \downarrow), while **dark gray** indicates deteriorations (worse performance). Neutral changes (e.g., +0.00) are
 832 uncolored.

Method/Fuser Model	Acc \uparrow	ECE \downarrow	ACE \downarrow	MCE \downarrow	Brier \downarrow	NLL \downarrow	TH \uparrow
Entropy W. Voting	81.71	8.48	9.4	38.51	0.15	0.53	13.08
Conf. W. Voting	80.00	10.43	13.0	12.98	0.16	0.50	12.64
Majority Voting	80.00	10.77	12.9	12.89	0.16	0.50	12.58
Sqrt Conf. W. Voting	80.00	10.43	13.0	12.98	0.16	0.50	12.64
LLM-as-a-Fuser							
Qwen3-235B-A22B	86.29 (+8.86)	6.42 (-5.36)	8.9 (-3.3)	70.00(+6.50)	0.12 (-0.04)	0.39 (-0.13)	17.38 (-0.14)
OpenAI-o3-mini	84.86 (+10.57)	8.16 (-7.81)	9.1 (-8.1)	21.68 (-18.32)	0.13 (-0.07)	0.48 (-0.14)	16.39 (+3.56)
GPT-4.1-mini	83.14 (+27.43)	10.24 (-22.46)	11.8 (-21.0)	80.00(+35.79)	0.14 (-0.21)	0.47 (-0.53)	16.37 (+13.08)
Claude-Sonnet-4	81.71 (+17.42)	9.06 (-8.92)	10.3 (-7.7)	65.00(+20.00)	0.15 (-0.09)	0.54 (-0.15)	12.31 (+2.42)
GPT-4.1	80.00 (+16.86)	14.92 (-11.47)	15.6 (-11.2)	23.96(-31.04)	0.18 (-0.11)	0.69 (-0.16)	16.04 (+8.49)
DeepSeek-V3-0324	78.86 (+29.15)	12.71 (-23.50)	13.5 (-22.9)	75.00(+24.76)	0.17 (-0.20)	0.54 (-0.49)	11.96 (+9.50)
Gemini-2.5-Flash	78.00 (+38.57)	15.72 (-14.77)	16.0 (-14.4)	33.33(-22.78)	0.19 (-0.07)	0.67 (-0.11)	13.49 (+10.78)
Deepseek-R1-0528	68.57 (-8.29)	21.44 (+9.37)	22.3 (+11.0)	41.67(-18.33)	0.24 (+0.11)	1.65 (+1.23)	10.34 (-4.25)
Mistral-Nemo	67.43 (+47.14)	20.49 (-53.73)	20.5 (-53.7)	27.79(-52.21)	0.22 (-0.49)	0.95 (-2.06)	13.53 (+25.17)
Llama-3.3-70B	62.86 (+20.86)	24.38 (-22.99)	24.8 (-22.0)	38.85(-24.93)	0.27 (-0.18)	1.39 (-1.36)	9.80 (+9.00)
GPT-4.1-nano	57.71 (+30.85)	37.25 (-19.78)	37.4 (-19.7)	75.00(+2.50)	0.38 (-0.14)	2.36 (+0.98)	5.48 (+5.55)
GPT-4o	49.71 (+0.00)	44.07 (+4.82)	44.3 (+5.0)	48.00(-9.50)	0.44 (+0.04)	2.06 (+0.91)	0.72 (-0.85)

847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863

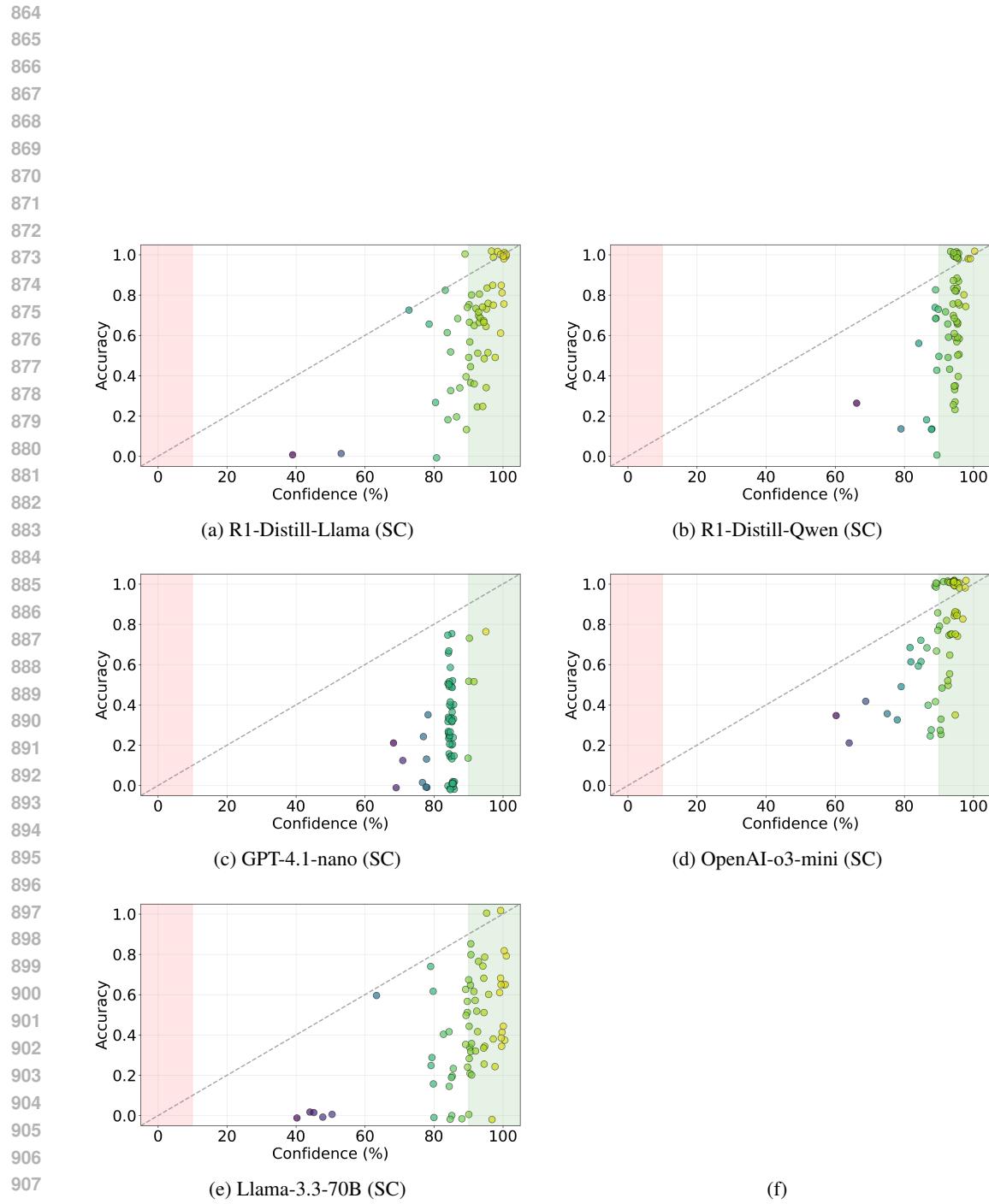


Figure 6: Supplementary figures: SC setting, page 1.

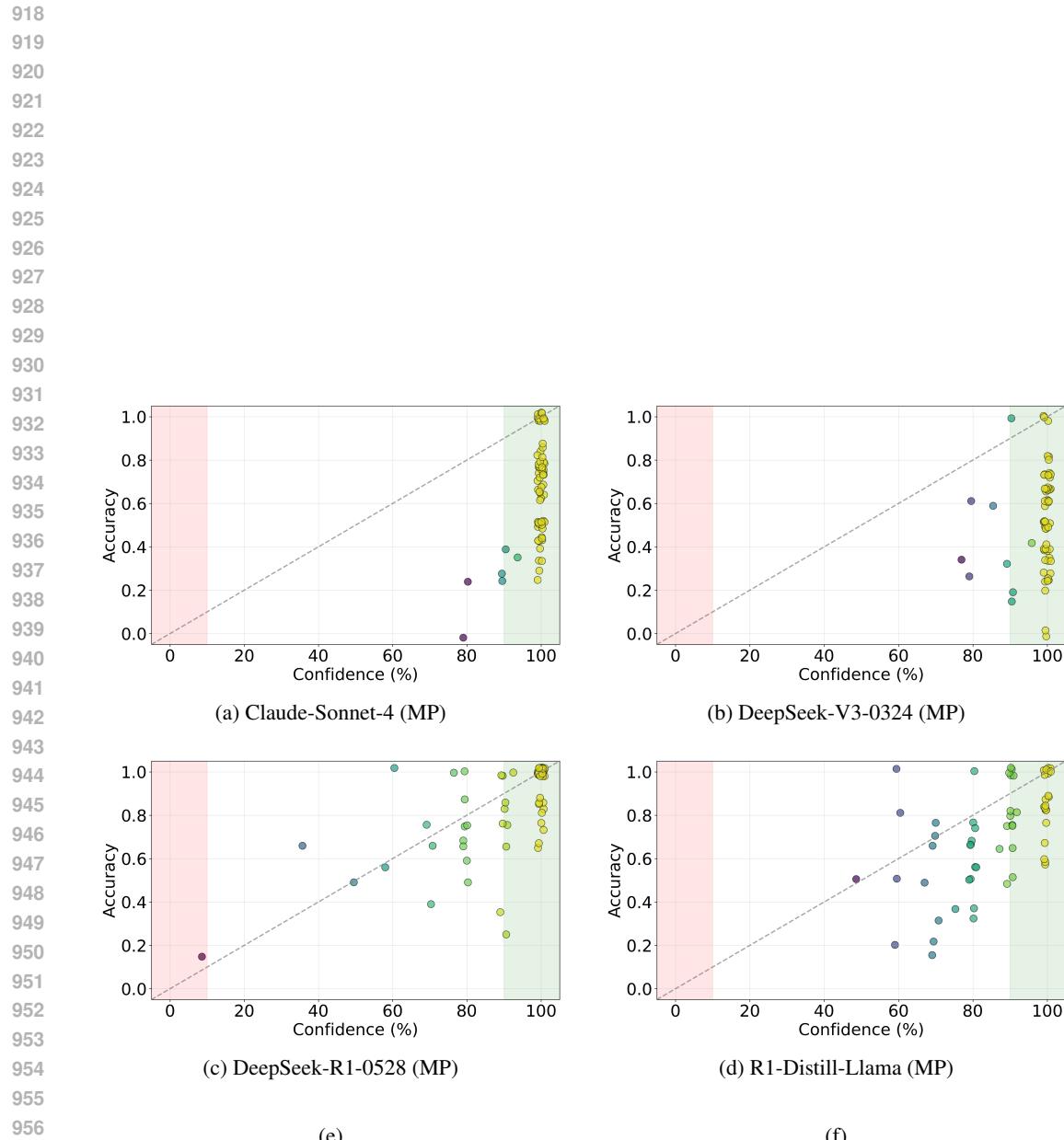


Figure 7: Supplementary figures: MP setting, page 2.

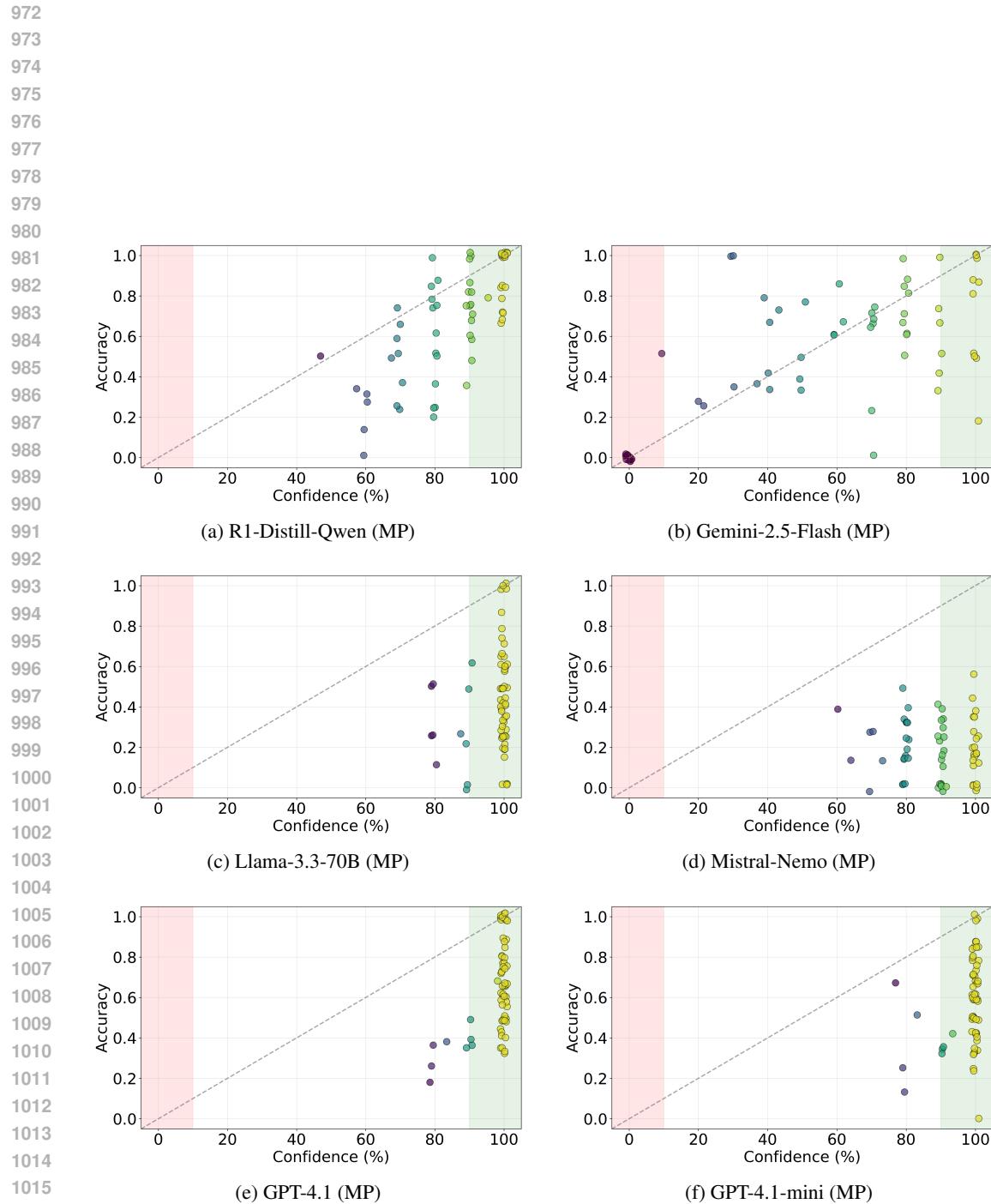


Figure 8: Supplementary figures: MP setting, page 3.

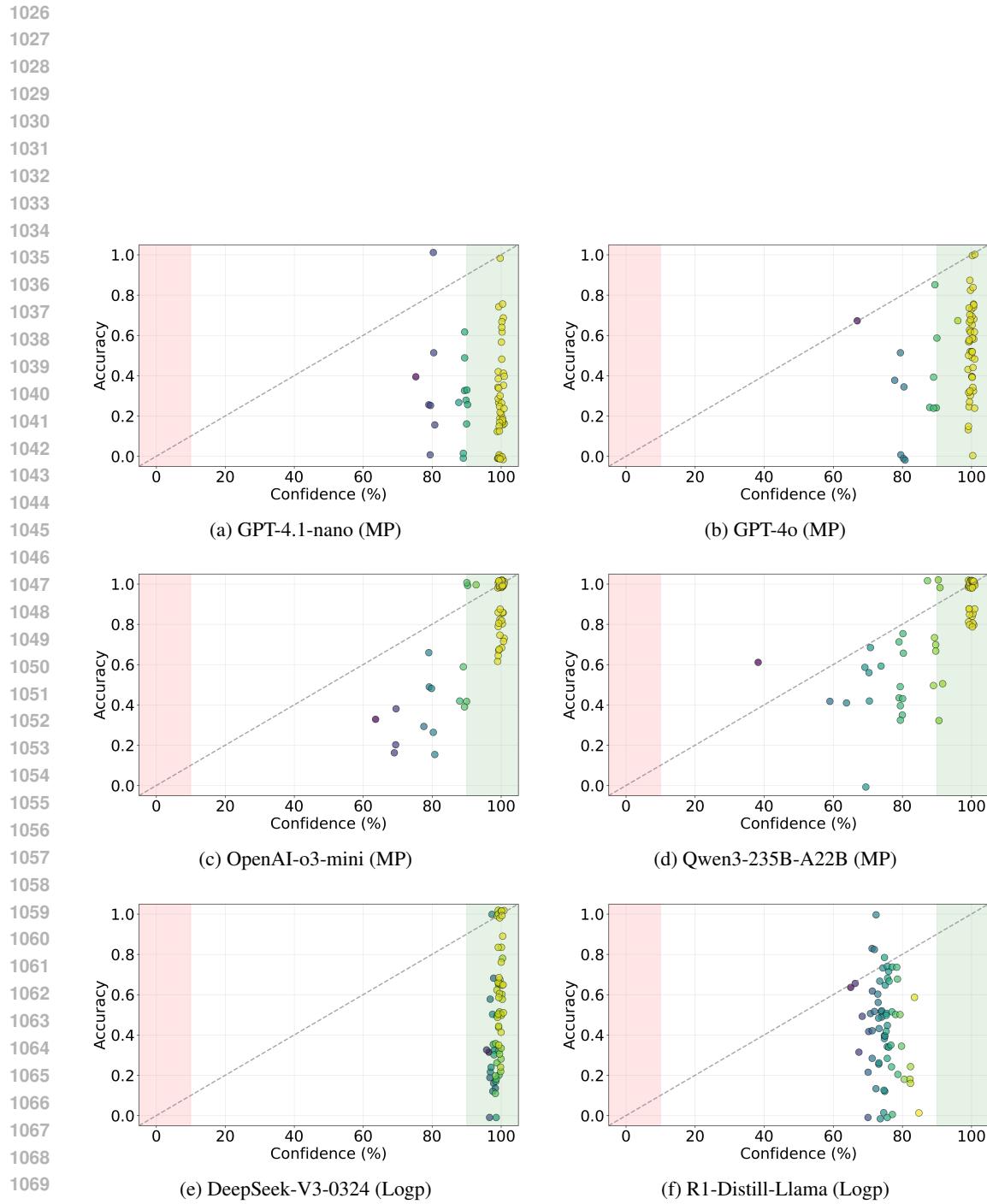


Figure 9: Supplementary figures: Logp setting, page 4.

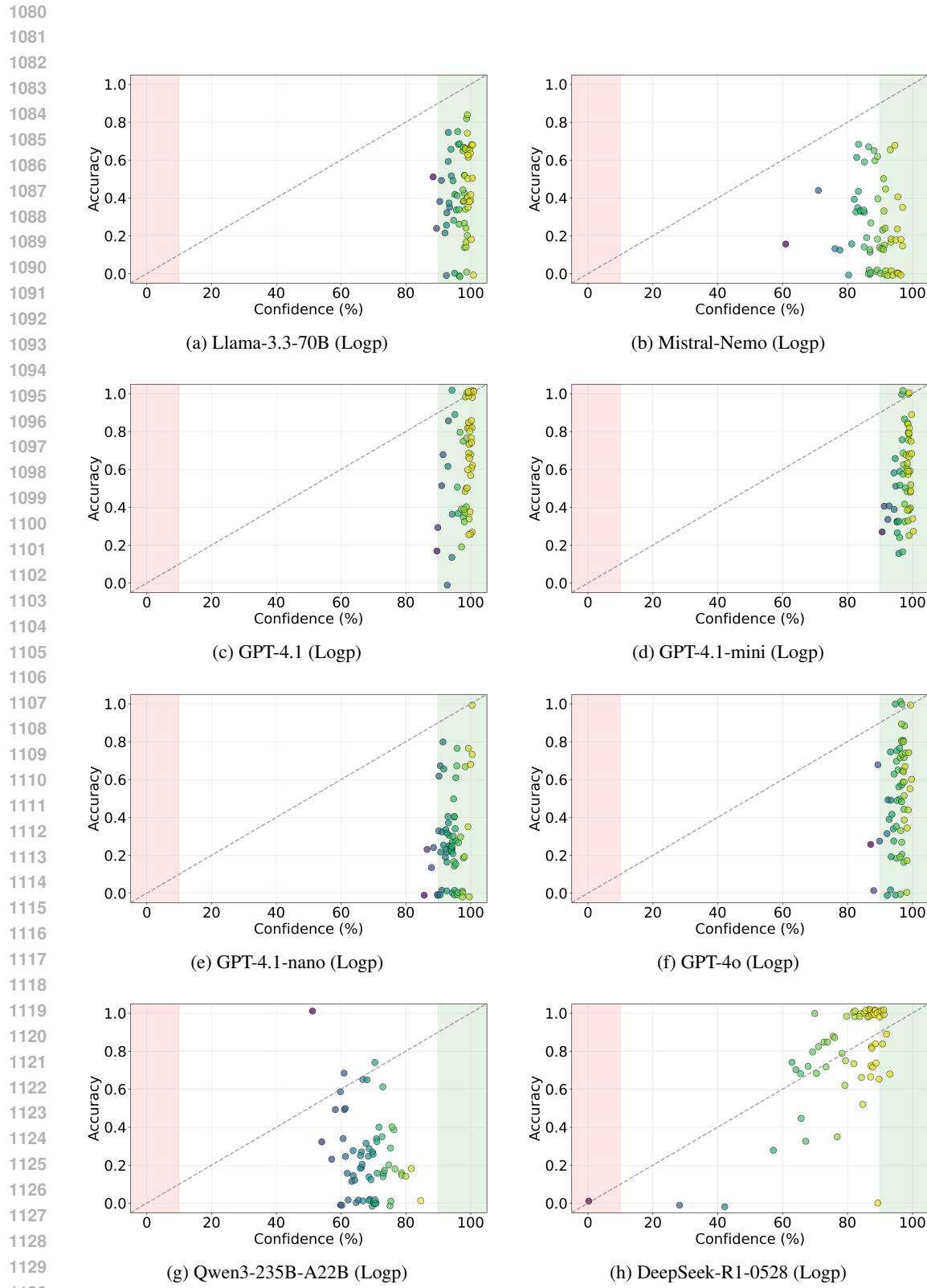


Figure 10: Supplementary figures: Logp setting, pages 5.