

000
001
002  **SPECTRUM TUNING:**
003 **POST-TRAINING FOR DISTRIBUTIONAL COVERAGE**
004 **AND IN-CONTEXT STEERABILITY**
005
006
007

008 **Anonymous authors**
009
010
011
012

013 Paper under double-blind review
014
015
016
017
018
019
020
021
022
023
024
025
026
027
028
029
030
031
032
033
034
035
036

ABSTRACT

013 Language model post-training has enhanced instruction-following and perfor-
014 mance on many downstream tasks, but also comes with an often-overlooked cost
015 on tasks with many possible valid answers. On many tasks such as creative writ-
016 ing, synthetic data generation, or steering to diverse preferences, models must
017 cover an entire distribution of outputs, rather than a single correct answer. We
018 characterize three desiderata for conditional distributional modeling: in-context
019 steerability, valid output space coverage, and distributional alignment, and docu-
020 ment across three model families how current post-training can reduce these prop-
021 erties. In particular, we disambiguate between two kinds of in-context learning:
022 ICL for eliciting existing underlying knowledge or capabilities, and *in-context*
023 *steerability*, where a model must use in-context information to override its priors
024 and steer to a novel data generating distribution. To better evaluate and improve
025 these desiderata, we introduce SPECTRUM SUITE, a large-scale resource compiled
026 from >40 data sources and spanning >90 tasks requiring models to steer to and
027 match diverse distributions ranging from varied human preferences to numerical
028 distributions and more. We find that while current post-training techniques elicit
029 underlying capabilities and knowledge, they hurt models' ability to flexibly steer
030 in-context. To mitigate these issues, we propose SPECTRUM TUNING, a post-
031 training method using SPECTRUM SUITE to improve steerability and distribu-
032 tional coverage. We find that SPECTRUM TUNING often improves over pretrained
033 and typical instruction-tuned models, enhancing steerability, spanning more of the
034 output space, and improving distributional alignment on held-out datasets.

037 **1 INTRODUCTION**
038
039

040 Current post-training recipes (Rafailov et al., 2024; Tie et al., 2025; Wang et al., 2025) have made
041 language models (LLMs) easier to use via instruction-following (Ouyang et al., 2022), improved
042 safety, and led to performance increases across many tasks, especially those with a single correct
043 answer (e.g., mathematical reasoning, programming, chat preferences, etc.). However, the effect of
044 current post-training on tasks requiring steerability and distribution matching is less studied. We
045 show that current post-training can also negatively impact three related desiderata for conditional
046 distributional modeling: in-context steerability, output coverage, and distributional alignment.

047 In this paper, we contribute: 1) an outline of these related desiderata, including the novel concept of
048 *in-context steerability*; 2) SPECTRUM SUITE, a dataset for evaluating and enhancing these desider-
049 ata; 3) a novel finding that while current post-training helps at many objective tasks, it can *hurt*
050 LLMs' in-context steerability; and 4) empirical evidence from our and related work that current
051 post-training hurts output coverage and distributional alignment. To alleviate these weaknesses, we
052 contribute 5) SPECTRUM TUNING, a post-training technique utilizing SPECTRUM SUITE to improve
053 these desiderata, and 6) show that our method enhances these properties compared to pretrained and
054 current instruction-tuned models. To our knowledge, our method is the first to improve distributional
055 alignment over pretrained models.

054 2 DESIDERATA FOR CONDITIONAL DISTRIBUTIONAL MODELING

056 Before the age of post-training, in-context learning was necessary to reliably get pretrained language
 057 models to perform tasks such as sentiment classification, translation, entailment, summarization, etc.
 058 (Brown et al., 2020; Dong et al., 2024). Let us call this use of in-context learning *capability elicitation*,
 059 as its main purpose is to elicit some latent knowledge or capability of a language model
 060 (Min et al., 2022b). As post-training methods have increased LLMs’ instruction-following capability,
 061 zero-shot instruct models have even surpassed their few-shot pretrained counterparts (Wei et al.,
 062 2022; Sanh et al., 2022; Ouyang et al., 2022), obviating the need for in-context capability elicitation.

063 **In-Context Steerability.** In contrast to knowledge elicitation, many tasks require steering, or mod-
 064 ifying output probabilities, based on novel information at inference time. For example, if a user
 065 wants an LLM to write an email in their style, it needs to either see examples of their writing or have
 066 an in-depth description of their style, and be able to effectively leverage this information to change
 067 its output distribution. This is distinct from pure capability/knowledge elicitation on unambiguous
 068 tasks, where the model can place a sharp prior on the “correct” answer. Instead, the model must 1)
 069 maintain a prior over many possible generation functions and 2) maximally leverage in-context in-
 070 formation in a well-calibrated way to form a posterior. Let us term this ability *in-context steerability*.
 071 For example, this steerability is necessary for predicting a particular user’s preferences or estimating
 072 an unknown numerical distribution from draws. In-context steerability can also be seen as implicit
 073 Bayesian reasoning (Qiu et al., 2025) or as a subset of in-context learning/instruction-following tasks
 074 where the model must utilize novel information in-context.

075 **Valid Output Coverage.** Many prompts entail multiple valid responses. For example, in creative
 076 story-writing, hypothesis proposal, and synthetic data generation, the number of possible valid out-
 077 puts can be thousands or more. While in some cases it may be sufficient to produce one reasonable
 078 output, more value may lie in producing *many* outputs so that a user can select the most interesting
 079 story, test all possible hypotheses, or otherwise span the entire task space. In the words of Wilson
 080 & Izmailov (2022), “we want the support of the model to be large so that we can represent any
 081 hypothesis we believe to be possible, even if it is unlikely.”

082 **Distributional Alignment.** Sometimes, a user may not want a particular output, but rather a *distribu-*
 083 *tion* over outputs (Meister et al., 2024). For example, Sorensen et al. (2024b) propose *distribu-*
 084 *tional pluralism* for modeling or representing a population by matching their opinion distribution. In
 085 addition, distributional alignment can simulate stochastic processes and estimate uncertainty. Dis-
 086 tinct from valid output coverage, distributional alignment includes a target probability mass function.

087 3 DATASET AND METHOD

089 3.1 SPECTRUM SUITE

090 To measure and elicit these
 091 properties, we compile datasets
 092 that either 1) exhibit natural
 093 person-to-person variation (e.g.,
 094 opinion modeling, chat prefer-
 095 ences, subjective NLP tasks);
 096 2) involve a large collection
 097 of interchangeable texts drawn
 098 from a particular distribution
 099 (e.g., synthetic data, poems in a
 100 particular format); 3) are i.i.d.
 101 draws from a random distribu-
 102 tion (e.g., draws from a normal
 103 distribution); or 4) involve rea-
 104 soning under uncertainty. We draw from >40 data sources in order to make >90 separate tasks.
 105 We unify each task data into a common format including: `description`: a natural language de-
 106 scription of the task, `input`: any given information for a particular data instance, and `output`:
 107 the output sequence which we would like the model to learn. Some tasks require an `input` associ-
 108 ated with each `output` (e.g., the question asked in a survey is needed to contextualize the answer),
 109 while other tasks consist of only `outputs`. In particular, we focus on individual modeling data on

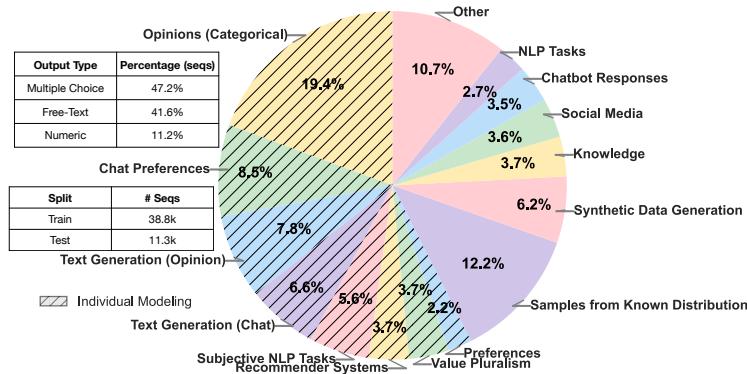


Figure 1: Task composition from SPECTRUM SUITE. Individual modeling tasks (data from the same person) are shaded.

108 tasks with human variation. We do so for a couple of reasons: many use cases involve steering to a
 109 particular individual at inference time; and these data sources are very rich as modeling each person
 110 involves a different data generation task. These data comprise 50.1k distinct sequences consisting
 111 of a description followed by multiple inputs/outputs. For summary statistics and task
 112 breakdown of SPECTRUM SUITE, see Figure 1. For information on all data sources, see App. C.
 113 We split SPECTRUM SUITE into non-overlapping train and test tasks, with held-out test tasks drawn
 114 from separate data sources to ensure generality.

115 **3.2 SPECTRUM TUNING** 

117 Let $T_i \in \mathcal{T}$ be some task (or, data generation process) that we want to model. Let Y_i be the output
 118 space to approximate, X_i be any known covariates (optional input), and Z_i be a latent context for
 119 the task (optional description). $T_i : X_i, Z_i \rightarrow P(Y_i)$ maps to a probability distribution over
 120 potential outputs. This is the classic meta-learning formulation (Hospedales et al., 2020), except that
 121 the target is a distribution over $P(Y^i)$ instead of a single y_i . Because the task T_i may be difficult to
 122 directly observe, we may instead wish to learn it from data (e.g., Monte Carlo samples).

123 The method (Algorithm 1) is simple: for a collection of tasks, tokenize the
 124 task context/description z_i followed
 125 by (randomly ordered) in-context ex-
 126 amples x_{ij}, y_{ij} , then perform super-
 127 vised finetuning calculating cross-
 128 entropy loss *only* on the output to-
 129 kens. Because cross-entropy loss on
 130 Monte Carlo samples from a distribu-
 131 tion encourages a well-calibrated es-
 132 timate of the underlying distribution
 133 in the underfit regime (≤ 1 epoch, Ji
 134 et al. 2021) the optimal model solu-
 135 tion is to approximate the true under-
 136 lying distribution $P(Y_i)$.

137 To build intuition on how SPECTRUM
 138 TUNING supports the desiderata, let
 139 us consider a few cases. When a
 140 model predicts the first output, it
 141 must rely only on the description, and
 142 shift its probabilities to outputs fit-
 143 ting the description. Because there
 144 can be many possible valid outputs
 145 and the model has no information
 146 about which output to expect, it is in-
 147 centivized to *cover* the entire possi-
 148 ble distribution of outputs. Addition-
 149 ally, if the distribution over valid out-
 150 puts is skewed in some predictable
 151 way (e.g., an opinion distribution),
 152 the model is further incentivized to *match*
 153 said distribution. On subsequent out-
 154 puts, the model must *steer* its output
 155 distribution, utilizing in-context ex-
 156 amples to update its beliefs in a well-calibrated
 157 way. Additionally, SPECTRUM SUITE
 158 tasks allow the model to utilize assump-
 159 tions which don’t apply to the pretrain-
 160 ing distribution: predictions are invari-
 161 ant to output ordering,¹ the underly-
 162 ing generative process remains constant,
 163 and the model can concentrate all proba-
 164 bility mass on valid outputs instead of on
 165 other possible text continuations. In many
 166 ways, SPECTRUM TUNING is simi-
 167 lar to supervised fine-tuning on instruc-
 168 tion data (Zhang et al., 2025c), as loss is cal-
 169 culated only on an output. However, it
 170 differs in several important respects: 1) many identically-distributed outputs are
 171 included in-context, encouraging meta-learning; 2) training on data that is distributional in nature;
 172 3) sole focus on distribution fitting instead of chat-style data; and 4) inputs are optional, unlike chat
 173 user messages which are always required.

Algorithm 1 SPECTRUM TUNING

Inputs: Pretrained LM m_θ ; train task distribution $\mathcal{T}^{\text{train}}$; to-
 kenizer $t(\cdot)$ with template for description/input/output;
 terminal token $\langle \text{END} \rangle$; loss ignore index i_{drop} ; descrip-
 tion drop probability p_{drop} (default 0.2).

Output: Finetuned parameters θ'

- 1: **for** each task $T \sim \mathcal{T}^{\text{train}}$ **do** ▷ Sample a task
- 2: Sample description z and support set $S = \{(x_j, y_j)\}_{j=1}^n$.
- 3: Randomly permute indices π of $\{1, \dots, n\}$.
- 4: **if** $\text{Uniform}(0, 1) > p_{\text{drop}}$ **then** ▷ Keep description
- 5: $seq \leftarrow t(z) \| t(x_{\pi[0]}) \| t(y_{\pi[0]}) \| \langle \text{END} \rangle$
- 6: $labels \leftarrow i_{\text{drop}}(t(z) \| t(x_{\pi[0]}) \| t(y_{\pi[0]}) \| \langle \text{END} \rangle)$ ▷ Loss on first output, no loss on description/output
- 7: **else** ▷ Description dropout w/ prob. p_{drop}
- 8: $seq \leftarrow t(x_{\pi[0]}) \| t(y_{\pi[0]}) \| \langle \text{END} \rangle$
- 9: $labels \leftarrow i_{\text{drop}}(t(x_{\pi[0]}) \| t(y_{\pi[0]}) \| \langle \text{END} \rangle)$ ▷ No loss on first output if description is missing
- 10: **end if**
- 11: **for** j in $\pi[1 :]$ **do** ▷ Add remaining
- 12: $seq \leftarrow seq \| t(x_j) \| t(y_j) \| \langle \text{END} \rangle$
- 13: $labels \leftarrow labels \| i_{\text{drop}}(t(x_j)) \| t(y_j) \| \langle \text{END} \rangle$ ▷ Loss on output, no loss on input
- 14: **end for**
- 15: $L \leftarrow \text{CrossEntropy}(m_\theta(seq), labels)$
- 16: $\theta \leftarrow \theta - \eta \nabla_\theta L$
- 17: **end for** ▷ Train for one epoch
- 18: **return** $\theta' \leftarrow \theta$

¹i.e. “exchangeable” in Bayesian analysis (Kokolakis, 2010), as the posterior is invariant to sample order.

162 3.3 IMPLEMENTATION DETAILS

We train models from three families using SPECTRUM TUNING on the train tasks from SPECTRUM SUITE: gemma-3-12b (Team et al., 2025), Llama-3.1-8B (Grattafiori et al., 2024), and Qwen3-14B (Yang et al., 2025). We refer to pretrained or base models as PT models and instruction-tuned post-trained models as IT models, and utilize each family’s provided PT/IT model as baselines. To match our meta-learning task setup (as opposed to chat), we adapt each model’s chat template to use the description/input/output roles instead of system/user/assistant (cf. Fig. A2). For SPECTRUM TUNING, we initialize with the PT model weights, except for the uninitialized (un)embedding weights for the two or three special format tokens which we initialize from the IT model. See App. F for more training details.

4 IN-CONTEXT STEERABILITY

We use SPECTRUM SUITE to evaluate models' ability to steer to varied generation tasks. We measure k -shot learning by 1) fitting the description and examples from a single task into context, 2) measuring the loss (negative log-likelihood) of each output conditioned on the prior examples under the model m_θ : $\text{NLL}_{m_\theta}(y_k) = -\log p_{m_\theta}(y_k|z, y_0, \dots, y_{k-1})$. Additionally, for multiple-choice datasets, we calculate the accuracy of the output: whether the greedily-decoded model response results in the correct answer. For each task, we choose K_{\max} such that it maximizes the total number of examples that we can evaluate when we restrict to only sequences with at least K_{\max} examples that fit into a 1024-token context-window. In order to maximize sample efficiency and evaluate a model's ability to steer for varied k , we report the average loss and accuracy for k -shot learning for $k \in \{1, \dots, K_{\max}\}$.

First, we ask: how does current instruction-tuning impact in-context steerability? For the PT models, we use the same prompt template for all models, with **Description**:/**Input**:/**Output**: delineated by newlines. To ensure we are leveraging maximum performance from the IT models, we test each IT model's performance on both the PT prompt and two performing prompt template (see App. L). SUITE for the PT/IT models. We include the principal results below.

Current instruction-tuning hurts in-context steerability. First, let’s examine the change in accuracy for the IT models. We report accuracy for all categorical data (multiple-choice + small support numeric distributions) in Figure 2. Out of 76 model family/task comparisons, instruction-tuning *significantly decreases* accuracy in 35 cases, doesn’t significantly affect accuracy in 33 cases, and significantly increases accuracy in only 7 cases. Additionally, two of the seven comparisons where instruction-tuning helped were on predicting an individual’s chatbot preferences—which is adjacent to precisely what instruct models are optimized for (chat). The performance drop is even more stark on loss: for Gemma and Qwen, loss is higher on 50/50 comparisons, while on Llama loss is worse in 11 cases, the same in 11 cases, and better in 3 cases. Loss results are similar on the free-text

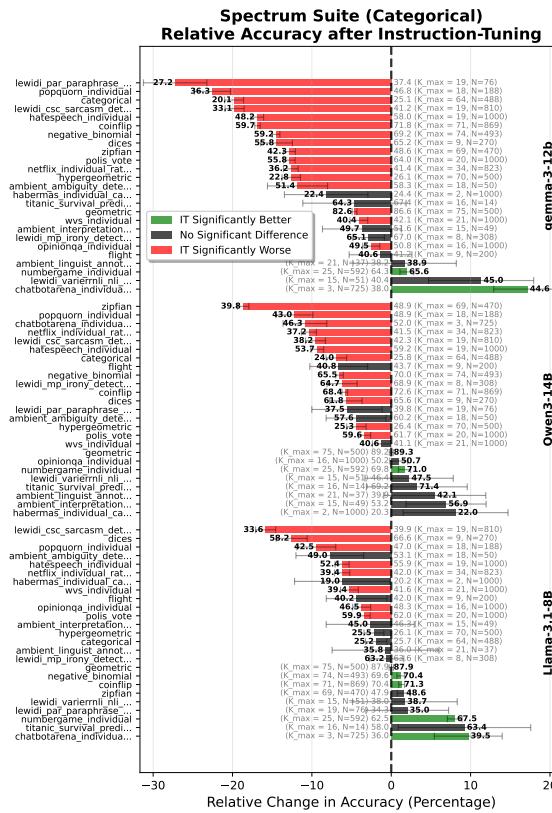


Figure 2: Change in accuracy on SPECTRUM SUITE from the pretrained to instruction-tuned model. Current instruction-tuning hurts in-context steerability.

216 SPECTRUM SUITE datasets: out of 144 comparisons, IT loss is worse than PT loss in 117 cases, tied
 217 in 25 cases, and better only in 2 cases.
 218

219 **ICL for general capability elicitation is not de-
 220 graded by instruction-tuning.** To disambiguate in-
 221 context steerability from general capability elicita-
 222 tion, we also run the exact same experiment with
 223 eight general capability task datasets (Fig. 3). In
 224 contrast with the SPECTRUM SUITE datasets, accu-
 225 racy *increases* in 8 of 24 cases, is the same in 13
 226 cases, and decreases in 2 cases.
 227

228 All in all, we believe that this characterizes a differ-
 229 ence in behavior for IT models—while they main-
 230 tain the ability to utilize in-context demonstrations
 231 for general capability elicitation, they seem to strug-
 232 gle to adapt at tasks that require heavy in-context
 233 steerability. Limited prior work has suggested that
 234 instruction-tuned models sometimes perform better
 235 without in-context examples (Asai et al., 2024; Lam-
 236 bert et al., 2025); however, to our knowledge, ours is
 237 the first work to empirically characterize this in-context
 238 learning performance degradation for in-
 239 context steerability tasks.
 240

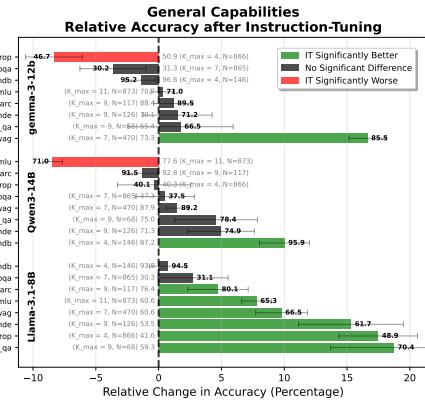
241 What explains this difference? While we leave an in-depth exploration of this phenomenon to future
 242 work, we hypothesize that it could be due to some combination of 1) instruction-tuning inducing very
 243 strong priors that are difficult to override even with in-context demonstrations, 2) over-optimization
 244 on tasks with a single ground truth, or 3) overfitting to particular benchmarks.
 245

246 4.1 SPECTRUM TUNING AND IN-CONTEXT STEERABILITY ON HELD-OUT TASKS

247 We have characterized that current instruction-tuned models struggle at in-context steerability, but
 248 how does our method compare? We evaluate Spectrum-Tuned (ST) models on SPECTRUM SUITE
 249 test tasks and compare them to their PT and IT counterparts (Table 1). Note that the test task data
 250 sources have no overlap with the train split, requiring generalization.
 251

252 Multiple-Choice Datasets	Metric	253 gemma-3-12b			254 Qwen3-14B			255 Llama-3.1-8B			
		256 ST (ours)	257 PT	258 IT	259 ST	260 PT	261 IT	262 ST	263 PT	264 IT	
265 habermas_individual_categorical ($K_{\max}=2$, $N=1000$)	266 Loss	2.47	2.50	10.5	1.97	2.62	9.10	1.99	2.58	2.74	
267 Acc	268	23.8	24.4	22.4	23.5	20.3	22.0	20.8	20.2	19.0	
269 wvs_individual ($K_{\max}=21$, $N=1000$)	270 Loss	1.36	1.50	4.10	1.48	1.74	4.35	1.42	1.57	1.76	
271 Acc	272	42.6	42.1	40.4	44.3	41.1	40.6	41.7	41.6	39.4	
273 numbergame_individual ($K_{\max}=25$, $N=592$)	274 Loss	.639	.705	1.80	.621	.697	1.28	.618	.864	.770	
275 Acc	276	70.2	64.3	65.6	70.6	69.8	71.0	69.1	62.5	67.5	
277 chatbotarena_individual_prefs ($K_{\max}=3$, $N=725$)	278 Loss	1.43	1.62	4.94	1.34	1.47	4.39	1.39	1.76	1.77	
279 Acc	280	38.6	38.0	44.6	51.4	52.0	46.3	38.9	36.0	39.5	
281 flight ($K_{\max}=9$, $N=200$)	282 Loss	1.09	1.32	4.06	1.08	1.29	2.92	1.12	1.45	1.41	
283 Acc	284	39.8	41.2	40.6	43.7	43.7	40.8	33.4	42.0	40.2	
285 Free-Text Datasets	Metric	286 ST (ours)	287 PT	288 IT	289 ST	290 PT	291 IT	292 ST	293 PT	294 IT	
		295 novacomet_hypothesis	296 ($K_{\max}=11$, $N=155$)	297 Loss	298 104	104	135	106	106	129	107
299 novacomet_premise	300 ($K_{\max}=55$, $N=51$)	301 Loss	302 27.7	303 28.0	304 35.5	28.1	27.5	38.0	27.8	27.7	28.6
305 habermas_question	306 ($K_{\max}=29$, $N=30$)	307 Loss	308 23.8	309 23.1	310 41.4	23.8	24.0	31.8	23.8	23.8	24.8
311 habermas_opinions	312 ($K_{\max}=2$, $N=186$)	313 Loss	314 930	315 928	316 1070	948	949	1070	943	944	991
317 habermas_individual	318 ($K_{\max}=2$, $N=1000$)	319 Loss	320 164	321 164	322 203	168	168	210	166	167	176
323 numbergame_perc	324 ($K_{\max}=24$, $N=182$)	325 Loss	326 4.23	327 4.22	328 6.68	4.22	4.24	5.61	4.24	4.43	4.41
329 globalqa	330 ($K_{\max}=8$, $N=231$)	331 Loss	332 14.0	333 14.4	334 21.5	14.0	14.4	20.9	14.2	14.7	15.6
335 chatbotarena_prompts	336 ($K_{\max}=3$, $N=988$)	337 Loss	338 70.2	339 69.4	340 117	69.1	68.2	97.8	72.0	72.0	77.6
341 chatbotarena_assistant	342 ($K_{\max}=5$, $N=716$)	343 Loss	344 127	345 125	346 259	124	124	169	134	133	149
347 chemistry_esol	348 ($K_{\max}=8$, $N=59$)	349 Loss	350 8.94	351 8.37	352 12.9	8.07	8.47	11.8	8.28	8.51	8.55
353 chemistry_oxidative	354 ($K_{\max}=9$, $N=101$)	355 Loss	356 7.57	357 7.58	358 11.6	7.64	7.84	10.2	7.64	7.72	7.84

289 Table 1: In-context steerability on held-out SPECTRUM SUITE-Test. SPECTRUM TUNING generally
 290 matches or improves upon the pretrained model performance. Best values (and ties, failing to find a
 291 significant difference at $\alpha = .05$) are bolded.
 292



293 Figure 3: Current instruction-tuning generally helps on capability benchmarks.
 294

270

271

272

273

274

275

276

Expected Calibration Error (ECE, \downarrow)	gemma-3-12b			Qwen3-14B			Llama-3.1-8B		
	ST (ours)	PT	IT	ST (ours)	PT	IT	ST (ours)	PT	IT
habermas.individual.categorical	0.116	0.069	0.239	0.032	0.05	0.198	0.037	0.084	0.055
wws.individual	0.006	0.015	0.223	0.017	0.02	0.191	0.005	0.012	0.024
numbergame.individual	0.015	0.029	0.163	0.027	0.026	0.108	0.028	0.024	0.017
chatbotarena.individual_prefs	0.020	0.041	0.194	0.048	0.046	0.189	0.046	0.075	0.049
flight	0.011	0.040	0.271	0.038	0.035	0.228	0.046	0.070	0.038

277

278

279

280

281

Table 2: Calibration on SPECTRUM SUITE-Test, binning label token probabilities every decile for expected calibration error ($ECE = \sum_{b=1}^B \frac{n_b}{N} |\text{acc}(b) - \text{conf}(b)|$, where $B = 10$ bins, n_b is the number of samples in bin b , $\text{acc}(b)$ is the accuracy in bin b , and $\text{conf}(b)$ is the average confidence in bin b). SPECTRUM TUNING (ST) usually results in the best calibration (9/15 cases).

282

283

284

285

286

287

288

SPECTRUM TUNING usually matches, and sometimes improves upon, PT steerability. Out of 15 multiple-choice (MC) loss comparisons, ST ties with PT models in one case and achieves lower loss compared to PT models in 14 cases. On MC accuracy, ST matches/improves/worsens on 10/3/2 comparisons. On the free-text datasets, ST matches PT in 28 cases, is worse in 1 case and is better in 4 cases. In most cases, SPECTRUM TUNING matches (but does not beat) the very strong baseline of a pretrained model at in-context steerability, but does improve performance more often than it hurts performance.

289

290

291

292

Models trained with SPECTRUM TUNING most often have the best calibration. We report calibration in Table 2. In 9/15 cases, the ST models have the best calibration. Additionally, the Gemma and Qwen IT models have worse calibration in 10/10 cases than their pretrained counterparts, showing another side effect of heavy instruction-tuning (cf. Tian et al. 2023; OpenAI et al. 2024).

293

294

5 SPANNING THE OUTPUT SPACE (OR; DIVERSITY VS. VALIDITY)

295

296

297

298

299

300

301

302

303

304

305

306

To measure how each model trades off validity and diversity, we create 22 generation tasks for which there can be many valid values and we can programmatically verify correctness ($\mathbb{1}_{\text{correct}}$). Given a prompt, we generate 100 completions o_1, \dots, o_{100} (temperature = 1 here and throughout) from each model, and report the following statistics: the percentage of outputs which are valid ($\sum_{i=1}^{100} \mathbb{1}_{\text{correct}}(o_i)$), the percentage of valid generations that are unique ($\frac{|\text{dedup}(\{o_i : \mathbb{1}_{\text{correct}}(o_i) = 1\})|}{\sum_{i=1}^{100} \mathbb{1}_{\text{correct}}(o_i)}$), and the number of distinct valid generations (or, *yield*: $|\text{dedup}(\{o_i : \mathbb{1}_{\text{correct}}(o_i) = 1\})|$). We perform deduplication with exact string matching. Yield is a particularly important metric for settings such as synthetic data generation, ideation, or creative writing where you want to cover a space as much as possible within some requirements. Additionally, we evaluate each model under three settings: zero-shot with a task description, three-shot with no task description, and three-shot with a task description (also see App. M). Results can be found in Fig 4. Tasks are the same across models.

307

308

309

310

Instruction-tuned models have high validity but low diversity. IT models produce valid outputs $> 70\%$ of the time, even in the zero-shot setting. However, this comes at the price of diversity, resulting in fewer than 30 unique valid generations in few-shot settings. Yield is even lower in the zero-shot setting—Qwen and Gemma average yields of only 5–6, while Llama averages only 24.

311

312

313

314

315

316

Pretrained models are more diverse, but rely on few-shot examples for validity. Pretrained models do not suffer from the same mode collapse, and consistently have higher diversity ($> 40\%$ of valid generations are unique). However, this comes at a trade-off with validity, where their generations are universally less valid than the IT models'. The pretrained models also rely heavily on the few-shot examples to elicit valid generations, achieving a validity of $< 20\%$ in the zero-shot case. However, in the few-shot cases, they have a significantly higher yield than the instruction-tuned models due to their higher coverage of the space.

317

318

319

320

321

322

SPECTRUM TUNING offers a Pareto improvement on diversity and validity, matching or exceeding pretraining yield. In eight of nine model/setting comparisons, SPECTRUM TUNING offers either a Pareto or strict improvement over the PT/IT models on validity/diversity. In all eight settings with a Pareto improvement, this also leads to a higher yield—i.e., **for a fixed generation budget, SPECTRUM TUNING generates the most unique valid generations.**

323

SPECTRUM TUNING achieves much higher yield in the zero-shot setting. Focusing in on the zero-shot setting, SPECTRUM TUNING particularly shines. The IT models are able to follow the

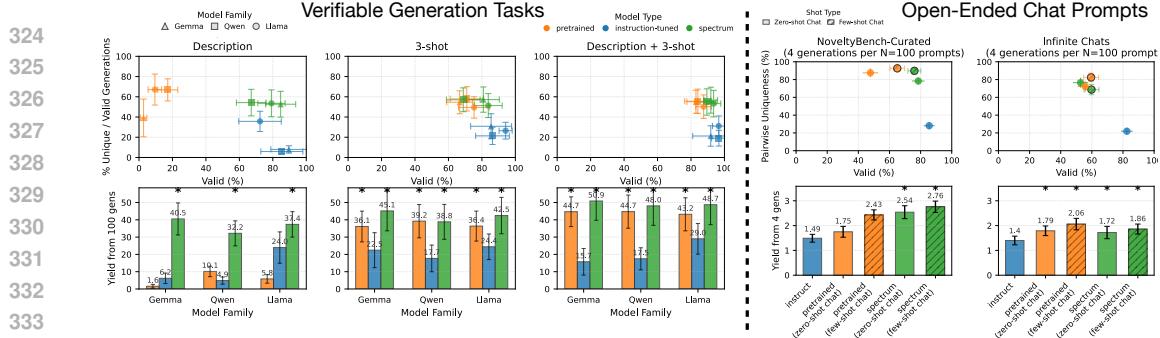


Figure 4: Diversity vs. Validity. Left: Results on 22 verifiable tasks across 100 generations. Right: Human-annotated validity results on two sets of 100 open-ended prompt sets (Gemma). SPECTRUM TUNING generally offers a Pareto improvement on diversity-validity over PT/IT models. In particular, SPECTRUM TUNING increases the yield (# of unique usable generations) in the zero-shot case and on NoveltyBench-Curated. Error bars are 95% confidence intervals over the SEM, and asterisks (*) show the best in family performance (within 95% confidence).

description and produce a valid output, but have very low diversity ($\sim 30\%$ for Llama, $\sim 5\%$ for Qwen and Gemma). Meanwhile, the pretrained models are unable to consistently generate valid outputs ($< 20\%$ validity). ST models, however, are able to follow the instructions and produce valid outputs $> 60\%$ of the time while maintaining 50% diversity. This leads to much higher yields compared to PT and IT models (Gemma: 40.5 vs. 6.2; Qwen: 32.2 vs. 10.1, Llama: 37.4 vs. 24.0).

SPECTRUM TUNING’s gains hold across temperature values. One way to trade-off validity for diversity for a given model is sweeping temperature. To ensure that our results hold across temperatures, we ran the same experiment with $T = [10, 5, 2, 1.5, 1, .9, .7, .5]$. We found that SPECTRUM TUNING A) still expanded the Pareto frontier and B) gave the highest possible yield when choosing an optimal temperature (see App. D for more details).

5.1 HUMAN EVAL

We extend the verifiable task experiments with a human evaluation on open-ended chat prompts: NoveltyBench-Curated (100 prompts, Zhang et al. 2025d) and Infinite-Chats-Eval (100 prompts, yet to be published, obtained from the authors). However, SPECTRUM TUNING does not optimize for chat capabilities, but rather for fitting to description/input/output. In order to elicit chat capabilities in-context, we try two approaches: zero-shot chat, where we prompt with description: You are a helpful AI assistant, input: <prompt>; and few-shot chat, where we utilize the same description and four examples of prompt inputs and chat responses as outputs. Additionally, we use a similar prompt for the pretrained model as a baseline, with the description, a prefix for the prompt of User:, and an output prefix of Assistant:, zero-shot and with the same four few-shot examples (similar to URIAL, Lin et al. 2023). More details in App. M.

For each prompt, we generate four completions from the model. We recruit annotators to judge whether a given generation is a valid response to the prompt. Each generation is annotated by four annotators, and we count the generation as valid if three of four annotators marked it as valid. Overall, annotators had a 73% pairwise agreement rate. Due to the cost of the evaluation, we only annotate generations for one model family, gemma-3-12b. For additional evaluation details, see App. H. For calculating diversity, we follow NoveltyBench’s approach and utilize their deberta-v3-large-based model for assigning two generations as duplicates. We report the Pairwise Uniqueness %, or the probability that any two valid generations are not considered duplicates, along with yield. Results are in Tab. 4.

Few-shot pretrained models improve yield over instruct models. While lagging in validity, pretrained models produce much more diverse responses than their instruct counterparts, and are able to achieve $>40\%$ validity from few-shot chat examples, improving yield and offering a strong baseline.

SPECTRUM TUNING offers a Pareto improvement on diversity/validity and improves yield over baselines on NoveltyBench-Curated. On NoveltyBench-Curated, our method offers higher validity than the pretrained model, while offering substantially higher diversity than the instruct

378 model. This improvement results in a statistically significant increase in yield over the baselines.
 379 On Infinite-Chats, the pretrained models and our models do not perform significantly differently,
 380 covering roughly the same space on the Pareto frontier and on yield. While disambiguating the
 381 reason for the differing performance may require further investigation, we do note that many of
 382 the Infinite-Chat eval prompts have specific requirements, such as “In five words”, “In a couple of
 383 paragraphs,” etc., which our models often fail to adhere to. In contrast, the NoveltyBench-Curated
 384 prompts are far more open-ended. It may be that our model performs best at generating shorter
 385 outputs, and future work may be needed to enhance precise instruction-following while maintaining
 386 diversity. However, on both datasets, the instruct model has significantly lower yield and diversity.
 387

388 6 DISTRIBUTIONAL ALIGNMENT AND PLURALISM

390 Next, we evaluate our system’s ability to steer to match a target distribution. We utilize seven held-
 391 out datasets² mainly focusing on human response distributions and a synthetic random draws task.
 392 We prompt models zero-shot with a description of the setting and a target question. We then calculate
 393 the probability of each possible valid output, normalize, and calculate Jensen-Shannon divergence
 394 from the target distribution. We also measure coverage, or the total probability mass on the set of
 395 valid answers. Results are in Table 3, and takeaways are as follows. (More details in App. N.)
 396

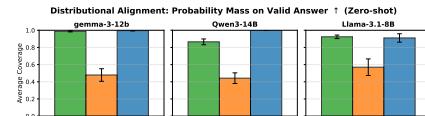
Dataset	gemma-3-12b			Qwen3-14B			Llama-3.1-8B		
	ST (ours)	PT	IT	ST (ours)	PT	IT	ST (ours)	PT	IT
Machine Personality Inventory (N=120, $ Y =6$)	0.083	0.126	0.347	0.100	0.093	0.405	0.063	0.087	0.131
Rotten Tomatoes (N=1000, $ Y =2$)	0.032	0.032	0.134	0.028	0.028	0.122	0.035	0.035	0.086
NYTimes Books (N=940, $ Y =4$)	0.051	0.063	0.328	0.070	0.088	0.344	0.046	0.061	0.247
GlobalOQA (N=1000, $ Y \leq 6$)	0.077	0.094	0.270	0.090	0.088	0.274	0.091	0.108	0.163
Urn (N=1000, $ Y \leq 6$)	0.021	0.071	0.185	0.051	0.059	0.198	0.032	0.124	0.086
Habermas (N=658, $ Y =7$)	0.149	0.147	0.436	0.123	0.127	0.434	0.151	0.155	0.242
Number Game (N=1000, $ Y =2$)	0.051	0.049	0.138	0.052	0.043	0.131	0.055	0.060	0.094

404 Table 3: Distributional alignment results. Instruction-tuning drastically hurts distributional alignment.
 405 SPECTRUM TUNING generalizes to unseen tasks and improves or matches distributional
 406 alignment compared to the pretrained model. Best result (within 95% statistical significance) in
 407 bold. N is the number of distinct instances, $|Y|$ is the number of possible outputs.
 408

409 **Instruction-tuned models have higher distributional divergence than pretrained models.** In
 410 line with prior work (Sorensen et al., 2024b), we find that instruction-tuned models show higher
 411 distributional divergence than pretrained models on all tasks. We believe that this is in large part
 412 due to their low-entropy, spiky distributions. In other words, for distribution matching, current
 413 instruction-tuning categorically hurts performance compared to the pretrained model.
 414

415 **SPECTRUM TUNING generally improves distributional alignment over pretrained models.** Out
 416 of 21 model/dataset comparisons, SPECTRUM TUNING improves distributional alignment in 10
 417 cases, matches PT models in 10 cases, and degrades performance in 1 case. Pretrained models are
 418 a strong baseline—the pretraining objective entirely consists of trying to estimate a well-calibrated
 419 distribution over the next token. To our knowledge, ours is the *first method to improve distributional
 420 alignment on unseen datasets* over pretrained models.

421 **SPECTRUM TUNING improves coverage of valid an-
 422 swers over pretrained models and roughly matches
 423 instruction-tuned models.** For each of the datasets, there
 424 is a limited set of valid answers. Pretrained models often
 425 struggle to shift their probability mass based on instruc-
 426 tions in a zero-shot manner to only cover the valid output
 427 distribution, achieving $\sim 50\%$ coverage in our evaluation.
 428 In contrast, SPECTRUM TUNING achieves $> 90\%$ coverage, nearly matching the instruction-tuned
 429 model coverage (Fig 5).



430 Figure 5: Valid answer coverage (\uparrow).
 431

²Machine Personality Inventory (Jiang et al., 2023), Rotten Tomatoes (u/Business-Platform301, 2024), NY-Times Books (Meister et al., 2024), GlobalOQA (Durmus et al., 2023), Urn (ours, new contribution), Habermas (Tessler et al., 2024), Number Game (Bigelow & Piantadosi, 2016; Tenenbaum, 1999).

432	Ablation Components						ICL Steerability			Dist. Align.		Valid Output Coverage		
	Abl.	Weight	Special Tokens	Train on	# Train	Loss only	MC Loss	MC Acc	Free-text	Dist. Align.	Yield -	Yield -	Yield -	
	#	Init	Embedding	SPECTRUM SUITE	Seqs	Outputs	(Norm.)	(Norm.)	Loss (Norm.)	JS-Div.	3-shot	3-shot	+ Description	
<i>A - Default: 1) Spectrum Tuning, 2) Pretrained, and 3) Instruction-Tuned</i>														
434	1	PT	IT	✓	38.8k	✓	1.00	1.00	1.00	.069	36.7	42.1	49.2	
435	2	PT	-	✗ (PT prompt)	-	-	<u>1.19</u>	<u>0.99</u>	<u>1.00</u>	<u>.083</u>	5.8	<u>37.2</u>	<u>44.2</u>	
436	3	IT	IT	✗ (IT prompt)	-	-	2.62	0.98	1.30	.228	<u>11.7</u>	21.5	20.7	
<i>B - Training method ablations: 1) Default; 4) Loss only first output (Instruct-SFT on S-Suite); 5) Loss only last output (Meta-ICL on S-Suite); 6) Loss on all tokens (S-Suite)</i>														
437	1	PT	IT	✓	38.8k	✓	1.00	1.00	1.00	.069	36.7	42.1	49.2	
438	4	PT	IT	✓	38.8k	first only	1.03	1.00	1.01	.067	37.9	33.0	44.0	
439	5	PT	IT	✓	38.8k	last only	1.02	0.99	1.00	.103	17.1	35.4	39.6	
440	6	PT	IT	✓	38.8k	✗	<u>1.01</u>	0.98	1.00	.075	33.0	40.6	47.1	
<i>C - Data ablation: 7) Train only on capability / knowledge elicitation data, 8) Train on Spectrum Suite, data size matched to capability data</i>														
441	7	PT	IT	✗ (capability data)	3.9k	✓	1.03	0.99	1.02	.111	12.7	21.2	39.5	
442	8	PT	IT	✓	3.9k	✓	1.03	1.00	1.01	.086	21.8	35.5	40.8	
<i>D - Weight Init Ablation: Spectrum Tuning with 1) Default weight init; 9) PT init, bracket as special token embed, 10) PT init, random special token embed, 11) IT init</i>														
443	1	PT	IT	✓	38.8k	✓	1.00	1.00	1.00	.069	36.7	42.1	49.2	
444	9	PT	<>/> (PT)	✓	38.8k	✓	1.43	1.03	<u>1.02</u>	.063	28.0	30.0	33.0	
445	10	PT	Random	✓	38.8k	✓	1.44	0.87	1.25	.079	21.0	21.0	26.4	
446	11	IT	IT	✓	38.8k	✓	1.08	<u>1.02</u>	1.05	.069	<u>33.4</u>	<u>42.0</u>	<u>45.2</u>	

Table 4: Ablations, averaged across models and tasks. Shaded rows are default Spectrum-Tuned results. We show averaged results for A) the default setup, B) training on SPECTRUM SUITE with different methods, C) training on capability-focused data in place of SPECTRUM SUITE, and D) different model weight initializations. Best result within each ablation is bolded, and second best is underlined. ICL Steerability results are normalized to the default configuration.

7 ABLATIONS AND GENERAL CAPABILITIES

In Table 4, we ablate parts of SPECTRUM TUNING in order to further disentangle the effect of each component. We report averaged results for all three desiderata across all models and tasks. In A), we see the normalized data from the prior sections, illustrating Spectrum-Tuned models improvements over base and default instruct models.

SPECTRUM SUITE’s selective loss is important for performance on all desiderata. In B), we hold the Spectrum Tuning data constant, and ablate the training method. We compare against training on the first output only (similar to Instruct-SFT),³ training on the last output only (similar to MetaICL, Min et al. 2022a), and calculating loss on all tokens, including `description`/`inputs`. We find that training on the first output only causes a degradation in few-shot learning capabilities (ICL loss, few-shot yield), and training on the last output only causes across the board degradation, especially on zero-shot tasks (distributional alignment, `description` yield). Training on all tokens (including `description`/`input`) leads to slight degradations across the board.

Training on capability-focused data only underperforms training on SPECTRUM SUITE. We train on a subset of data in the same format as SPECTRUM SUITE, but focused on capability data instead of data requiring steerability (Table 4, C). We find that including the SPECTRUM SUITE data is important for eliciting the desiderata. Finally, we find that D) the default weight initialization (PT model weights, IT special token embeddings) overall elicits the best performance, although initializing the special tokens with bracket token embeddings seems to improve the multiple-choice accuracy and distributional alignment.

While the default recipe offers strong performance, future work could i) further optimize hyperparameters (as we have done limited optimization),⁴ ii) reduce reliance on initializing the special tokens from IT models, and iii) probe which data is most important in eliciting gains.

SPECTRUM TUNING does not harm general model capabilities. Lastly, we evaluate whether our method affects general model capabilities. While we do not necessarily expect our method to improve upon standard evaluations where there is a single correct answer, we want to understand if it degrades performance compared to pretrained models. While we find that Spectrum-Tuned models generally perform worse than instruction-tuned models at these tasks (as expected), we find

³However, we also consider this distinct from traditional instruction-tuning, as the focus is on fitting the data generation task of the `description` as opposed to generating a helpful chat assistant response.

⁴In fact, after running the main suite of experiments, we suspected that our models were somewhat underfit. We found that simply reducing the batch size resulted in significant gains in distributional alignment and yield (see App. G for more details). We believe that this illustrates exciting opportunities for further optimization and improvements to improve performance—the performance ceiling has not been hit.

486 that Spectrum-Tuned models have similar performance to the pretrained models on which they are
 487 based. In other words, we see no evidence of harm to general capabilities with SPECTRUM TUNING.
 488 For more details, see Appendix C.5
 489

490 8 RELATED WORK 491

492 **Diversity, distributional alignment, and steerability.** Several other works have documented di-
 493 versity collapse in LLMs (Shumailov et al., 2023; Dohmatob et al., 2024; Yang et al., 2024; Zhang
 494 et al., 2024a; Li et al., 2024; West & Potts, 2025), often linking it to alignment (Murthy et al.,
 495 2024; Kirk et al., 2024a; 2023) or insufficient training data diversity (Chen et al., 2024). Potential
 496 consequences of diversity collapse include reduced creativity, loss of minority perspectives, spread
 497 of bias, and overall decline in model utility and trustworthiness (Anderson et al., 2024; Kapania
 498 et al., 2024). Distributional alignment has been explored by a few prior works (Meister et al., 2024;
 499 Durmus et al., 2023; Sorensen et al., 2024b), but literature here is far less developed. Additionally,
 500 other works have focused on measuring steerability to system messages (Lee et al., 2024), persona
 501 descriptions (Miehling et al., 2025; Castricato et al., 2024), and values or attributes (Sorensen et al.,
 502 2024b; 2025). Our work builds on these directions by generalizing steerability to include any in-
 503 context information, including examples, and evaluating on a broader swath of distributions.
 504

505 **Pluralistic alignment and integrating disagreement into LLMs.** Many have recently challenged
 506 the idea of a single ground truth (Aroyo et al., 2023; Basile et al., 2021; Gordon et al., 2022). Plu-
 507 ralistic alignment (Sorensen et al., 2024b; Kirk et al., 2024b) is concerned with integrating diverse
 508 values and perspectives directly into the alignment process. Steerability in particular is related to
 509 user fairness and ensuring that AI systems are usable by diverse stakeholders (Alamdar et al., 2024).
 510

511 **Related Methods** Zhang et al. (2024a) found that training on samples from diffuse distributions
 512 helps LLMs to avoid mode collapse, and served as inspiration for some experiments. SPECTRUM
 513 TUNING is similar in spirit, but also includes in-context samples and leverages orders of magnitude
 514 more data. Entropy maximization in finetuning can help increase diversity (Li et al., 2025). MetaICL
 515 (Min et al., 2022a) uses in-context examples as in our method, but focuses on NLP datasets with
 516 a single ground truth and only trains on the last example. Centaur (Binz et al., 2024) similarly
 517 modifies cross-entropy loss to only focus on tokens of interest, but focuses on a different data distri-
 518 bution (cognitive-science human experiments). Some very recent works have somewhat improved
 519 the diversity/validity Pareto frontier by adding some sort of diversity regularization to preference
 520 optimization or RL reward (Lanchantin et al., 2025; Chung et al., 2025; Li et al., 2025). Finally,
 521 several recent papers have found that prompting instruct models for multiple samples in-context can
 522 help to mitigate mode collapse (Zhang et al., 2025a;b;d).
 523

524 9 DISCUSSION AND CONCLUSION 525

526 We have outlined three desiderata for conditional distributional modeling with LLMs: in-context
 527 steerability, output space coverage, and distributional alignment, and shown across three model
 528 families that current post-training can systematically hurt these properties. These results have
 529 implications for user steerability—e.g., when possible, pretrained models may be preferred over
 530 instruction-tuned models when steering to a particular user in a well-calibrated way is important.⁵
 531 In addition, we have introduced SPECTRUM SUITE and SPECTRUM TUNING, a resource and post-
 532 training method for enhancing these desiderata. Models trained with SPECTRUM TUNING usually
 533 match or exceed their pretrained counterparts at these properties—to our knowledge, ours is the
 534 first method to improve upon pretrained models at distributional alignment or in-context steerabil-
 535 ity. However, much work remains. Promising directions for future work include 1) exploring which
 536 data is most important for eliciting the desiderata; 2) further characterizing why and how instruction-
 537 tuning hurts in-context steerability; 3) more work to combine the strengths of instruction-tuned mod-
 538 els and SPECTRUM TUNING models (e.g., Zhu et al. 2025);⁶ and 4) scaling SPECTRUM TUNING to
 539 larger models and more data.
 540

541 ⁵However, access to the pretrained model is restricted in many proprietary cases. This illustrates a gap: Can
 542 companies offer very steerable and distributionally-aligned models, while maintaining safety constraints?
 543

544 ⁶On the other hand, it is possible that top-1 chat performance and our desiderata are so fundamentally in
 545 tension, that we may need to specialize models to either top-1 chat performance or our desiderata, and select
 546 the appropriate model for each use case or combine strengths at inference (e.g., Zhu et al. 2025)
 547

540
541
ETHICS STATEMENT542
543
544
545
546
547
548
549
In this paper, we seek to enable AI systems that can work for a variety of perspectives and estimate
human preferences and opinions in a well-calibrated manner. We believe that these are net positive
developments, allowing AI systems to work properly for more people. Additionally, well-calibrated
human preferences may be especially important as AI systems are used agentically - it will be
important that an agent have a good model of what the user wants, as opposed to a modal preference.
Calibration, where current instruction-tuned systems really struggle, can be especially important for
agents to safely act autonomously when they are (properly) very confident about a users' preference,
and ask for direction when they are less confident.550
551
552
553
554
555
556
With **SPECTRUM SUITE**, we perform experiments on several datasets which may include personal
information such as demographics. However, all included datasets are anonymized, we attempt to
use the data only in line with their intended use, and we do not distribute the underlying datasets in
SPECTRUM SUITE directly. Instead, we refer people interested in extending our work to the original
data sources, and provide only the code to unify the data into the *description/input/output*
format. Because of this, we believe that our compilation of **SPECTRUM SUITE** does not pose an
additional privacy risk.557
558
REPRODUCIBILITY STATEMENT
559560
561
562
563
564
565
566
We have attempted to ensure that every portion of the paper is reproducible, and release code con-
taining: **SPECTRUM SUITE** construction, including processing and pointers to hydrate each dataset;
SPECTRUM TUNING training code; and code for running all evaluations. We also release the weights
for all trained **SPECTRUM TUNING** models. We include additional training details on hardware and
hyperparameters used in App. F and additional experimental details in App. L, M, N. In App. O, we
show demonstrative example prompts for each test task in **SPECTRUM SUITE** and include example
prompts for remaining tasks in supplementary materials.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 REFERENCES
595

596 Parand A. Alamdari, Toryn Q. Klassen, Rodrigo Toro Icarte, and Sheila A. McIlraith. Being
597 considerate as a pathway towards pluralistic alignment for agentic ai, 2024. URL <https://arxiv.org/abs/2411.10613>.

599 Barrett R Anderson, Jash Hemant Shah, and Max Kreminski. Homogenization effects of large
600 language models on human creative ideation. In *Proceedings of the 16th conference on creativity*
601 & cognition, pp. 413–425, 2024.

602 Lora Aroyo, Alex S. Taylor, Mark Diaz, Christopher M. Homan, Alicia Parrish, Greg Serapio-
603 Garcia, Vinodkumar Prabhakaran, and Ding Wang. Dices dataset: Diversity in conversational ai
604 evaluation for safety, 2023. URL <https://arxiv.org/abs/2306.11247>.

606 Akari Asai, Sneha Kudugunta, Xinyan Yu, Terra Blevins, Hila Gonen, Machel Reid, Yulia Tsvetkov,
607 Sebastian Ruder, and Hannaneh Hajishirzi. BUFFET: Benchmarking large language models for
608 few-shot cross-lingual transfer. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Pro-
609 ceedings of the 2024 Conference of the North American Chapter of the Association for Compu-
610 tational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1771–1800,
611 Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/
612 2024.naacl-long.100. URL <https://aclanthology.org/2024.naacl-long.100/>.

613 Valerio Basile, Michael Fell, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, Massimo
614 Poesio, and Alexandra Uma. We need to consider disagreement in evaluation. In Kenneth Church,
615 Mark Liberman, and Valia Kordoni (eds.), *Proceedings of the 1st Workshop on Benchmarking:
616 Past, Present and Future*, pp. 15–21, Online, August 2021. Association for Computational Lin-
617 guistics. doi: 10.18653/v1/2021.bppf-1.3. URL [https://aclanthology.org/2021.
618 bppf-1.3/](https://aclanthology.org/2021.bppf-1.3/).

619 Mehar Bhatia, Shravan Nayak, Gaurav Kamath, Marius Mosbach, Karolina Stańczak, Vered
620 Schwartz, and Siva Reddy. Value drifts: Tracing value alignment during llm post-training, 2025.
621 URL <https://arxiv.org/abs/2510.26707>.

622 Eric Bigelow and Steven T. Piantadosi. A large dataset of generalization patterns in the number
623 game. *Journal of Open Psychology Data*, 4(1):e4, 2016. doi: 10.5334/jopd.19. URL <https://openpsychologydata.metajnl.com/articles/10.5334/jopd.19/>. Published
624 2016-03-18; accessed 2025-09-21.

626 Marcel Binz, Elif Akata, Matthias Bethge, Franziska Brändle, Fred Callaway, Julian Coda-Forno,
627 Peter Dayan, Can Demircan, Maria K. Eckstein, Noémi Éltető, Thomas L. Griffiths, Susanne
628 Haridi, Akshay K. Jagadish, Li Ji-An, Alexander Kipnis, Sreejan Kumar, Tobias Ludwig, Marvin
629 Mathony, Marcelo Mattar, Alireza Modirshanechi, Surabhi S. Nath, Joshua C. Peterson, Milena
630 Rmus, Evan M. Russek, Tankred Saanum, Natalia Scharfenberg, Johannes A. Schubert, Luca
631 M. Schulze Buschoff, Nishad Singhi, Xin Sui, Mirko Thalmann, Fabian Theis, Vuong Truong,
632 Vishaal Udandarao, Konstantinos Voudouris, Robert Wilson, Kristin Witte, Shuchen Wu, Dirk
633 Wulff, Huadong Xiong, and Eric Schulz. Centaur: a foundation model of human cognition, 2024.
634 URL <https://arxiv.org/abs/2410.20268>.

635 Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhari-
636 wal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal,
637 Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M.
638 Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz
639 Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec
640 Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL
641 <https://arxiv.org/abs/2005.14165>.

642 Silvia Casola, Simona Frenda, Soda Marem Lo, Erhan Sezerer, Antonio Uva, Valerio Basile,
643 Cristina Bosco, Alessandro Pedrani, Chiara Rubagotti, Viviana Patti, and Davide Bernardi. Multi-
644 PICo: Multilingual perspectivist irony corpus. In Lun-Wei Ku, Andre Martins, and Vivek
645 Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Compu-
646 tational Linguistics (Volume 1: Long Papers)*, pp. 16008–16021, Bangkok, Thailand, August
647 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.849. URL
648 <https://aclanthology.org/2024.acl-long.849/>.

648 Louis Castricato, Nathan Lile, Rafael Rafailov, Jan-Philipp Fränken, and Chelsea Finn. Persona:
 649 A reproducible testbed for pluralistic alignment, 2024. URL <https://arxiv.org/abs/2407.17387>.
 650

651 Hao Chen, Abdul Waheed, Xiang Li, Yidong Wang, Jindong Wang, Bhiksha Raj, and Marah I.
 652 Abdin. On the diversity of synthetic data and its impact on training large language models, 2024.
 653 URL <https://arxiv.org/abs/2410.15226>.
 654

655 John Joon Young Chung, Vishakh Padmakumar, Melissa Roemmele, Yuqian Sun, and Max
 656 Kreminski. Modifying large language model post-training for diverse creative writing, 2025.
 657 URL <https://arxiv.org/abs/2503.17126>.
 658

659 Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and
 660 Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge,
 661 2018. URL <https://arxiv.org/abs/1803.05457>.
 662

663 Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser,
 664 Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John
 665 Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*,
 666 2021.
 667

668 Elvis Dohmatob, Yunzhen Feng, Arjun Subramonian, and Julia Kempe. Strong model collapse.
 669 *arXiv preprint arXiv:2410.04840*, 2024.
 670

671 Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu,
 672 Zhiyong Wu, Tianyu Liu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-
 673 context learning, 2024. URL <https://arxiv.org/abs/2301.00234>.
 674

675 Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner.
 676 Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs, 2019.
 677 URL <https://arxiv.org/abs/1903.00161>.
 678

679 Yann Dubois, Percy Liang, and Tatsunori Hashimoto. Length-controlled alpacaeval: A simple
 680 debiasing of automatic evaluators. In *First Conference on Language Modeling*, 2024. URL
 681 <https://openreview.net/forum?id=CybBmzWBX0>.
 682

683 Esin Durmus, Karina Nyugen, Thomas I. Liao, Nicholas Schiefer, Amanda Askell, Anton Bakhtin,
 684 Carol Chen, Zac Hatfield-Dodds, Danny Hernandez, Nicholas Joseph, Liane Lovitt, Sam McCandlish,
 685 Orowa Sikder, Alex Tamkin, Janel Thamkul, Jared Kaplan, Jack Clark, and Deep Ganguli.
 686 Towards measuring the representation of subjective global opinions in language models, 2023.
 687

688 EVS/WVS. European values study and world values survey: Joint evs/wvs 2017–2022
 689 dataset, 2024. URL <https://www.gesis.org/en/european-values-study/data-and-documentation/joint-evs/wvs-2017-2022-dataset>. Identical ver-
 690 sion also via WVS site with DOI 10.14281/18241.26; accessed 2025-09-21.
 691

692 Sara Fish, Paul Götz, David C. Parkes, Ariel D. Procaccia, Gili Rusak, Itai Shapira, and Manuel
 693 Wüthrich. Generative social choice, 2025. URL <https://arxiv.org/abs/2309.01291>.
 694

695 Mitchell L. Gordon, Michelle S. Lam, Joon Sung Park, Kayur Patel, Jeff Hancock, Tatsunori
 696 Hashimoto, and Michael S. Bernstein. Jury learning: Integrating dissenting voices into machine
 697 learning models. In *CHI Conference on Human Factors in Computing Systems*, CHI '22, pp.
 698 1–19. ACM, April 2022. doi: 10.1145/3491102.3502004. URL <http://dx.doi.org/10.1145/3491102.3502004>.
 699

700 Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad
 701 Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd
 702 of models. *arXiv preprint arXiv:2407.21783*, 2024.
 703

704 Yuling Gu, Oyvind Tafjord, Bailey Kuehl, Dany Haddad, Jesse Dodge, and Hannaneh Hajishirzi.
 705 OLMES: A standard for language model evaluations. In Luis Chiruzzo, Alan Ritter, and Lu Wang
 706 (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 5005–5033,
 707 Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-
 708 89176-195-7. doi: 10.18653/v1/2025.findings-naacl.282. URL <https://aclanthology.org/2025.findings-naacl.282/>.
 709

702 Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Ja-
 703 cob Steinhardt. Measuring massive multitask language understanding, 2021. URL <https://arxiv.org/abs/2009.03300>.

704

705 Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural
 706 networks: A survey, 2020. URL <https://arxiv.org/abs/2004.05439>.

707

708 Hyewon Jang and Diego Frassinelli. Generalizable sarcasm detection is just around the corner,
 709 of course! In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024*
 710 *Conference of the North American Chapter of the Association for Computational Linguistics: Human*
 711 *Language Technologies (Volume 1: Long Papers)*, pp. 4238–4249, Mexico City, Mexico,
 712 June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.238.
 713 URL <https://aclanthology.org/2024.naacl-long.238/>.

714

715 Ziwei Ji, Justin D. Li, and Matus Telgarsky. Early-stopped neural networks are consistent, 2021.
 716 URL <https://arxiv.org/abs/2106.05932>.

717

718 Guangyuan Jiang, Manjie Xu, Song-Chun Zhu, Wenjuan Han, Chi Zhang, and Yixin Zhu. Evalu-
 719 ating and inducing personality in pre-trained language models, 2023. URL <https://arxiv.org/abs/2206.07550>.

720

721 Shivani Kapania, William Agnew, Motahhare Eslami, Hoda Heidari, and Sarah Fox. 'simulacrum
 722 of stories': Examining large language models as qualitative research participants. *arXiv preprint*
 723 *arXiv:2409.19430*, 2024.

724

725 Hannah Rose Kirk, Bertie Vidgen, Paul Röttger, and Scott A Hale. The benefits, risks and bounds of
 726 personalizing the alignment of large language models to individuals. *Nature Machine Intelligence*,
 727 6(4):383–392, 2024a.

728

729 Hannah Rose Kirk, Alexander Whitefield, Paul Röttger, Andrew Bean, Katerina Margatina, Juan
 730 Ciro, Rafael Mosquera, Max Bartolo, Adina Williams, He He, Bertie Vidgen, and Scott A. Hale.
 731 The prism alignment dataset: What participatory, representative and individualised human feed-
 732 back reveals about the subjective and multicultural alignment of large language models, 2024b.
 733 URL <https://arxiv.org/abs/2404.16019>.

734

735 Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward
 736 Grefenstette, and Roberta Raileanu. Understanding the effects of rlhf on llm generalisation and
 737 diversity. *arXiv preprint arXiv:2310.06452*, 2023.

738

739 G. Kokolakis. Bayesian statistical analysis. In Penelope Peterson, Eva Baker, and Barry Mc-
 740 Gaw (eds.), *International Encyclopedia of Education (Third Edition)*, pp. 37–45. Elsevier, Ox-
 741 ford, third edition edition, 2010. ISBN 978-0-08-044894-7. doi: <https://doi.org/10.1016/B978-0-08-044894-7.01308-7>. URL <https://www.sciencedirect.com/science/article/pii/B9780080448947013087>.

742

743 Nikolay Kolyada, Khalid Al-Khatib, Michael Völske, Shahbaz Syed, and Benno Stein. Webis
 744 changemyview corpus 2020 (webis-cmv-20), 2020. URL <https://doi.org/10.5281/zenodo.3778298>. Version v1; file used: threads.jsonl. Accessed 2025-09-21.

745

746 Deepak Kumar, Patrick Gage Kelley, Sunny Consolvo, Joshua Mason, Elie Bursztein, Zakir Du-
 747 rumeric, Kurt Thomas, and Michael Bailey. Designing toxic content classification for a diversity
 748 of perspectives, 2021. URL <https://arxiv.org/abs/2106.04511>.

749

750 Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brah-
 751 man, Lester James Validad Miranda, Alisa Liu, Nouha Dziri, Xinxi Lyu, Yuling Gu, Saumya
 752 Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Christo-
 753 pher Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Ha-
 754 jishirzi. Tulu 3: Pushing frontiers in open language model post-training. In *Second Conference on*
 755 *Language Modeling*, 2025. URL <https://openreview.net/forum?id=iluGbfHHpH>.

756

757 Jack Lanchantin, Angelica Chen, Shehzaad Dhuliawala, Ping Yu, Jason Weston, Sainbayar
 758 Sukhbaatar, and Ilia Kulikov. Diverse preference optimization, 2025. URL <https://arxiv.org/abs/2501.18101>.

756 Seongyun Lee, Sue Hyun Park, Seungone Kim, and Minjoon Seo. Aligning to thousands of preferences via system message generalization, 2024. URL <https://arxiv.org/abs/2405.17977>.

757

758

759

760 Elisa Leonardelli, Silvia Casola, Siyao Peng, Giulia Rizzi, Valerio Basile, Elisabetta Fersini, Diego Frassinelli, Hyewon Jang, Maja Pavlovic, Barbara Plank, and Massimo Poesio. Lewidi-2025 at nlperspectives: third edition of the learning with disagreements shared task. In *Proceedings of the 4th Workshop on Perspectivist Approaches to NLP (NLPerspectives)*. Association for Computational Linguistics, nov 2025.

761

762

763

764

765 Margaret Li, Weijia Shi, Artidoro Pagnoni, Peter West, and Ari Holtzman. Predicting vs. acting: A trade-off between world modeling & agent modeling. *arXiv preprint arXiv:2407.02446*, 2024.

766

767 Tianjian Li, Yiming Zhang, Ping Yu, Swarnadeep Saha, Daniel Khashabi, Jason Weston, Jack Lan- chantin, and Tianlu Wang. Jointly reinforcing diversity and quality in language model generations, 2025. URL <https://arxiv.org/abs/2509.02534>.

768

769

770

771 Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. The unlocking spell on base llms: Rethinking alignment via in-context learning, 2023. URL <https://arxiv.org/abs/2312.01552>.

772

773

774 Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, Dublin, Ireland, May 2022a. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL <https://aclanthology.org/2022.acl-long.229>.

775

776

777

778

779

780 Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods, 2022b. URL <https://arxiv.org/abs/2109.07958>.

781

782

783 Alisa Liu, Zhaofeng Wu, Julian Michael, Alane Suhr, Peter West, Alexander Koller, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. We’re afraid language models aren’t modeling ambiguity. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 790–807, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.51. URL <https://aclanthology.org/2023.emnlp-main.51>.

784

785

786

787

788

789 Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL <http://www.aclweb.org/anthology/P11-1015>.

790

791

792

793

794 Nicole Meister, Carlos Guestrin, and Tatsunori Hashimoto. Benchmarking distributional alignment of large language models, 2024. URL <https://arxiv.org/abs/2411.05403>.

795

796

797 Erik Miehling, Michael Desmond, Karthikeyan Natesan Ramamurthy, Elizabeth M. Daly, Pierre Dognin, Jesus Rios, Djallel Bouneffouf, and Miao Liu. Evaluating the prompt steerability of large language models, 2025. URL <https://arxiv.org/abs/2411.12405>.

798

799

800 Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. Metaicl: Learning to learn in context, 2022a. URL <https://arxiv.org/abs/2110.15943>.

801

802

803 Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work?, 2022b. URL <https://arxiv.org/abs/2202.12837>.

804

805

806 Jared Moore, Tanvi Deshpande, and Diyi Yang. Are large language models consistent over value- laden questions?, 2024. URL <https://arxiv.org/abs/2407.02996>.

807

808

809 mstz. Titanic (survival) — hugging face dataset, 2023. URL <https://huggingface.co/datasets/mstz/titanic>. Subset: survival; 891 rows; accessed 2025-09-21.

810 Sonia K Murthy, Tomer Ullman, and Jennifer Hu. One fish, two fish, but not the whole sea: Alignment
 811 reduces language models' conceptual diversity. *arXiv preprint arXiv:2411.04427*, 2024.

812

813 Jeremy Neiman. Generating haiku with deep learning. *Towards Data Science*, December 2018. URL
 814 [https://towardsdatascience.com/](https://towardsdatascience.com/generating-haiku-with-deep-learning-dbf5d18b4246/)
 815 generating-haiku-with-deep-learning-dbf5d18b4246/. Accessed 2025-09-21.

816

817

818 Netflix, Inc. Netflix prize data, 2009. URL <https://www.kaggle.com/datasets/netflix-inc/netflix-prize-data>. Dataset from the Netflix Prize competition; ac-
 819 cessed 2025-09-21.

820

821

822 OpenAI. Collective alignment 1: Public input on model defaults (version 1.0). <https://huggingface.co/datasets/openai/collective-alignment-1>, 2025. Dataset; ac-
 823 cessed 2025-09-21.

824

825 OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Floren-
 826 cia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red
 827 Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Moham-
 828 mad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher
 829 Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brock-
 830 man, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann,
 831 Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis,
 832 Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey
 833 Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux,
 834 Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila
 835 Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix,
 836 Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gib-
 837 son, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan
 838 Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hal-
 839 lacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan
 840 Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu,
 841 Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun
 842 Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Ka-
 843 mali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook
 844 Kim, Christina Kim, Yongjik Kim, Jan Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel
 845 Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen
 846 Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel
 847 Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez,
 848 Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv
 849 Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney,
 850 Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick,
 851 Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel
 852 Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Ra-
 853 jeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe,
 854 Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel
 855 Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe
 856 de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny,
 857 Michelle Pokrass, Vitchyr H. Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl,
 858 Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra
 859 Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders,
 860 Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Sel-
 861 sam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor,
 862 Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky,
 863 Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang,
 864 Nikolas Tezak, Madeleine B. Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston
 865 Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya,
 866 Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward,
 867 Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng,

864 Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Work-
 865 man, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming
 866 Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao
 867 Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2024. URL
 868 <https://arxiv.org/abs/2303.08774>.

869 Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong
 870 Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to fol-
 871 low instructions with human feedback. *Advances in neural information processing systems*, 35:
 872 27730–27744, 2022.

873 Jiaxin Pei and David Jurgens. When do annotator demographics matter? measuring the influence of
 874 annotator demographics with the popquorn dataset, 2023. URL <https://arxiv.org/abs/2306.06826>.

875 Linlu Qiu, Fei Sha, Kelsey Allen, Yoon Kim, Tal Linzen, and Sjoerd van Steenkiste. Bayesian teach-
 876 ing enables probabilistic reasoning in large language models, 2025. URL <https://arxiv.org/abs/2503.17523>.

877 Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and
 878 Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model,
 879 2024. URL <https://arxiv.org/abs/2305.18290>.

880 Samyam Rajbhandari, Olatunji Ruwase, Jeff Rasley, Shaden Smith, and Yuxiong He. Zero-infinity:
 881 Breaking the gpu memory wall for extreme scale deep learning, 2021. URL <https://arxiv.org/abs/2104.07857>.

882 Mayk Caldas Ramos, Shane S. Michtavy, Marc D. Porosoff, and Andrew D. White. Bayesian
 883 optimization of catalysts with in-context learning, 2023.

884 David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien
 885 Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a
 886 benchmark, 2023. URL <https://arxiv.org/abs/2311.12022>.

887 David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien
 888 Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a
 889 benchmark. In *First Conference on Language Modeling*, 2024. URL <https://openreview.net/forum?id=Ti67584b98>.

890 Paul Röttger, Musashi Hinck, Valentin Hofmann, Kobi Hackenburg, Valentina Pyatkin, Faeze Brah-
 891 man, and Dirk Hovy. Issuebench: Millions of realistic prompts for measuring issue bias in llm
 892 writing assistance, 2025. URL <https://arxiv.org/abs/2502.08395>.

893 Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: an adver-
 894 sarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106, August 2021. ISSN
 895 0001-0782. doi: 10.1145/3474381. URL <https://doi.org/10.1145/3474381>.

896 Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, An-
 897 toine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen
 898 Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chh-
 899 ablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo
 900 Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang,
 901 Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan
 902 Fries, Ryan Teehan, Tali Bers, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M.
 903 Rush. Multitask prompted training enables zero-shot task generalization, 2022. URL <https://arxiv.org/abs/2110.08207>.

904 Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cinoo Lee, Percy Liang, and Tatsunori Hashimoto.
 905 Whose opinions do language models reflect? *arXiv preprint arXiv:2303.17548*, 2023.

906 Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Yarin Gal, Nicolas Papernot, and Ross Ander-
 907 son. The curse of recursion: Training on generated data makes models forget. *arXiv preprint*
 908 *arXiv:2305.17493*, 2023.

918 Social Security Administration. Baby names from social security card applications — national
 919 data. <https://www.ssa.gov/oact/babynames/limits.html>, 2025. Data are from
 920 a 100% sample of Social Security card applications; names with ≤ 5 occurrences are suppressed.
 921 Accessed 2025-09-21.

922 Taylor Sorensen, Liwei Jiang, Jena D. Hwang, Sydney Levine, Valentina Pyatkin, Peter West, Nouha
 923 Dziri, Ximing Lu, Kavel Rao, Chandra Bhagavatula, Maarten Sap, John Tasioulas, and Yejin
 924 Choi. Value kaleidoscope: Engaging ai with pluralistic human values, rights, and duties. *Pro-
 925 ceedings of the AAAI Conference on Artificial Intelligence*, 38(18):19937–19947, March 2024a.
 926 ISSN 2159-5399. doi: 10.1609/aaai.v38i18.29970. URL <http://dx.doi.org/10.1609/aaai.v38i18.29970>.

927 Taylor Sorensen, Jared Moore, Jillian Fisher, Mitchell Gordon, Niloofar Mireshghallah, Christo-
 928 pher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, Tim Althoff, and Yejin
 929 Choi. A roadmap to pluralistic alignment, 2024b. URL <https://arxiv.org/abs/2402.05070>.

930 Taylor Sorensen, Pushkar Mishra, Roma Patel, Michael Henry Tessler, Michiel Bakker, Georgina
 931 Evans, Iason Gabriel, Noah Goodman, and Verena Rieser. Value profiles for encoding human
 932 variation, 2025. URL <https://arxiv.org/abs/2503.15484>.

933 Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung,
 934 Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. Challenging BIG-bench
 935 tasks and whether chain-of-thought can solve them. In Anna Rogers, Jordan Boyd-Graber,
 936 and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL*
 937 2023, pp. 13003–13051, Toronto, Canada, July 2023. Association for Computational Linguis-
 938 tics. doi: 10.18653/v1/2023.findings-acl.824. URL <https://aclanthology.org/2023.findings-acl.824>.

939 Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej,
 940 Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas
 941 Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Cas-
 942 bon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xi-
 943 aohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Cole-
 944 man, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry,
 945 Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi,
 946 Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe
 947 Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa
 948 Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András
 949 György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia
 950 Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini,
 951 Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel
 952 Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivaku-
 953 mar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eu-
 954 gene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna
 955 Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian
 956 Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wi-
 957 eting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh,
 958 Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine,
 959 Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael
 960 Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Ni-
 961 lay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Ruben-
 962 stein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya
 963 Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu,
 964 Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti
 965 Sheth, Siim Pöder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi
 966 Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry,
 967 Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein
 968 Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat
 969 Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas
 970

972 Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Bar-
 973 ral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam
 974 Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena
 975 Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier
 976 Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot.
 977 Gemma 3 technical report, 2025. URL <https://arxiv.org/abs/2503.19786>.

978 Joshua Tenenbaum. *A Bayesian Framework for Concept Learning*. PhD thesis, Massachusetts
 979 Institute of Technology, Cambridge, MA, 1999.

980 Michael Henry Tessler, Michiel A. Bakker, Daniel Jarrett, Hannah Sheahan, Martin J. Chadwick,
 981 Raphael Koster, Georgina Evans, Lucy Campbell-Gillingham, Tantum Collins, David C. Parkes,
 982 Matthew Botvinick, and Christopher Summerfield. Ai can help humans find common ground
 983 in democratic deliberation. *Science*, 386(6719):eadq2852, 2024. doi: 10.1126/science.adq2852.
 984 URL <https://www.science.org/doi/abs/10.1126/science.adq2852>.

985 The Computational Democracy Project. Open polis data. <https://github.com/compdemocracy/openData>, 2025. GitHub repository; data exports from select public Polis
 986 conversations; accessed 2025-09-21.

987 Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailev, Huaxiu Yao, Chelsea
 988 Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated
 989 confidence scores from language models fine-tuned with human feedback, 2023. URL <https://arxiv.org/abs/2305.14975>.

990 Guiyao Tie, Zeli Zhao, Dingjie Song, Fuyang Wei, Rong Zhou, Yurou Dai, Wen Yin, Zhejian Yang,
 991 Jiangyue Yan, Yao Su, Zhenhan Dai, Yifeng Xie, Yihan Cao, Lichao Sun, Pan Zhou, Lifang He,
 992 Hechang Chen, Yu Zhang, Qingsong Wen, Tianming Liu, Neil Zhenqiang Gong, Jiliang Tang,
 993 Caiming Xiong, Heng Ji, Philip S. Yu, and Jianfeng Gao. A survey on post-training of large
 994 language models, 2025. URL <https://arxiv.org/abs/2503.06072>.

995 trexmatt. 200,000+ jeopardy! questions (csv dump from j-archive). https://drive.google.com/file/d/0BwT5wj_P7BKXU19tOUJWYZVvUjA/view?resourcekey=0-uFrn8bQkUfSCvJlmtKGcdQ, 2014. Original announcement on r/datasets; accessed
 996 2025-09-21.

997 u/Business-Platform301. Rotten tomatoes movies 1970–2024. <https://drive.google.com/file/d/12IpMERb4j83h5gGTdTpV0WZOf5ceY7b3/view>, 2024.
 998 Archive: https://www.reddit.com/r/datasets/comments/1ecj6m2/dataset_for_rotten_tomatoes_movies_1970_2024/. Accessed 2025-09-23.

999 Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Liyuan Liu, Baolin Peng, Hao Cheng, Xuehai
 1000 He, Kuan Wang, Jianfeng Gao, Weizhu Chen, Shuohang Wang, Simon Shaolei Du, and Yelong
 1001 Shen. Reinforcement learning for reasoning in large language models with one training example,
 1002 2025. URL <https://arxiv.org/abs/2504.20571>.

1003 Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming
 1004 Ren, Aaran Arulraj, Xuan He, Ziyian Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang,
 1005 Rongqi Fan, Xiang Yue, and Wenhui Chen. MMLU-pro: A more robust and challenging multi-
 1006 task language understanding benchmark. In *The Thirty-eight Conference on Neural Information
 1007 Processing Systems Datasets and Benchmarks Track*, 2024a. URL <https://openreview.net/forum?id=y10DM6R2r3>.

1008 Zhilin Wang, Alexander Bukharin, Olivier Delalleau, Daniel Egert, Gerald Shen, Jiaqi Zeng, Oleksii
 1009 Kuchaiev, and Yi Dong. Helpsteer2-preference: Complementing ratings with preferences, 2024b.
 1010 URL <https://arxiv.org/abs/2410.01257>.

1011 Leon Weber-Genzel, Siyao Peng, Marie-Catherine De Marneffe, and Barbara Plank. VariErr NLI:
 1012 Separating annotation error from human label variation. In Lun-Wei Ku, Andre Martins, and
 1013 Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Com-
 1014 putational Linguistics (Volume 1: Long Papers)*, pp. 2256–2269, Bangkok, Thailand, August

1026 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.123. URL
 1027 <https://aclanthology.org/2024.acl-long.123/>.

1028

1029 Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du,
 1030 Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners, 2022. URL
 1031 <https://arxiv.org/abs/2109.01652>.

1032 Peter West and Christopher Potts. Base models beat aligned models at randomness and creativity,
 1033 2025. URL <https://arxiv.org/abs/2505.00047>.

1034

1035 Peter West, Ronan Le Bras, Taylor Sorensen, Bill Yuchen Lin, Liwei Jiang, Ximing Lu, Khyathi
 1036 Chandu, Jack Hessel, Ashutosh Baheti, Chandra Bhagavatula, and Yejin Choi. Novacomet: Open
 1037 commonsense foundation models with symbolic knowledge distillation, 2023. URL <https://arxiv.org/abs/2312.05979>.

1038

1039 Andrew Gordon Wilson and Pavel Izmailov. Bayesian deep learning and a probabilistic perspective
 1040 of generalization, 2022. URL <https://arxiv.org/abs/2002.08791>.

1041

1042 Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi,
 1043 Pierrick Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick
 1044 von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger,
 1045 Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface’s transformers: State-of-
 1046 the-art natural language processing, 2020. URL <https://arxiv.org/abs/1910.03771>.

1047

1048 An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang
 1049 Gao, Chengan Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu,
 1050 Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin
 1051 Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang,
 1052 Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui
 1053 Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang
 1054 Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger
 1055 Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan
 1056 Qiu. Qwen3 technical report, 2025. URL <https://arxiv.org/abs/2505.09388>.

1057

1058 Shu Yang, Muhammad Asif Ali, Lu Yu, Lijie Hu, and Di Wang. Model autophagy analysis to expli-
 1059 cate self-consumption within human-AI interactions. In *First Conference on Language Modeling*,
 2024. URL <https://openreview.net/forum?id=FX4fUThO9H>.

1060

1061 Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a ma-
 1062 chine really finish your sentence?, 2019. URL <https://arxiv.org/abs/1905.07830>.

1063

1064 Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. Evaluating
 1065 large language models at evaluating instruction following. In *The Twelfth International Confer-
 1066 ence on Learning Representations*, 2024. URL <https://openreview.net/forum?id=tr0KidwPLc>.

1067

1068 Jiayi Zhang, Simon Yu, Derek Chong, Anthony Sicilia, Michael R. Tomz, Christopher D. Manning,
 1069 and Weiyan Shi. Verbalized sampling: How to mitigate mode collapse and unlock llm diversity,
 2025a. URL <https://arxiv.org/abs/2510.01171>.

1070

1071 Lily Hong Zhang, Smitha Milli, Karen Jusko, Jonathan Smith, Brandon Amos, Wassim, Bouaziz,
 1072 Manon Revel, Jack Kussman, Lisa Titus, Bhaktipriya Radharapu, Jane Yu, Vidya Sarma, Kris
 1073 Rose, and Maximilian Nickel. Cultivating pluralism in algorithmic monoculture: The community
 1074 alignment dataset. *arXiv preprint arXiv: 2507.09650*, 2025b.

1075

1076 Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi
 1077 Hu, Tianwei Zhang, Fei Wu, and Guoyin Wang. Instruction tuning for large language models: A
 1078 survey, 2025c. URL <https://arxiv.org/abs/2308.10792>.

1079 Yiming Zhang, Avi Schwarzschild, Nicholas Carlini, Zico Kolter, and Daphne Ippolito. Forcing
 1080 diffuse distributions out of language models. *arXiv preprint arXiv:2404.10859*, 2024a.

1080 Yiming Zhang, Avi Schwarzschild, Nicholas Carlini, Zico Kolter, and Daphne Ippolito. Forcing dif-
1081 fuse distributions out of language models, 2024b. URL <https://arxiv.org/abs/2404.10859>.

1083 Yiming Zhang, Harshita Diddee, Susan Holm, Hanchen Liu, Xinyue Liu, Vinay Samuel, Barry
1084 Wang, and Daphne Ippolito. Noveltybench: Evaluating language models for humanlike diversity,
1085 2025d. URL <https://arxiv.org/abs/2504.05228>.

1087 Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang,
1088 Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica.
1089 Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.

1090 Alan Zhu, Parth Asawa, Jared Quincy Davis, Lingjiao Chen, Boris Hanin, Ion Stoica, Joseph E.
1091 Gonzalez, and Matei Zaharia. Bare: Leveraging base language models for few-shot synthetic
1092 data generation, 2025. URL <https://arxiv.org/abs/2502.01697>.

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

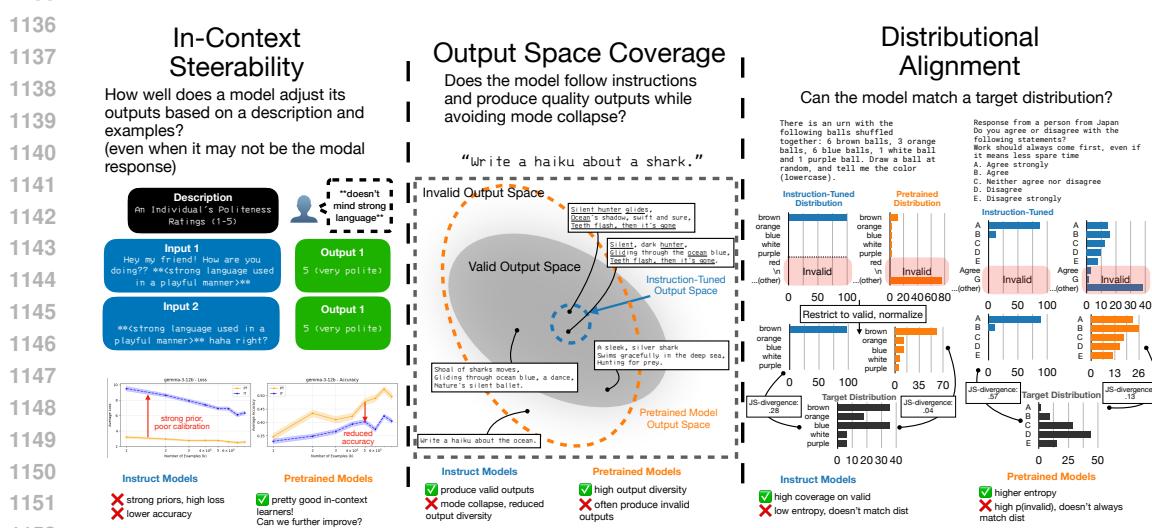
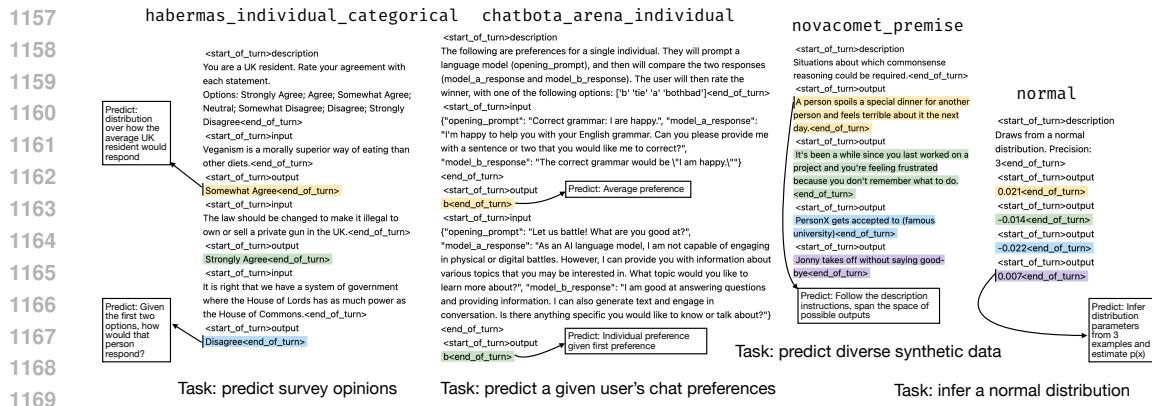
1134 **A SUPPLEMENTARY FIGURES**Figure A1: Three desiderata for conditional distributional modeling. Example outputs and data are drawn from google/gemma-3-12b.

Figure A2: Example tasks from SPECTRUM SUITE in the format used for SPECTRUM TUNING. In our method, we shuffle the data, put it into the above format, and finetune with cross-entropy loss only on the (highlighted) output tokens, including the terminal token.

B FREQUENTLY ASKED QUESTIONS, INTUITIONS, AND HYPOTHESES

Q1: What unifies the three desiderata?

A1: At first glance the desiderata may not seem very related, but they actually all have something in common - they all have to do with tasks where there is not a canonical, single correct answer. Rather, all three desiderata involve either matching or steering to a broad spectrum of potentially valid answers. This is in contrast with the majority of tasks on which we currently train and evaluate instruction-tuned LLMs.

Q2: Why does instruction-tuning post-training lead to spiky distributions and mode collapse?

A2: We have two principal hypotheses for this: 1) the RL objective in RLHF/DPO/GRPO/etc. encourages the model to collapse its distribution to the highest reward output (c.f. West & Potts 2025) and 2) most instruction-tuning training and evaluations focus on tasks with a single verifiable answer. While outside the scope of this work, comparing the desiderata at different stages of

1188 instruction-tuning (e.g., during and after Instruct-SFT, during and after RL) would help to elucidate
 1189 this.⁷

1190 *Q3: It makes sense that SPECTRUM TUNING improves in-context steerability, as it maps easily onto
 1191 the training data format. However, why does Spectrum Tuning improve diversity and distributional
 1192 alignment/calibration?*

1193 A3: While we hope to flesh out our understanding of this mechanism in future work, our best
 1194 intuition is this - It largely has to do with the fact that 1) all training tasks involve interchangeable
 1195 data and 2) we shuffle the data before training. As a simple example, let us consider the
 1196 `diffuse_distribution` task: “Output a random country in Asia, chosen completely at random,
 1197 without replacement.” In training, we collect a list of all countries in Asia, shuffle them, and
 1198 finetune on them as outputs: e.g., “Brunei”, “Lebanon”, “Singapore”, “Laos”, “Vietnam”, ... An
 1199 instruction-tuned model will often exhibit mode collapse - outputting the same country each time.
 1200 Meanwhile, a base model will often output a valid country, but is heavily affected by training data
 1201 frequency / n-gram statistics. In contrast, in the limit, Spectrum Tuning encourages the model to
 1202 actually instantiate a uniform distribution over all countries in Asia - increasing the diversity of
 1203 outputs across many samples. For distributional alignment and calibration, it is a similar story -
 1204 base models are heavily affected by things like n-gram statistics, instruct models have uncalibrated,
 1205 spiky distributions. In contrast, Spectrum Tuning in the limit encourages the model to fit the actual
 1206 described distribution, (partially) overcoming n-gram frequency.

1207 C SPECTRUM SUITE DATA SOURCES

1208 C.1 DATA CONSTRUCTION

1209 As SPECTRUM SUITE is the first-such large-scale resource of such subjective datasets requiring
 1210 steering, it was necessarily constructed in a somewhat ad-hoc manner. However, here we provide
 1211 some general principles for data that we attempted to source:

- 1212 1. Any NLP datasets with corresponding annotator IDs, allowing us to link multiple annotations
 1213 to the same person. We especially sourced from datasets where variation is to be
 1214 expected, as opposed to be eliminated.
- 1215 2. Datasets related to opinion modeling or computational democracy;
- 1216 3. Synthetically-generated NLP datasets;
- 1217 4. Lists of interchangeable things;
- 1218 5. Draws from random distributions;
- 1219 6. Tabular data.

1220 C.2 DATA SOURCES

1221 Below, we cite all data sources used in SPECTRUM SUITE. Additionally, we include any subtask
 1222 names along with the number of sequences included in SPECTRUM SUITE. We release the pro-
 1223 cessing code to go from raw data to our description/input/output in our github repo
 1224 (ANONYMIZED).

1225 Note that many data sources have much more additional data that we could utilize (e.g., OpinionQA
 1226 (Santurkar et al., 2023), Polis (The Computational Democracy Project, 2025), synthetically gener-
 1227 ated random data). We generally restricted each data source to a maximum of 1-2k sequences to
 1228 ensure training data diversity, and in all but a couple of cases with very few data instances (e.g.
 1229 Diffuse Distributions; Zhang et al. 2024b) additionally ensured that the same piece of data was not
 1230 used in more than one sequence.

1231 C.3 TRAIN SPLIT

1232 **Ambient Ambiguity Detection** (Liu et al., 2023)

1233 ⁷For an example of the checkpoint setup one might use, please refer to Bhatia et al. 2025, where they explore
 1234 the effect of post-tuning on value drift.

- 1242 • ambient_ambiguity_detection (50 sequences)
- 1243 • ambient_annotation_distributions (50 sequences)
- 1244 • ambient_disambiguation (50 sequences)
- 1245 • ambient_interpretation_labels (50 sequences)
- 1246 • ambient_linguist_annotations (54 sequences)
- 1247 • ambient_premise_hypothesis (50 sequences)
- 1248
- 1249

1250 **Social Security Administration Baby Names** (Social Security Administration, 2025)

1251

- 1252 • babynames (500 sequences)

1253

1254 **Base-Refine Synthetic Data Generation** (Zhu et al., 2025)

1255

- 1256 • bare_enron (55 sequences)
- 1257 • bare_gsm8k (108 sequences)
- 1258 • bare_hotpot (50 sequences)
- 1259 • bare_lcb (136 sequences)
- 1260 • bare_newsgroups (60 sequences)
- 1261 • bare_pubmed (46 sequences)

1262

1263 **Draws from a binomial distribution (generated)**

1264

- 1265 • binomial (500 sequences)

1266

1267 **Draws from a shuffled deck of cards (generated)**

1268

- 1269 • cards (100 sequences)

1270

1271 **Draws from a categorical distribution (generated)**

1272

- 1273 • categorical (500 sequences)

1274 **ChangeMyView Reddit** (Kolyada et al., 2020)

1275

- 1276 • changemyview_categories (809 sequences)
- 1277 • changemyview_posts (1159 sequences)

1278

1279 **Draws from a biased coin (generated)**

1280

- 1281 • coinflip (1000 sequences)

1282

1283 **Collective Alignment Dataset** (OpenAI, 2025)

1284

- 1285 • collective_alignment_individual (993 sequences)

1286

1287 **Community Alignment Dataset** (Zhang et al., 2025b)

1288

- 1289 • community_alignment_individual_preferences (770 sequences)
- 1290 • community_alignment_individual_reply (1031 sequences)
- 1291 • community_alignment_initial_prompt (139 sequences)
- 1292 • community_alignment_response (941 sequences)

1293

1294 **DICES dataset** (Aroyo et al., 2023)

1295

- 1296 • dices (295 sequences)

1297 **Diffuse Distributions** (Zhang et al., 2024b)

1296 • diffuse_distribution (270 sequences)
 1297

1298 **Generative Social choice** (Fish et al., 2025)
 1299

1300 • generativesocialchoice_freetext (200 sequences)
 1301 • generativesocialchoice_validation (400 sequences)
 1302

1303 **Draws from a geometric distribution (generated)**
 1304 • geometric (500 sequences)
 1305

1306 **Draws from a geometric beta distribution (generated)**
 1307

1308 • geometric_beta (500 sequences)
 1309

1310 **Grade-school math problems (GSM8K)** (Cobbe et al., 2021)
 1311

1312 • gsm8k_answer_from_question (50 sequences)
 1313 • gsm8k_question (50 sequences)
 1314 • gsm8k_question_answer (50 sequences)
 1315 • gsm8k_question_from_answer (50 sequences)
 1316

1317 **Haikus** (Neiman, 2018)
 1318 • haikus (600 sequences)
 1319

1320 **Hatespeech annotations from diverse annotators** (Kumar et al., 2021)
 1321

1322 • hatespeech_individual (1000 sequences)
 1323

1324 **Helpsteer2 Synthetic Chat Preferences** (Wang et al., 2024b)
 1325

1326 • helpsteer (320 sequences)
 1327

1328 **Draws from a hypergeometric distribution, generated** (Wang et al., 2024b)
 1329

1330 • hypergeometric (500 sequences)
 1331

1332 **IssueBench (measuring political leaning of LLMs)** (Röttger et al., 2025)
 1333

1334 • issuebench (4 sequences)
 1335

1336 **Jeopardy! questions and answers** (trexmatt, 2014)
 1337

1338 • jeopardy_answer_prediction (1000 sequences)
 1339 • jeopardy_question_generation (1000 sequences)
 1340

1341 **Sarcasm detection (multiple annotators)** (Jang & Frassinelli, 2024)
 1342

1343 • lewidi_csc_sarcasm_detection_individual (872 sequences)
 1344

1345 **Irony detection (multiple annotators)** (Casola et al., 2024)
 1346

1347 • lewidi_mp_irony_detection_individual (475 sequences)
 1348

1349 **Paraphrase detection with rationales (multilpe annotators)** (Leonardelli et al., 2025)
 1350

1351 • lewidi_par_paraphrase_detection_individual (80 sequences)
 1352 • lewidi_par_paraphrase_detection_individual_categorical (80 se-
 1353 quences)

1350 **Entailment (multiple annotators)** (Weber-Genzel et al., 2024)
 1351
 1352 • lewidi_varierrnli_nli_detection_individual (52 sequences)
 1353 • lewidi_varierrnli_nli_detection_individual_categorical (52 se-
 1354 quences)
 1355

1356 **Draws from a multinomial distribution (generated)**
 1357 • multinomial (500 sequences)
 1358

1359 **Draws from a negative binomial distribution (generated)**
 1360 • negative_binomial (500 sequences)
 1361

1363 **Netflix views and rating data** (Netflix, Inc., 2009)
 1364 • netflix_individual_ratings (1000 sequences)
 1365 • netflix_individual_views (2000 sequences)
 1366

1367 **Draws from a normal distribution (generated)**
 1368 • normal (1000 sequences)
 1369

1371 **OpinionQA: Large-scale opinion survey dataset** (Santurkar et al., 2023)
 1372 • opinionqa_individual (3000 sequences)
 1374 • opinionqa_questions (15 sequences)
 1375

1376 **Draws from a poisson distribution (generated)**
 1377 • poisson (500 sequences)
 1378

1379 **Polis OpenData: Votes from a digital town hall** (The Computational Democracy Project, 2025)
 1380 • polis_comment (336 sequences)
 1382 • polis_vote (7452 sequences)
 1383

1384 **Popquorn: Annotator disagreement on 5 NLP tasks, with demographics** (Pei & Jurgens, 2023)
 1385 • popquorn_individual (400 sequences)
 1386 • popquorn_og_categorical (80 sequences)
 1387

1388 **Prism: World-wide, pluralistic chat preferences** (Kirk et al., 2024b)
 1389 • prism_individual_preferences (1333 sequences)
 1390 • prism_prompts (54 sequences)
 1392 • prism_prompts_individual (1393 sequences)
 1393

1394 **Titanic survival prediction: classic machine learning tabular dataset** (mstz, 2023)
 1395 • titanic_all_variables (14 sequences)
 1396 • titanic_survival_prediction (14 sequences)
 1398

1399 **Value Consistency: Multi-lingual value laden questions** (Moore et al., 2024)
 1400 • valueconsistency (21 sequences)
 1401

1403 **ValuePrism: datasets with moral judgments and relevant values, rights, and duties** (Sorensen
 et al., 2024a)

1404 • valueprism_misc (400 sequences)
 1405 • valueprism_situation (105 sequences)
 1406 • valueprism_vrd (500 sequences)
 1407 • valueprism_vrds_noncontextual (74 sequences)

1409 **Draws from a zipfian distribution (generated)**

1411 • zipfian (500 sequences)

1413 **C.4 TEST SPLIT**

1415 **ChatbotArena Individual Preferences** (Zheng et al., 2023)

1417 • chatbotarena_assistant (928 sequences)
 1418 • chatbotarena_individual_prefs (1183 sequences)
 1419 • chatbotarena_prompts (1000 sequences)

1421 **Tabular Chemistry Dataset** (Ramos et al., 2023)

1423 • chemistry_esol (310 sequences)
 1424 • chemistry_oxidative (102 sequences)

1426 **Synthetic Flight Preferences** (Qiu et al., 2025)

1428 • flight (200 sequences)

1429 **GlobalOQA: Country-specific Value Surevy Distributions** (Durmus et al., 2023)

1431 • globaloqa (274 sequences)

1433 **Habermas Dataset: AI Deliberation with UK residents** (Tessler et al., 2024)

1434 • habermas_individual (1996 sequences)
 1435 • habermas_individual_categorical (2000 sequences)
 1436 • habermas_opinions (199 sequences)
 1437 • habermas_question (43 sequences)

1440 **NovaCOMET: Synthetic Commonsense Dataset** (West et al., 2023)

1442 • novacomet_hypothesis (170 sequences)
 1443 • novacomet_premise (68 sequences)

1444 **NumberGame dataset: cognitive science dataset used to study human reasoning under uncertainty** (Bigelow & Piantadosi, 2016)

1447 • numbergame_individual (606 sequences)
 1448 • numbergame_perc (182 sequences)

1450 **World Values Survey, Wave 7: Global survey on human values** (EVS/WVS, 2024)

1452 • wvs_individual (2000 sequences)

1454 **C.5 CAPABILITY SPLIT**

1456 **AI2 Reasoning Challenge** (Clark et al., 2018)

1457 • arc (118 sequences)

1458 **DROP: Reading Comprehension** (Dua et al., 2019)
 1459

- 1460 • drop (943 sequences)

1461 **GPQA: Google-Proof QA Benchmark** (Rein et al., 2023)
 1462

- 1463 • gpqa (995 sequences)

1464 **Hellaswag: commonsense benchmark** (Zellers et al., 2019)
 1465

- 1466 • hellaswag (503 sequences)

1467 **IMDB sentiment classification** (Maas et al., 2011)
 1468

- 1469 • imdb (192 sequences)

1470 **MMLU: Massive Multitask Language Understanding Benchmark** (Hendrycks et al., 2021)
 1471

- 1472 • mmlu (1000 sequences)

1473 **TruthfulQA: factual questions** (Lin et al., 2022b)
 1474

- 1475 • truthful_qa (69 sequences)

1476 **Winogrande: Commonsense sentence completion** (Sakaguchi et al., 2021)
 1477

- 1478 • winogrande (127 sequences)

D EFFECT OF TEMPERATURE ON DIVERSITY VS. VALIDITY

1485 Temperature can have a major effect on the diversity vs. validity tradeoff when sampling from a
 1486 model. In §5, we observed that, when sampling across three levels of prompting information and
 1487 three model families, Spectrum tuning offered a pareto improvement on diversity vs. validity and
 1488 overall improved yield. However, the question still remains - does Spectrum tuning still offer an
 1489 improvement, even after sweeping temperature values?

1490 To answer this question, we evaluated the same models under the same setup, but sampled at various
 1491 temperatures: [10, 5, 2, 1.5, 1, 0.9, 0.7, 0.5]. In Figure A3, we plot diversity vs.
 1492 validity for all three model families, prompting methods, and model types. We find that, in eight
 1493 of nine settings, Spectrum Tuning expands the diversity / validity Pareto frontier, as compared to
 1494 using instruction-tuned or pretrained models alone. In addition, Spectrum Tuning models typically
 1495 expand the Pareto frontier in the high validity region, increasing diversity for a given validity. In
 1496 line with the temperature=1 results, Spectrum Tuning’s gains offer the largest improvement in the
 1497 lowest information setting, when only a description of the task is provided.

1498 In Figure A4, we also plot the yield for each setting against the temperature. We find that in eight
 1499 of nine cases, Spectrum Tuning offers the highest possible yield across all models and temperatures
 1500 - implying that, even if when selecting the optimal temperature for each generation task, we would
 1501 expect the highest number of distinct valid generations from the Spectrum-Tuned models.

1502 Taken together, we find that the gains from Spectrum Tuning hold even when leaving temperature
 1503 as a free variable.

E GENERAL CAPABILITY PERFORMANCE

1505 We test whether SPECTRUM TUNING affects general model capabilities. While we do not necessarily
 1506 expect our method to improve upon standard evaluations where there is a single correct answer,
 1507 we want to understand if it degrades performance compared to pretrained models. We evaluate
 1508 general knowledge capabilities with Big-Bench Hard (BBH, 3-shot, Suzgun et al. 2023), GPQA
 1509 (5-shot with chain of thought, Rein et al. 2024), MMLU-Pro (5-shot with chain of thought, Wang
 1510

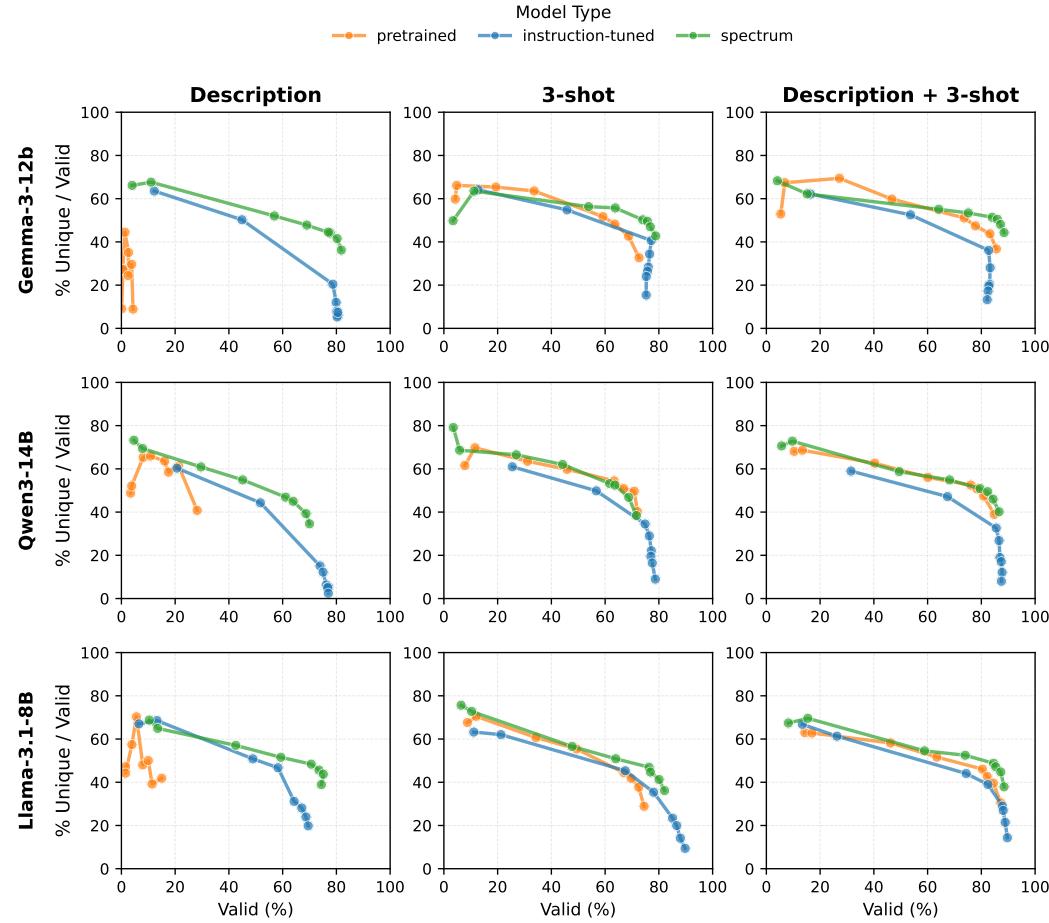


Figure A3: Effect of temperature on diversity and validity. Tested temperatures: $[10, 5, 2, 1.5, 1, 0.9, 0.7, 0.5]$. Lines are connected for temperature in ascending order, with the right-most endpoint being lowest temperature and the left-most endpoint being highest temperature. Spectrum Tuning generally offers a Pareto improvement, especially in the high validity region.

et al. 2024a), and TruthfulQA (6-shot, Lin et al. 2022a); instruction following with IFEval (Zeng et al., 2024); and chat ability with AlpacaEval v2 (Dubois et al., 2024). We use the default Olmes hyperparameters for evaluating pretrained models, and Tulu-v3 hyperparameters and task descriptions for evaluating instruction-tuned models (Gu et al., 2025; Lambert et al., 2025). In general, we find that models trained with SPECTRUM TUNING perform similarly to the pretrained models, and in some cases exceed them; however, as expected, instruction-tuned models perform much better, particularly on instruction following and chat tasks.

F TRAINING DETAILS

We lightly tuned hyperparameters by training the gemma-3-12b model on a subset of tasks from SPECTRUM SUITE-Train and tracking performance on held-out train tasks. We used the same hyperparameters for Llama and Qwen, performing no additional hyperparameter tuning. Training for all models was done on four 80GB A100 GPUs using DeepSpeed Zero3 (Rajbhandari et al., 2021) and Hugging Face Transformers (Wolf et al., 2020). Training took about 16 hours for the Llama models, 26 hours for the Gemma models, and 30 hours for the Qwen models.

Hyperparameters used:

- max_length: 1024

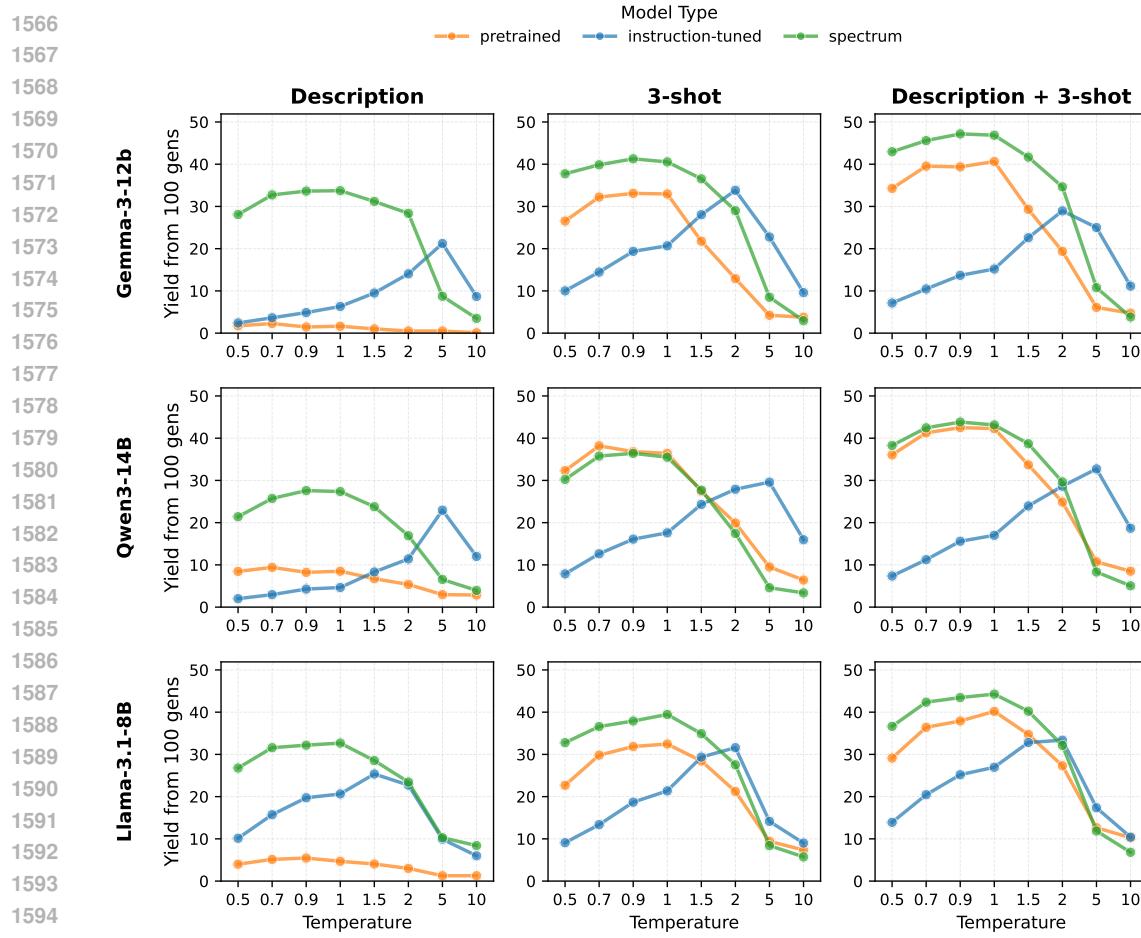


Figure A4: Effect of temperature on yield across each setting. When selecting the optimal temperature for each model, Spectrum Tuning offers the highest overall yield in 8/9 cases (all but Qwen3-14B / 3-shot). Spectrum Tuning also offers the highest yield in most temperature settings $T \leq 2$.

Dataset	gemma-3-12b			Qwen3-14B			Llama-3.1-8B		
	ST (ours)	PT	IT	ST (ours)	PT	IT	ST (ours)	PT	IT
AlpacaEval 2	<u>5.935</u>	6.897	53.846	<u>30.421</u>	33.541	63.123	3.642	<u>3.579</u>	24.641
BBH	0.738	<u>0.727</u>	0.821	0.786	<u>0.789</u>	<u>0.770</u>	0.641	<u>0.631</u>	0.722
GPQA	0.257	<u>0.250</u>	0.377	<u>0.339</u>	0.386	0.411	0.246	<u>0.208</u>	0.315
IFEval	<u>0.407</u>	0.436	0.806	<u>0.712</u>	0.726	0.871	0.377	<u>0.296</u>	0.793
MMLU-Pro	0.458	<u>0.448</u>	0.592	0.584	<u>0.555</u>	0.684	0.358	0.360	0.481
TruthfulQA	0.516	0.483	0.610	<u>0.498</u>	0.529	0.553	0.435	0.446	0.551

Table A1: General Capability Results. *Worst* performance is underlined. SPECTRUM TUNING and pretrained models perform similarly.

- `per_device_train_batch_size`: 1
- `gradient_accumulation_steps`: 512
- `learning_rate`: 3e-6
- `learning_rate_scheduler`: `linear_decay`

1620 G RESULTS WITH UPDATED HYPERPARAMETERS

1621
 1622 After running the main suite of experiments for the paper and experimenting with the models, we
 1623 had reason to believe that our Spectrum-Tuned models, especially the Qwen and Llama models,
 1624 were underfit. Note that, for the main set of experiments, we only lightly fit hyperparameters only
 1625 on the Gemma models using a held-out subset of the train tasks as a validation set, and used the
 1626 same hyperparameters for Qwen / Llama.

1627 To further explore the effect of updating hyperparameters, we experimented with reducing the batch
 1628 size in order to take more gradient updates. In the original hyperparameter mix, we use an effective
 1629 batch size of 2048 (512 gradient steps \times 1 train sequence per device \times 4 GPUs). We halve the batch
 1630 size three times, and report aggregate results in Table A2.

1631
 1632
 1633 Table A2: Hyperparameter ablations, averaged across models and tasks. Shaded are default SPEC-
 1634 TRUM TUNING models. Best result bolded, second best underlined.

1635
 1636
 1637
 1638 We find that 1) decreasing the batch size results a substantial jump in zero-shot yield, and slight
 1639 improvements in few-shot yield and distributional alignment. Additionally, decreasing the batch size
 1640 increases multiple choice accuracy, but at the cost of higher loss on multiple choice answers. All in
 1641 all, we think that this illustrates that there are likely to be additional gains from further optimization,
 1642 and that our initial hyperparameters were likely underfit.

1643
 1644 We think that the models trained with effective batch size 512 offer a good tradeoff between ICL
 1645 steerability, distributional alignment, and valid output coverage, and report their full results in Tables
 1646 A3-A5 and Figure A5.

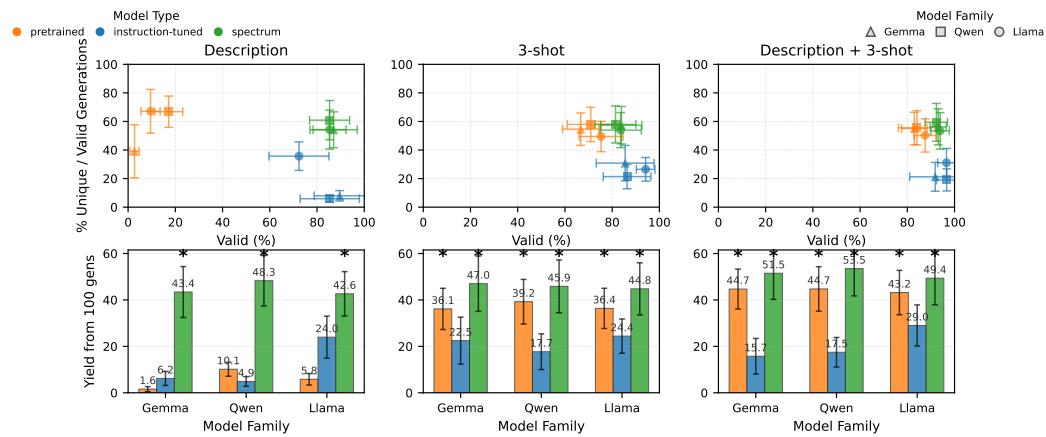
1647 1648 1649 1650 1651 1652 1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 Dataset	1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 Metric	1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 gemma-3-12b			1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 Qwen3-14B			1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 Llama-3.1-8B		
		1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 ours	1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 pt	1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 it	1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 ours	1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 pt	1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 			
Multiple-Choice Datasets										
gemma-3-12b										
<code>habermas.individual.categorical</code> (max.k=2, N=1000)	Loss	3.53	2.50	10.5	2.01	2.62	9.10	2.58	2.58	2.74
	Acc	24.0	24.4	22.4	24.9	20.3	22.0	23.2	20.2	19.0
<code>wvs.individual</code> (max.k=21, N=1000)	Loss	1.36	1.50	4.10	1.38	1.74	4.35	1.42	1.57	1.76
	Acc	44.7	42.1	40.4	45.2	41.1	40.6	44.5	41.6	39.4
<code>numbergame.individual</code> (max.k=25, N=592)	Loss	.665	.705	1.80	.617	.697	1.28	.611	.864	.770
	Acc	70.2	64.3	65.6	71.2	69.8	71.0	69.2	62.5	67.5
<code>chatbotarena.individual_prefs</code> (max.k=3, N=725)	Loss	1.52	1.62	4.94	1.35	1.47	4.39	1.43	1.76	1.77
	Acc	48.9	38.0	44.6	51.7	52.0	46.3	39.5	36.0	39.5
<code>flight</code> (max.k=9, N=200)	Loss	1.11	1.32	4.06	1.09	1.29	2.92	1.09	1.45	1.41
	Acc	41.0	41.2	40.6	43.1	43.7	40.8	40.9	42.0	40.2
Free-text Datasets										
gemma-3-12b										
<code>novacomet.hypothesis</code> (max.k=11, N=155)	Loss	105	104	135	107	106	129	110	106	112
	Acc	27.7	28.0	35.5	27.7	27.5	38.0	27.9	27.7	28.6
<code>habermas.question</code> (max.k=29, N=30)	Loss	23.9	23.1	41.4	23.8	24.0	31.8	23.8	23.8	24.8
	Acc	92.7	92.8	1070	947	949	1070	944	944	991
<code>habermas.opinions</code> (max.k=2, N=186)	Loss	164	164	203	167	168	210	166	167	176
	Acc	4.26	4.22	6.68	4.13	4.24	5.61	4.31	4.43	4.41
<code>globaloqa</code> (max.k=8, N=231)	Loss	14.2	14.4	21.5	14.0	14.4	20.9	14.5	14.7	15.6
	Acc	69.8	69.4	117	67.9	68.2	97.8	72.0	72.0	77.6
<code>chatbotarena.prompt</code> (max.k=3, N=988)	Loss	127	125	259	124	124	169	136	133	149
	Acc	8.45	8.37	12.9	8.45	8.47	11.8	8.30	8.51	8.55
<code>chemistry.esol</code> (max.k=8, N=59)	Loss	7.57	7.58	11.6	7.57	7.84	10.2	7.68	7.72	7.84
	Acc	10.6	10.5	11.6	10.5	10.4	11.6	10.3	10.2	10.2
<code>chemistry.oxidative</code> (max.k=9, N=101)	Loss	7.57	7.58	11.6	7.57	7.84	10.2	7.68	7.72	7.84

1673
 1674 Table A3: In-context steerability results on models trained with an effective batch size of 512.

1674
1675
1676
1677
1678
1679
1680
1681
1682
1683

Dataset	gemma-3-12b			Qwen3-14B			Llama-3.1-8B		
	ours	pt	it	ours	pt	it	ours	pt	it
habermas_individual_categorical	0.13	0.069	0.239	0.049	0.05	0.198	0.108	0.084	0.055
wvs_individual	0.007	0.015	0.223	0.007	0.02	0.191	0.005	0.012	0.024
numbergame_individual	0.019	0.029	0.163	0.037	0.026	0.108	0.027	0.024	0.017
chatbotarena_individual_prefs	0.02	0.041	0.194	0.056	0.046	0.189	0.062	0.075	0.049
flight	0.019	0.04	0.271	0.055	0.035	0.228	0.03	0.07	0.038

1684 Table A4: Calibration for models trained with an effective batch size of 512.
1685
1686
1687
1688
1689
1690
1691



1707 Figure A5: Diversity vs. validity on verifiable tasks for models trained with an effective batch size
1708 of 512.
1709
1710
1711
1712
1713
1714
1715

Dataset	Metric	gemma-3-12b			Qwen3-14B			Llama-3.1-8B		
		ours	pt	it	ours	pt	it	ours	pt	it
mpi	JS-Div	.101	.126	.347	.107	.0928	.405	.0489	.0874	.131
rotten_tomatoes	JS-Div	.0227	.0323	.134	.0341	.0283	.122	.0245	.0354	.0859
nytimes	JS-Div	.0547	.0628	.328	.0453	.0876	.344	.0655	.0613	.247
global_loqa	JS-Div	.0678	.0936	.270	.0749	.0878	.274	.0828	.108	.163
urn	JS-Div	.0136	.0713	.185	.0186	.0592	.198	.0186	.124	.0865
habermas	JS-Div	.142	.147	.436	.125	.127	.434	.129	.155	.242
numbergame	JS-Div	.0663	.0488	.138	.0440	.0428	.131	.0423	.0600	.0943

1724 Table A5: Distributional alignment for model strained with an effective batch size of 512.
1725
1726
1727

1728 H HUMAN EVALUATION

1730 We conducted a large-scale human annotation study to evaluate the validity and quality of outputs
 1731 from different model configurations. The study used a pairwise comparison design where annotators
 1732 evaluated outputs from two models simultaneously for the same prompts. We recruited 245 U.S.-
 1733 based English speaking annotators who had submitted at least 1000 prior tasks with an approval
 1734 rating of at least 95% through Prolific and collected a total of 2,400 annotations. Our task took
 1735 about 30 minutes and we paid at least 7.5 USD for an average of at least 15 USD an hour.

1736 Specifically, we sampled 100 prompts from two evaluation datasets, a curated prompt set and
 1737 infinite-chats-eval, and collected human judgments for each. Our experimental design compared
 1738 three model configurations (baseline instruction-tuned, our approach, and pretrained) in both zero-
 1739 shot and few-shot settings. Each unique combination of (prompt, model pair) was evaluated by two
 1740 independent annotators, resulting in 200 annotation instances per model pair per dataset.

1741
 1742 **Annotation Interface and Procedure** Participants accessed the annotation task through a web-
 1743 based interface. First, participants were asked to thoroughly read through the comprehensive anno-
 1744 tation guidelines with examples of valid and invalid responses (See Figure A6 and Figure A7). For
 1745 each annotation instance, annotators were presented with a prompt and four generations from each
 1746 of two models (labeled Model A and Model B). The model identities and presentation order were
 1747 randomized to prevent systematic bias. The interface displayed the outputs side-by-side to facilitate
 1748 direct comparison (See Figure A8 for the user interface and questions).

1749 For each task, annotators made three types of judgments:

- 1750 • **Validity Assessment:** Annotators independently marked each of the eight generations (4
 1751 per model) as either valid or invalid. We provided detailed guidelines defining validity as
 1752 responses that directly address the prompt, follow all specified requirements, stay on-topic
 1753 throughout, and contain factually reasonable content. Invalid responses included those that
 1754 refuse to answer, violate format requirements, trail off into unrelated content, or contain
 1755 significant errors.
- 1756 • **Diversity Comparison:** Annotators assessed which model’s set of four outputs exhibited
 1757 greater diversity, with options for Model A, Model B, or “about the same.”
- 1758 • **Overall Quality Judgment:** Independent of diversity, annotators selected which model’s
 1759 outputs were better overall, again with options for either model or “about the same.”

1760 To ensure annotation quality, we implemented several measures: (1) Comprehensive annotation
 1761 guidelines with examples of valid and invalid responses, (2) Tracking of time spent per annotation,
 1762 and (3) Post-annotation feedback collection to identify any systematic issues.

1763
 1764 **Inter-Annotator Agreement** Inter-annotator agreement for validity judgments showed 76.5%
 1765 pairwise percentage agreement, with Cohen’s $\kappa = 0.441$, indicating moderate agreement. For the
 1766 subjective diversity and quality assessments, agreement rates were lower (diversity: 38.8%, quality:
 1767 41.7%), as expected given the more nuanced nature of these judgments.

1770 I LLM USAGE DESCRIPTION

1771 In preparation of this research and manuscript, LLMs were used for:

- 1772 • Implementing code for experiments and analysis based on detailed author descriptions. All
 1773 LLM code was inspected by the authors for correctness.
- 1774 • Formatting for tables, latex, and bibtex citation for non-traditional sources (e.g., urls).
- 1775 • Draft critique by pointing out typos and potentially confusing wording in the draft.

1776 However, all research ideation and writing was performed solely by the authors, and we claim full
 1777 responsibility for the contents of this paper.

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

Annotation Guidelines

>Your Task

You will evaluate model outputs for prompts that can have multiple valid answers. For each prompt, you'll see outputs from two different models (A and B).

What You'll Annotate

- **Validity:** For each individual output, mark whether it's a valid/correct response to the prompt
- **Diversity:** Compare the variety of responses between the two models
- **Overall Quality:** Judge which model's outputs are better overall

More Guidelines on Validity (Important!)

A response is **VALID** if it:

- Directly addresses what the prompt asks for
- Follows ALL specific requirements in the prompt (length, format, style, etc.)
- Stays on topic throughout the entire response
- Is factually reasonable (doesn't contain obvious falsehoods)

A response is **INVALID** if it:

- Ignores or refuses to answer the prompt
- Violates specific requirements (e.g., prompt asks for "a few sentences" but gives only one word)
- Starts correctly but then trails off into unrelated content
- Contains significant errors or nonsensical content

⚠ Special Note on Cut-off Responses: Some responses may be cut off mid-sentence due to length limits (512 tokens). If a response is clearly incomplete but what's there is valid and on-topic, consider it **VALID**. Only mark as invalid if the existing content itself is problematic.

Figure A6: Guidelines for human annotation

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

Examples to Guide Your Judgment

Example 1: Not Following Requirements

Prompt: In a few sentences explain what threats do scams pose to individuals?

Response: "Scams can cause a financial loss to the victim, potentially leading to debt, bankruptcy, and even mental and emotional harm."

X INVALID

Reason: The prompt explicitly asks for "a few sentences" but the response is only one sentence. This violates the format requirement.

Example 2: Trailing Off Topic

Prompt: What are the benefits of regular exercise?

Response: "Regular exercise improves cardiovascular health, strengthens muscles, and boosts mental well-being. It can help with weight management and reduce the risk of chronic diseases. By the way, did you know that dolphins are actually mammals, not fish? They breathe air and nurse their young..."

X INVALID

Reason: The response starts correctly but then trails off into completely unrelated content about dolphins.

Example 3: Cut-off but Valid

Prompt: Describe the water cycle.

Response: "The water cycle is a continuous process where water moves through Earth's atmosphere, land, and oceans. It begins with evaporation, where water from oceans, lakes, and rivers turns into water vapor due to heat from the sun. This vapor rises into the atmosphere where it cools and condenses to form clouds. When the water droplets in clouds become heavy enough, they fall as precipitation in the form of rain, snow, or hail. The precipitated water then flows into rivers and streams, eventually making its way back to the ocean, where the cycle begins..."

✓ VALID

Reason: Although the response is cut off mid-word ("begins..."), the existing content is accurate, on-topic, and addresses the prompt well.

Example 4: Refusal to Answer

Prompt: Write a haiku about the ocean.

Response: "I cannot write creative content like poetry. Would you like me to provide information about oceans instead?"

X INVALID

Reason: The model refuses to do what the prompt asks, even though writing a haiku is a reasonable request.

Example 5: Good Valid Response

Prompt: Name three primary colors.

Response: "The three primary colors are red, blue, and yellow."

✓ VALID

Reason: Directly and correctly answers the prompt.

Other Important Notes

- You need to complete at least **10 annotations**
- After 10, you can choose to continue or end the session
- The model order (A/B) is randomized - focus on the outputs, not which side they're on
- Consider each output independently when judging validity
- For diversity and quality questions, consider all outputs from each model together
- Take your time - there's no rush. Quality matters more than speed

Figure A7: Guideline examples for human annotation

1890
 1891
 1892
 1893
 1894
 1895
 1896
 1897
 1898
 1899
 1900
 1901
1902 Prompt: One thing you might find in a magical forest.
 1903
 1904
 1905
 1906
 1907
 1908
 1909
 1910
 1911
 1912
 1913
 1914
 1915
 1916
 1917
 1918
 1919
 1920
 1921
 1922
 1923
 1924
 1925
 1926
 1927
 1928
 1929
 1930
 1931
 1932
 1933
 1934
 1935
 1936
 1937
 1938
 1939
 1940
 1941
 1942
 1943

Completed: 0 annotations (Minimum 10 more required) Time: 0:16

Prompt: One thing you might find in a magical forest.

Model A

A talking mushroom ring. #1

They might offer cryptic advice, tell silly jokes, or even lead you on a quest!

Valid Invalid

A shimmering, moss-covered stone circle, humming with a faint, otherworldly energy. #2

It could be a portal, a place of ancient power, or simply a beautiful, mysterious landmark - but it definitely feels magical!

Valid Invalid

A shimmering, moss-covered stone circle humming with ancient energy. #3

It could be a portal, a place of power, or just a really cool, magical picnic spot.

Valid Invalid

A shimmering, moss-covered stone that whispers forgotten prophecies when touched. #4

Hope that sparks your imagination!

Valid Invalid

Model B

A variety of magical flora such as talking trees, glowing mushrooms, or even a magical unicorn. #1

Valid Invalid

A magic mirror. You can see everything that's going on in the world. #2

Valid Invalid

In a magical forest, you might find mystical creatures like unicorns, fairies, or mythical beasts like griffins. These creatures are often seen in folklore and fantasy stories as inhabitants of magical worlds. Additionally, the forest might be adorned with fantastical plants with peculiar properties or emit glowing lights, creating an enchanting and otherworldly atmosphere. The magical forest might also have enchanted trees, each whispering secrets or granting wishes.

Valid Invalid

In a magical forest, you might find anything from enchanted creatures to hidden treasures or mystical creatures. #4

Valid Invalid

Which set of outputs would you say is more diverse (i.e., the most different from each other)?

A outputs are more diverse
 They are about the same
 B outputs are more diverse

If you had to pick, which set of outputs do you think is better overall, regardless of diversity?

A outputs are overall better
 They are about the same
 B outputs are overall better

Submit & Next

Figure A8: User Interface for Annotation

1944
1945

J SUPPLEMENTARY FIGURES

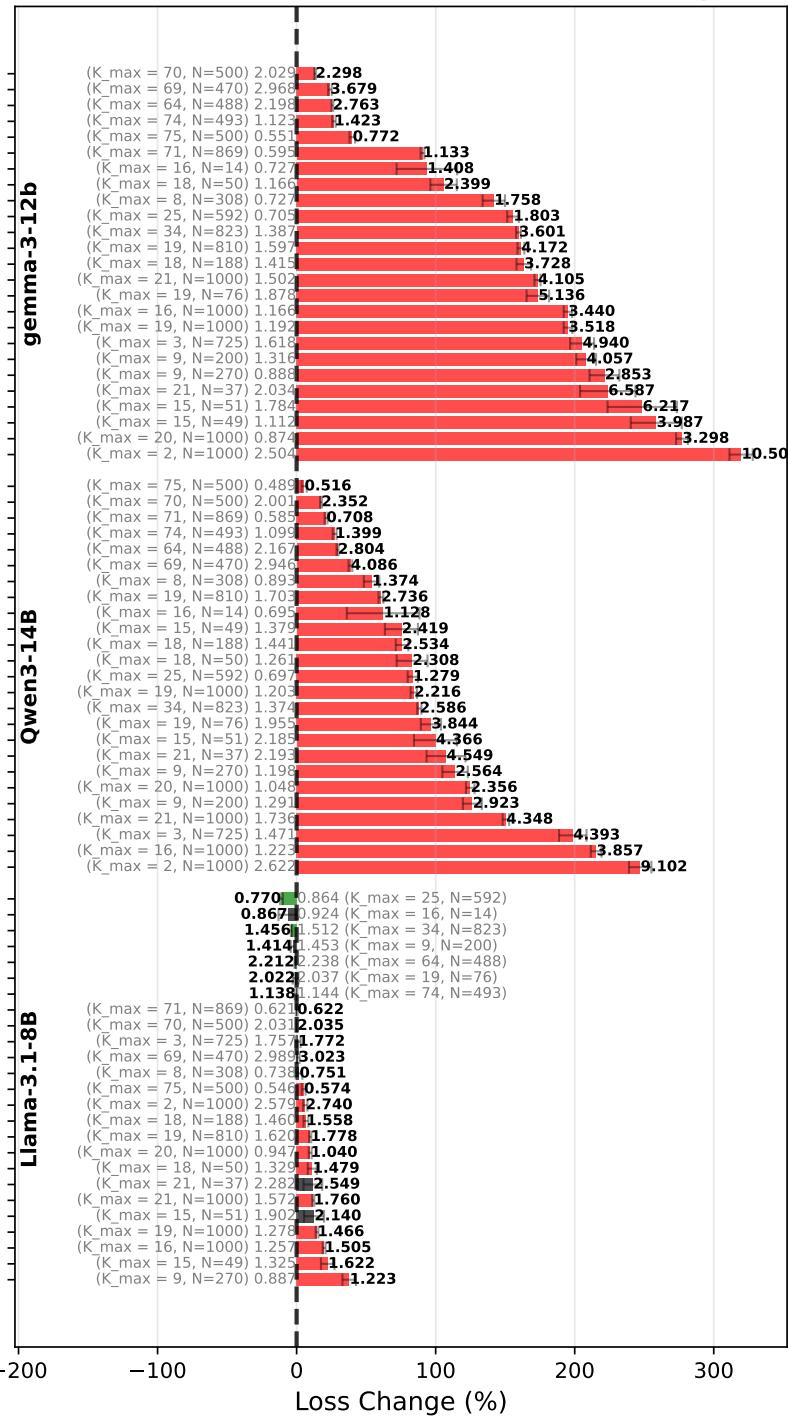
1946
1947
1948
1949Spectrum Suite (Categorical)
Relative Loss After Instruction-Tuning1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997

Figure A9: SPECTRUM SUITE categorical loss after instruction-tuning

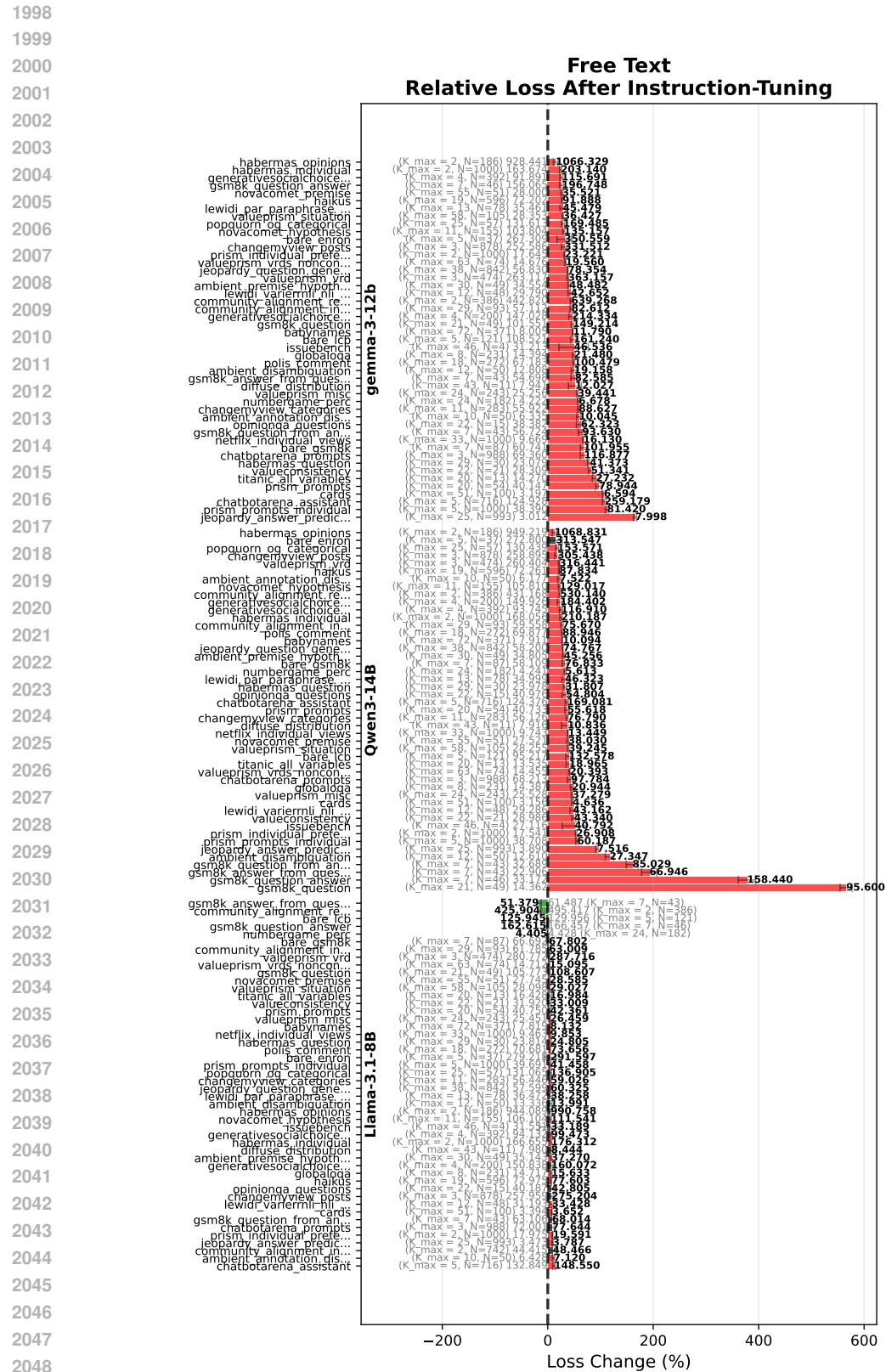


Figure A10: SPECTRUM SUITE free-text loss after instruction-tuning

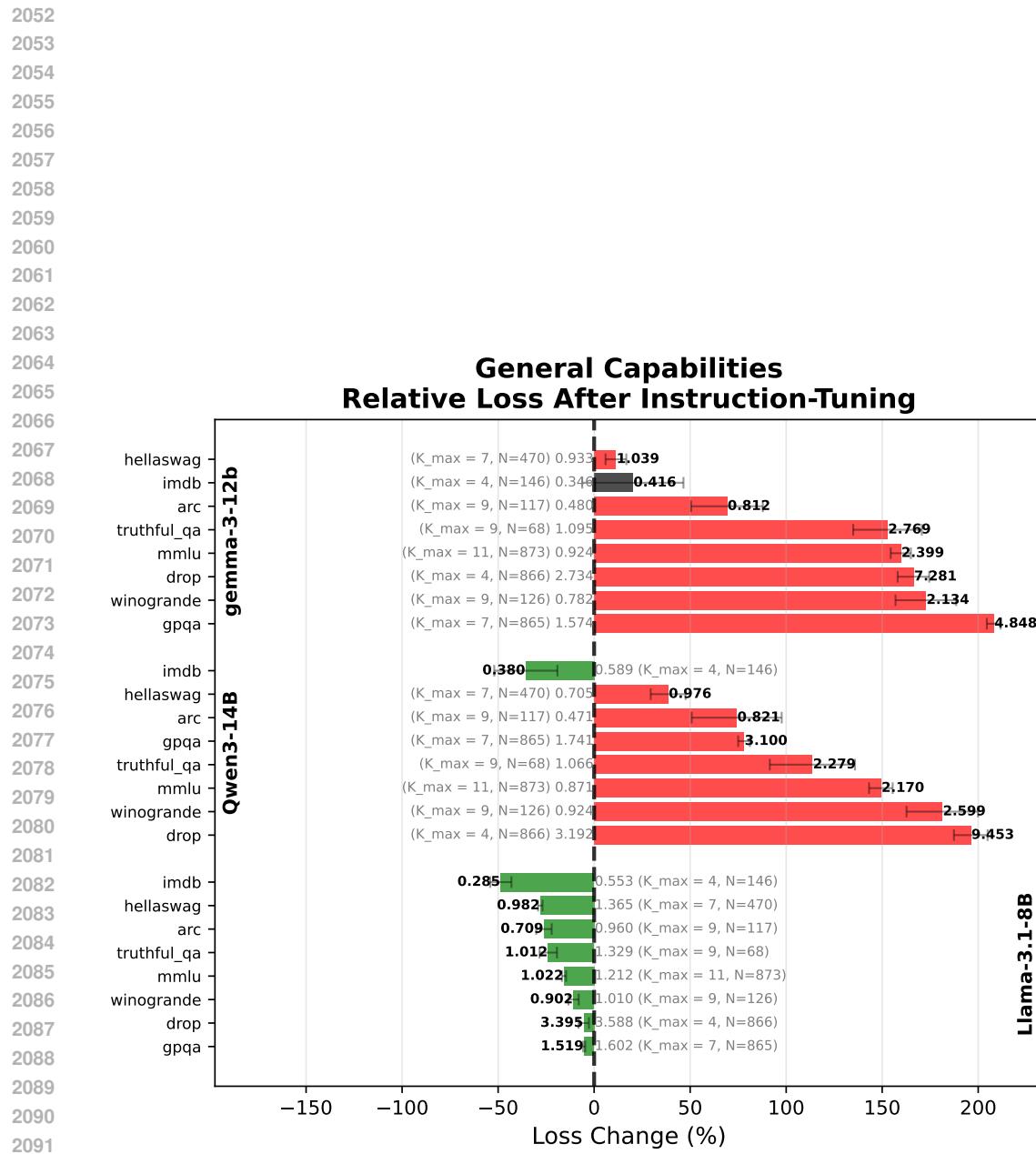


Figure A11: SPECTRUM SUITE general capability loss after instruction-tuning

2106 **K SPECTRUM TUNING TEMPLATES**
21072108 For all templates, loss is calculated on the highlighted output tokens.
21092110 **gemma-3 (w/ inputs)**

```

2111 <start_of_turn>description
2112 DESCRIPTION TEXT<end_of_turn>
2113 <start_of_turn>input
2114 INPUT 1 TEXT<end_of_turn>
2115 <start_of_turn>output
2116 OUTPUT 1 TEXT<end_of_turn>
2117 <start_of_turn>input
2118 INPUT 2 TEXT<end_of_turn>
2119 <start_of_turn>output
2120 OUTPUT 2 TEXT<end_of_turn>
2121 <start_of_turn>input
2122 INPUT 3 TEXT<end_of_turn>
2123 <start_of_turn>output
2124 OUTPUT 3 TEXT<end_of_turn>
2125 ...
2126

```

2125 **gemma-3 (w/out inputs)**

```

2127 <start_of_turn>description
2128 DESCRIPTION TEXT<end_of_turn>
2129 <start_of_turn>output
2130 OUTPUT 1 TEXT<end_of_turn>
2131 <start_of_turn>input
2132 OUTPUT 2 TEXT<end_of_turn>
2133 <start_of_turn>input
2134 OUTPUT 3 TEXT<end_of_turn>
2135 ...
2136

```

2135 **Qwen3 (w/ inputs)**

```

2137 <|im_start|>description
2138 DESCRIPTION TEXT<|im_end|>
2139 <|im_start|>input
2140 INPUT 1 TEXT<|im_end|>
2141 <|im_start|>output
2142 OUTPUT 1 TEXT<|im_end|>
2143 <|im_start|>input
2144 INPUT 2 TEXT<|im_end|>
2145 <|im_start|>output
2146 OUTPUT 2 TEXT<|im_end|>
2147 <|im_start|>input
2148 INPUT 3 TEXT<|im_end|>
2149 <|im_start|>output
2150 OUTPUT 3 TEXT<|im_end|>
2151 ...
2152

```

2151 **Qwen3 (w/out inputs)**

```

2153 <|im_start|>description
2154 DESCRIPTION TEXT<|im_end|>
2155 <|im_start|>output
2156 OUTPUT 1 TEXT<|im_end|>
2157 <|im_start|>output
2158 OUTPUT 2 TEXT<|im_end|>
2159 <|im_start|>output
2160 OUTPUT 3 TEXT<|im_end|>
2161 ...
2162

```

2160 **Llama-3.1** (w/ inputs)
 2161
 2162 <|start_header_id|>description<|end_header_id|>
 2163 DESCRIPTION TEXT<|eot_id|><|start_header_id|>input<|end_header_id|>
 2164
 2165 INPUT 1 TEXT<|eot_id|><|start_header_id|>output<|end_header_id|>
 2166
 2167 **OUTPUT 1 TEXT<|eot_id|>**<|start_header_id|>input<|end_header_id|>
 2168
 2169 INPUT 2 TEXT<|eot_id|><|start_header_id|>output<|end_header_id|>
 2170
 2171 **OUTPUT 2 TEXT<|eot_id|>**<|start_header_id|>input<|end_header_id|>
 2172
 2173 INPUT 3 TEXT<|eot_id|><|start_header_id|>output<|end_header_id|>
 2174
 2175 **OUTPUT 3 TEXT<|eot_id|>...**

2175 **Llama-3.1** (w/out inputs)
 2176
 2177 <|start_header_id|>description<|end_header_id|>
 2178 DESCRIPTION TEXT<|eot_id|><|start_header_id|>output<|end_header_id|>
 2179
 2180 **OUTPUT 1 TEXT<|eot_id|>**<|start_header_id|>output<|end_header_id|>
 2181
 2182 **OUTPUT 2 TEXT<|eot_id|>**<|start_header_id|>output<|end_header_id|>
 2183
 2184 **OUTPUT 3 TEXT<|eot_id|>...**

L PRETRAINED / INSTRUCTION-TUNED ICL TEMPLATES

2188 **Pretrained Template** (w/ inputs)

2189 Note that each output ends with two newlines to ensure a terminal token (coloring not visible).

2191 Description: DESCRIPTION TEXT
 2192
 2193 Input: INPUT 1 TEXT
 2194
 2195 Output: **OUTPUT 1 TEXT**
 2196
 2197 Input: INPUT 2 TEXT
 2198
 2199 Output: **OUTPUT 2 TEXT**
 2200
 2201 Input: INPUT 3 TEXT
 2202
 2203 Output: **OUTPUT 3 TEXT**
 2204
 2205 ...

2204 **Pretrained Template** (w/out inputs)

2205 Note that each output ends with two newlines to ensure a terminal token (coloring not visible).

2207 Description: DESCRIPTION TEXT
 2208
 2209 Output: **OUTPUT 1 TEXT**
 2210
 2211 Output: **OUTPUT 2 TEXT**
 2212
 2213 Output: **OUTPUT 3 TEXT**
 2214
 2215 ...

2214 **Simple Instruct Template**

2215 Qwen3 (task w/inputs)

```

2216
2217 <|im_start|>system
2218 DESCRIPTION TEXT<|im_end|>
2219 <|im_start|>user
2220 INPUT 1 TEXT<|im_end|>
2221 <|im_start|>assistant
2222 <think>
2223 </think>
2224 OUTPUT 1 TEXT<|im_end|>
2225 <|im_start|>user
2226 INPUT 2 TEXT<|im_end|>
2227 <|im_start|>assistant
2228 <think>
2229 </think>
2230
2231 OUTPUT 2 TEXT<|im_end|>
2232 <|im_start|>user
2233 INPUT 3 TEXT<|im_end|>
2234 <|im_start|>assistant
2235 <think>
2236 </think>
2237
2238 OUTPUT 3 TEXT<|im_end|>

```

2239 Qwen3 (task w/out inputs)

```

2240
2241 <|im_start|>system
2242 DESCRIPTION TEXT<|im_end|>
2243 <|im_start|>user
2244 Generate<|im_end|>
2245 <|im_start|>assistant
2246 <think>
2247 </think>
2248
2249 OUTPUT 1 TEXT<|im_end|>
2250 <|im_start|>user
2251 Generate<|im_end|>
2252 <|im_start|>assistant
2253 <think>
2254 </think>
2255
2256 OUTPUT 2 TEXT<|im_end|>
2257 <|im_start|>user
2258 Generate<|im_end|>
2259 <|im_start|>assistant
2260 <think>
2261 </think>
2262
2263 OUTPUT 3 TEXT<|im_end|>

```

2264 gemma-3 (task w/inputs)

```

2265 <start_of_turn>user
2266 DESCRIPTION TEXT
2267
2268 INPUT 1 TEXT<end_of_turn>

```

```

2268 <start_of_turn>model
2269 OUTPUT 1 TEXT<end_of_turn>
2270 <start_of_turn>user
2271 INPUT 2 TEXT<end_of_turn>
2272 <start_of_turn>model
2273 OUTPUT 2 TEXT<end_of_turn>
2274 <start_of_turn>user
2275 INPUT 3 TEXT<end_of_turn>
2276 <start_of_turn>model
2277 OUTPUT 3 TEXT<end_of_turn>
2278
2279 gemma-3 (task w/out inputs)
2280
2281 <start_of_turn>user
2282 DESCRIPTION TEXT
2283
2284 Generate<end_of_turn>
2285 <start_of_turn>model
2286 OUTPUT 1 TEXT<end_of_turn>
2287 <start_of_turn>user
2288 Generate<end_of_turn>
2289 <start_of_turn>model
2290 OUTPUT 2 TEXT<end_of_turn>
2291 <start_of_turn>user
2292 Generate<end_of_turn>
2293 <start_of_turn>model
2294 OUTPUT 3 TEXT<end_of_turn>
2295
2296 Llama-3.1 (task w/inputs)
2297
2298 <|begin_of_text|><|start_header_id|>system<|end_header_id|>
2299
2300 Cutting Knowledge Date: December 2023
2301 Today Date: DD MM YYYY
2302
2303 DESCRIPTION TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
2304
2305 INPUT 1 TEXT<|eot_id|><|start_header_id|>assistant<|end_header_id|>
2306
2307 OUTPUT 1 TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
2308
2309 INPUT 2 TEXT<|eot_id|><|start_header_id|>assistant<|end_header_id|>
2310
2311 OUTPUT 2 TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
2312
2313 INPUT 3 TEXT<|eot_id|><|start_header_id|>assistant<|end_header_id|>
2314
2315 OUTPUT 3 TEXT<|eot_id|>
2316
2317 Llama-3.1 (task w/out inputs)
2318
2319 <|begin_of_text|><|start_header_id|>system<|end_header_id|>
2320
2321 Cutting Knowledge Date: December 2023
2322 Today Date: 26 Jul 2024
2323
2324 DESCRIPTION TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
2325
2326 Generate<|eot_id|><|start_header_id|>assistant<|end_header_id|>
2327
2328 OUTPUT 1 TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
2329
2330 Generate<|eot_id|><|start_header_id|>assistant<|end_header_id|>
2331
2332

```

```

2322 OUTPUT 2 TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
2323
2324 Generate<|eot_id|><|start_header_id|>assistant<|end_header_id|>
2325
2326 OUTPUT 3 TEXT<|eot_id|>
2327

```

2328 Detailed Instruct Template

2329 Qwen (task w/ inputs)

```

2330 <|im_start|>system
2331 You are tasked with generating outputs from a particular, potentially
2332   ↪ stochastic, generative process. You will be given some combination of
2333   ↪ :
2334 - Description: A natural description of the generative process / data
2335   ↪ distribution
2336 - Input: An input on which to condition the generative process.
2337 - Example outputs: Example outputs from the process, either in a user
2338   ↪ message or as prior generations from a chat message. You may assume
2339   ↪ that any given outputs are exchangeable with one another (order-
2340   ↪ invariant) and generated from the same process (roughly i.i.d.). If
2341   ↪ the output data pertains to a single object, it just contains the
2342   ↪ output. If it contains multiple objects, use json formatting with
2343   ↪ keys for the name of the output variable.
2344 You will be provided at least either a description or an example output.
2345
2346 Given these components, your job is to generate JUST the output in your
2347   ↪ response, roughly approximating the underlying generative process,
2348   ↪ maintaining any underlying stochasticity (if any is present). If you
2349   ↪ are asked to generate again, you will either be given an additional
2350   ↪ input to condition on, or will just be told to "Generate".
2351
2352 Description: DESCRIPTION TEXT<|im_end|>
2353 <|im_start|>user
2354 INPUT 1 TEXT<|im_end|>
2355 <|im_start|>assistant
2356 <think>
2357 </think>
2358 OUTPUT 1 TEXT<|im_end|>
2359 <|im_start|>user
2360 INPUT 2 TEXT<|im_end|>
2361 <|im_start|>assistant
2362 <think>
2363 </think>
2364 OUTPUT 2 TEXT<|im_end|>
2365 <|im_start|>user
2366 INPUT 3 TEXT<|im_end|>
2367 <|im_start|>assistant
2368 <think>
2369 </think>
2370 OUTPUT 3 TEXT<|im_end|>

```

2371 Qwen (task w/out inputs)

```

2372 <|im_start|>system
2373 You are tasked with generating outputs from a particular, potentially
2374   ↪ stochastic, generative process. You will be given some combination of
2375   ↪ :

```

2376 - Description: A natural description of the generative process / data
 2377 → distribution
 2378 - Input: An input on which to condition the generative process.
 2379 - Example outputs: Example outputs from the process, either in a user
 2380 → message or as prior generations from a chat message. You may assume
 2381 → that any given outputs are exchangeable with one another (order-
 2382 → invariant) and generated from the same process (roughly i.i.d.). If
 2383 → the output data pertains to a single object, it just contains the
 2384 → output. If it contains multiple objects, use json formatting with
 2385 → keys for the name of the output variable.
 2386 You will be provided at least either a description or an example output.

2386 Given these components, your job is to generate JUST the output in your
 2387 → response, roughly approximating the underlying generative process,
 2388 → maintaining any underlying stochasticity (if any is present). If you
 2389 → are asked to generate again, you will either be given an additional
 2390 → input to condition on, or will just be told to "Generate".

2391

2392 Description: DESCRIPTION TEXT<|im_end|>
 2393 <|im_start|>user
 2394 Generate<|im_end|>
 2395 <|im_start|>assistant
 2396 <think>
 2397 </think>
 2398
 2399 OUTPUT 1 TEXT<|im_end|>
 2400 <|im_start|>user
 2401 Generate<|im_end|>
 2402 <|im_start|>assistant
 2403 <think>
 2404 </think>
 2405
 2406 OUTPUT 2 TEXT<|im_end|>
 2407 <|im_start|>user
 2408 Generate<|im_end|>
 2409 <|im_start|>assistant
 2410 <think>
 2411 </think>
 2412
 2413 OUTPUT 3 TEXT<|im_end|>

2414 gemma-3 (task w/inputs)

2415 <start_of_turn>user
 2416 You are tasked with generating outputs from a particular, potentially
 2417 → stochastic, generative process. You will be given some combination of
 2418 → :
 2419 - Description: A natural description of the generative process / data
 2420 → distribution
 2421 - Input: An input on which to condition the generative process.
 2422 - Example outputs: Example outputs from the process, either in a user
 2423 → message or as prior generations from a chat message. You may assume
 2424 → that any given outputs are exchangeable with one another (order-
 2425 → invariant) and generated from the same process (roughly i.i.d.). If
 2426 → the output data pertains to a single object, it just contains the
 2427 → output. If it contains multiple objects, use json formatting with
 2428 → keys for the name of the output variable.
 2429 You will be provided at least either a description or an example output.

Given these components, your job is to generate JUST the output in your
 → response, roughly approximating the underlying generative process,

2430 → maintaining any underlying stochasticity (if any is present). If you
 2431 → are asked to generate again, you will either be given an additional
 2432 → input to condition on, or will just be told to "Generate".
 2433

2434 Description: DESCRIPTION TEXT
 2435

2436 INPUT 1 TEXT<end_of_turn>
 2437 <start_of_turn>model
 2438 OUTPUT 1 TEXT<end_of_turn>
 2439 <start_of_turn>user
 2440 INPUT 2 TEXT<end_of_turn>
 2441 <start_of_turn>model
 2442 OUTPUT 2 TEXT<end_of_turn>
 2443 <start_of_turn>user
 2444 INPUT 3 TEXT<end_of_turn>
 2445 <start_of_turn>model
 2446 OUTPUT 3 TEXT<end_of_turn>

2446 gemma-3 (task w/out inputs)
 2447

2448 <start_of_turn>user
 2449 You are tasked with generating outputs from a particular, potentially
 2450 → stochastic, generative process. You will be given some combination of
 2451 → :
 2452 - Description: A natural description of the generative process / data
 2453 → distribution
 2454 - Input: An input on which to condition the generative process.
 2455 - Example outputs: Example outputs from the process, either in a user
 2456 → message or as prior generations from a chat message. You may assume
 2457 → that any given outputs are exchangeable with one another (order-
 2458 → invariant) and generated from the same process (roughly i.i.d.). If
 2459 → the output data pertains to a single object, it just contains the
 2460 → output. If it contains multiple objects, use json formatting with
 2461 → keys for the name of the output variable.
 2462 You will be provided at least either a description or an example output.
 2463
 2464 Given these components, your job is to generate JUST the output in your
 2465 → response, roughly approximating the underlying generative process,
 2466 → maintaining any underlying stochasticity (if any is present). If you
 2467 → are asked to generate again, you will either be given an additional
 2468 → input to condition on, or will just be told to "Generate".
 2469

2467 Description: DESCRIPTION TEXT
 2468

2469 Generate<end_of_turn>
 2470 <start_of_turn>model
 2471 OUTPUT 1 TEXT<end_of_turn>
 2472 <start_of_turn>user
 2473 Generate<end_of_turn>
 2474 <start_of_turn>model
 2475 OUTPUT 2 TEXT<end_of_turn>
 2476 <start_of_turn>user
 2477 Generate<end_of_turn>
 2478 <start_of_turn>model
 2479 OUTPUT 3 TEXT<end_of_turn>

2480 Llama-3.1 (task w/inputs)
 2481

2482 <|begin_of_text|><|start_header_id|>system<|end_header_id|>
 2483 Cutting Knowledge Date: December 2023
 Today Date: DD MM YYYY

2484
 2485 You are tasked with generating outputs from a particular, potentially
 2486 → stochastic, generative process. You will be given some combination of
 2487 → :
 2488 - Description: A natural description of the generative process / data
 2489 → distribution
 2490 - Input: An input on which to condition the generative process.
 2491 - Example outputs: Example outputs from the process, either in a user
 2492 → message or as prior generations from a chat message. You may assume
 2493 → that any given outputs are exchangeable with one another (order-
 2494 → invariant) and generated from the same process (roughly i.i.d.). If
 2495 → the output data pertains to a single object, it just contains the
 2496 → output. If it contains multiple objects, use json formatting with
 2497 → keys for the name of the output variable.
 2498 You will be provided at least either a description or an example output.
 2499
 2500 Given these components, your job is to generate JUST the output in your
 2501 → response, roughly approximating the underlying generative process,
 2502 → maintaining any underlying stochasticity (if any is present). If you
 2503 → are asked to generate again, you will either be given an additional
 2504 → input to condition on, or will just be told to "Generate".
 2505
 2506 Description: DESCRIPTION TEXT<|eot_id|><|start_header_id|>user<|
 2507 → end_header_id|>
 2508 INPUT 1 TEXT<|eot_id|><|start_header_id|>assistant<|end_header_id|>
 2509 OUTPUT 1 TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
 2510 INPUT 2 TEXT<|eot_id|><|start_header_id|>assistant<|end_header_id|>
 2511 OUTPUT 2 TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
 2512 INPUT 3 TEXT<|eot_id|><|start_header_id|>assistant<|end_header_id|>
 2513 OUTPUT 3 TEXT<|eot_id|>
 2514
 2515

2516 Llama-3.1 (task w/out inputs)

2517 <|begin_of_text|><|start_header_id|>system<|end_header_id|>
 2518
 2519 Cutting Knowledge Date: December 2023
 2520 Today Date: DD MM YYYY
 2521
 2522 You are tasked with generating outputs from a particular, potentially
 2523 → stochastic, generative process. You will be given some combination of
 2524 → :
 2525 - Description: A natural description of the generative process / data
 2526 → distribution
 2527 - Input: An input on which to condition the generative process.
 2528 - Example outputs: Example outputs from the process, either in a user
 2529 → message or as prior generations from a chat message. You may assume
 2530 → that any given outputs are exchangeable with one another (order-
 2531 → invariant) and generated from the same process (roughly i.i.d.). If
 2532 → the output data pertains to a single object, it just contains the
 2533 → output. If it contains multiple objects, use json formatting with
 2534 → keys for the name of the output variable.
 2535 You will be provided at least either a description or an example output.
 2536
 2537 Given these components, your job is to generate JUST the output in your
 2538 → response, roughly approximating the underlying generative process,
 2539 → maintaining any underlying stochasticity (if any is present). If you
 2540 → are asked to generate again, you will either be given an additional
 2541 → input to condition on, or will just be told to "Generate".

```

2538
2539
2540 Description: DESCRIPTION TEXT<|eot_id|><|start_header_id|>user<|
2541   ↳ end_header_id|>
2542
2543 Generate<|eot_id|><|start_header_id|>assistant<|end_header_id|>
2544 OUTPUT 1 TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
2545
2546 Generate<|eot_id|><|start_header_id|>assistant<|end_header_id|>
2547 OUTPUT 2 TEXT<|eot_id|><|start_header_id|>user<|end_header_id|>
2548
2549 Generate<|eot_id|><|start_header_id|>assistant<|end_header_id|>
2550
2551 OUTPUT 3 TEXT<|eot_id|>
2552

```

2553 **Best performing instruct prompts**

2554 We found that Llama-3.1-8B-Instruct performed best on SPECTRUM SUITE with the pre-
2555 trained prompt, google/gemma-3-12b-bit and qwen/Qwen3-14B performed best with the
2556 detailed instruct prompt. We utilize those prompts with the corresponding models for all ICL exper-
2557 iments.

2558 M OUTPUT COVERAGE / DIVERSITY VS. VALIDITY EXPERIMENT DETAILS

2561 M.1 VERIFIABLE EVALUATION

2563 For this evaluation, we utilize the same prompts as in the ICL experiments - see App. L.

2564 Below, we include the description and examples for each of the tasks. Please reference the codebase
2565 for validation functions.

```

2566 Task: color_interesting_ex
2567 Description: Generate a color name.
2568 Examples: ['Otterly Brown', 'Petal Pink', 'Cherry']
2569
2570 Task: color_normal_ex
2571 Description: Generate a color name.
2572 Examples: ['Green', 'Red', 'White']
2573
2574 Task: car_brand
2575 Description: Car brand.
2576 Examples: ['Acura', 'Ford', 'Tesla']
2577
2578 Task: car_make_model
2579 Description: Car make and model.
2580 Examples: ['Acura Integra', 'Ford Mustang', 'Tesla Model 3']
2581
2582 Task: us_states_abbreviations
2583 Description: US state abbreviation
2584 Examples: ['KY', 'UT', 'OR']
2585
2586 Task: us_states_any_format
2587 Description: US state name or abbreviation
2588 Examples: ['Kentucky', 'UT', 'Oregon']
2589
2590 Task: us_states_full_names
2591 Description: Name a US state
2592 Examples: ['Kentucky', 'Utah', 'Oregon']
2593
2594 Task: prime_numbers
2595 Description: Generate a prime number
2596 Examples: ['617', '13', '47']

```

```

2592
2593 Task: small_prime_numbers
2594 Description: Generate a prime number less than 100
2595 Examples: ['29', '5', '97']
2596
2597 Task: basic_emails
2598 Description: Email address
2599 Examples: ['ANONYMIZED', 'alex.jones@domain.net', 'itsagoodday@gmail.com
2600 → ']
2601
2602 Task: professional_emails
2603 Description: Generate a professional email address.
2604 Examples: ['ANONYMIZED', 'sarah.johannesburg@organization.org', '→ yash@anthropic.com']
2605
2606 Task: weekdays_abbreviated
2607 Description: Day of the week abbreviation
2608 Examples: ['Thu', 'Wed.', 'SUN']
2609
2610 Task: weekdays_any_format
2611 Description: Day of the week (full name or abbreviation)
2612 Examples: ['Monday', 'Tue', 'SUN']
2613
2614 Task: weekdays_full
2615 Description: Name a day of the week
2616 Examples: ['Thursday', 'Wednesday', 'Sunday']
2617
2618 Task: random_seed
2619 Description: Generate a number to use for a random seed.
2620 Examples: ['15', '420', '8392013']
2621
2622 Task: claudie_gerunds
2623 Description: Generate an English gerund ending in -ing.
2624 Examples: ['Schlepping', 'Hoisting', 'Thinking']
2625
2626 Task: rng_1_10
2627 Description: Generate a number between 1 and 10.
2628 Examples: ['3', '7', '10']
2629
2630 Task: rng_1_100
2631 Description: Generate a number between 1 and 100.
2632 Examples: ['35', '94', '71']
2633
2634 Task: international_phone_numbers
2635 Description: International phone number with country code.
2636 Examples: ['+1 413-121-2591', '+44 10 2958 3938', '+81 3 8328 5625']
2637 Task: us_phone_numbers
2638 Description: US phone number
2639 Examples: ['(210) 346-0967', '646 562-1938', '800-895-0522']
2640
2641

```

M.2 IN-CONTEXT CHAT ELICITATION FOR PRETRAINED AND INSTRUCTION-TUNED MODELS

2642 For inducing chat capabilities from the pretrained/spectrum-tuned models for the
2643 NoveltyBench/Infinite-Chats-Eval experiments, we utilize the following few-shot ex-

amples. Responses are generated by an instruct model in the same model family: [google/gemma-3-27b-it](https://huggingface.co/google/gemma-3-27b-it), [qwen/Qwen3-32B](https://huggingface.co/qwen/Qwen3-32B), [meta/Llama-3.1-70B-Instruct](https://huggingface.co/meta/Llama-3.1-70B-Instruct).

2649 For all pretrained models, we utilized the following prompt template:

You are a helpful AI assistant.

User: <EXAMPLE PROMPT1>

Assistant: <EXAMPLE RESPONSE1>

...

User: <INFERENCE PROMPT>

Assistant:

For the spectrum-tuned models, we use You are a helpful AI assistant. as the description, the example prompt as input, and the example response as output, using the corresponding model's **SPECTRUM TUNING** template.

Below, we include the demonstrations for each model family.

gemma-3-12b

```
{"prompt": "What is the capital of France?", "response": "The capital of France is **Paris**. \n\nIt's known for its iconic landmarks like the Eiffel Tower, Louvre Museum, and Arc de Triomphe, as well as its fashion, cuisine, and culture."}  
{"prompt": "What color should I paint my car? Respond with just a color", "response": "Forest green."}  
{"prompt": "Explain the 2008 financial crisis to me in about one paragraph.", "response": "The 2008 financial crisis was a severe worldwide economic downturn triggered by the collapse of the US housing market. Years of low interest rates and lax lending standards led to a boom in subprime mortgages and loans given to borrowers with poor credit. These mortgages were packaged into complex financial products called mortgage-backed securities and sold to investors globally. When housing prices began to fall in 2006-2007, many borrowers defaulted, causing these securities to lose value. This triggered a cascading effect: banks and financial institutions holding these assets faced massive losses, credit markets froze, businesses couldn't get loans, and ultimately, major institutions like Lehman Brothers collapsed. The crisis spread globally, leading to a severe recession, job losses, and government bailouts to stabilize the financial system."}  
{"prompt": "Tell me a joke about Texas.", "response": "Why did the Texan bring a ladder to the rodeo? \n\n...He heard the steaks were high! \n\n\n<Face with Tears of Joy emoji>\n\n\nHope that gave you a little chuckle! Let me know if you'd like to hear another one."}
```

Llama-3.1-8B

```
{"prompt": "What is the capital of France?", "response": "The capital of  
→ France is Paris."}  
{ "prompt": "What color should I paint my car? Respond with just a color",  
→ "response": "Navy blue"}  
{ "prompt": "Explain the 2008 financial crisis to me in about one  
→ paragraph.", "response": "The 2008 financial crisis, also known as  
→ the Global Financial Crisis (GFC), was a severe economic downturn  
→ that began in the United States in 2007 and spread globally, lasting  
→ through 2009. The crisis was triggered by a housing market bubble  
→ bursting in the US, fueled by lax lending practices and excessive  
→ speculation. Many homeowners, especially those with subprime  
→ mortgages, were unable to afford their mortgage payments, leading to
```

```

2700    ↪ widespread defaults and foreclosures. This created a crisis in the
2701    financial markets, as many banks and financial institutions had
2702    invested heavily in mortgage-backed securities (MBS) that were now
2703    worthless. As a result, these institutions found themselves facing
2704    massive losses, which in turn led to a credit crisis, stock market
2705    crash, and a global recession. The crisis was exacerbated by other
2706    factors, including over-reliance on complex financial instruments,
2707    inadequate regulation, and a global economic imbalance. The crisis
2708    led to unprecedented government interventions, including massive
2709    bailouts and stimulus packages, to stabilize the financial system and
2710    stimulate economic growth."}
2711 {"prompt": "Tell me a joke about Texas.", "response": "Why did the Texan
2712    bring a ladder to the party? \n\nBecause he heard the drinks were on
2713    the house."}
2714

```

Qwen3-14B

```

2715 {"prompt": "What is the capital of France?", "response": "The capital of
2716    ↪ France is **Paris**."}
2717 {"prompt": "What color should I paint my car? Respond with just a color",
2718    ↪ "response": "Crimson Red"}
2719 {"prompt": "Explain the 2008 financial crisis to me in about one
2720    ↪ paragraph.", "response": "The 2008 financial crisis was triggered by
2721    ↪ a housing market bubble in the United States that burst, leading to a
2722    ↪ severe credit crunch and global economic downturn. Starting in the
2723    ↪ early 2000s, lax lending standards and speculative investment fueled
2724    ↪ a rapid rise in home prices. Banks and financial institutions began
2725    ↪ offering high-risk subprime mortgages to borrowers with poor credit,
2726    ↪ then bundled these loans into complex financial products like
2727    ↪ mortgage-backed securities (MBS) and collateralized debt obligations
2728    ↪ (CDOs), which were widely traded. When housing prices peaked and
2729    ↪ began to fall in 2006-2007, many borrowers defaulted on their
2730    ↪ mortgages, causing the value of these financial instruments to
2731    ↪ plummet. This led to massive losses for banks and investors,
2732    ↪ triggering a wave of bankruptcies and failures, including major
2733    ↪ institutions like Lehman Brothers. The crisis spread globally as
2734    ↪ financial systems were interconnected, resulting in a deep recession
2735    ↪ with soaring unemployment, foreclosures, and a widespread loss of
2736    ↪ consumer and business confidence. In response, governments and
2737    ↪ central banks implemented massive bailouts and stimulus measures to
2738    ↪ stabilize the economy."}
2739 {"prompt": "Tell me a joke about Texas.", "response": "Sure! Here's a
2740    ↪ Texas-themed joke for you:\n\nWhy don't cowboys in Texas ever get
2741    ↪ cold?\n\nBecause the only *draft* they know is from the beer and the
2742    ↪ wind... and even that's a *light beer breeze*!"}
2743

```

N DISTRIBUTIONAL ALIGNMENT EXPERIMENT DETAILS

N.1 EXAMPLE PROMPTS

2744 Here, we include example prompts for the urn task across spectrum/pretrained/instruction-tuned
2745 models. The same prompts were used for all model families and tasks, only varying the task and
2746 output descriptions.

spectrum

```

2748 <start_of_turn>description
2749 There is an urn with the following balls shuffled together: 1 red ball, 2
2750    ↪ blue balls, 4 orange balls and 6 yellow balls.<end_of_turn>
2751 <start_of_turn>input
2752 Draw a ball at random, and tell me the color (lowercase).<end_of_turn>
2753 <start_of_turn>output

```

2754
 2755 Target outputs: ['red' 'blue' 'orange' 'yellow']
 2756 Target distribution: [0.07692307692307693, 0.15384615384615385,
 2757 → 0.3076923076923077, 0.46153846153846156]

2758
 2759 **pretrained**

2760 Description: There is an urn with the following balls shuffled together:
 2761 → 1 red ball, 2 blue balls, 4 orange balls and 6 yellow balls.
 2762
 2763 Input: Draw a ball at random, and tell me the color (lowercase).
 2764
 2765 Output:

2766
 2767 Target outputs: ['red' 'blue' 'orange' 'yellow']
 2768 Target distribution: [0.07692307692307693, 0.15384615384615385,
 2769 → 0.3076923076923077, 0.46153846153846156]

2770
 2771 **chat**

2772 <start_of_turn>user
 2773 Given the description and an input, respond with just an output.
 2774
 2775 Description: There is an urn with the following balls shuffled together:
 2776 → 1 red ball, 2 blue balls, 4 orange balls and 6 yellow balls.
 2777
 2778 Draw a ball at random, and tell me the color (lowercase).<end_of_turn>
 2779 <start_of_turn>model

2780
 2781 Target outputs: ['red' 'blue' 'orange' 'yellow']
 2782 Target distribution: [0.07692307692307693, 0.15384615384615385,
 2783 → 0.3076923076923077, 0.46153846153846156]

2784
 2785 **N.2 TASK EXAMPLES**

2786 Here, we show an example data instance from each distributional alignment dataset.

2787
 2788 **global_oqa**

2789 <start_of_turn>description
 2790 Responses from a person from this country: Uruguay<end_of_turn>
 2791 <start_of_turn>input
 2792 Please tell me for each of the following statements whether you think it
 2793 → can always be justified, never be justified, or something in between,
 2794 → using this card.
 2795
 2796 Terrorism as a political, ideological or religious mean
 2797 Options:
 2798 A. Never justifiable
 2799 B. 2
 2800 C. 3
 2801 D. 4
 2802 E. 5
 2803 F. 6
 2804 G. 7
 2805 H. 8
 2806 I. 9
 2807 J. Always justifiable
 2808 K. Don't know
 2809 L. No answer
 2810 M. Missing; Not available<end_of_turn>
 2811 <start_of_turn>output

```

2808
2809 Target outputs: ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M']
2810
2811 Target distribution: [0.8281718281718281, 0.046953046953046945,
2812   ↪ 0.020979020979020976, 0.012987012987012986, 0.01898101898101898,
2813   ↪ 0.004995004995004994, 0.011988011988011986, 0.007992007992007992,
2814   ↪ 0.013986013986013983, 0.014985014985014982, 0.009990009990009988,
2815   ↪ 0.007992007992007992, 0.0]
2816
2817 nytimes_books
2818 <start_of_turn>description
2819 You are a random survey respondent. Respondent demographic: Democrat<
2820   ↪ end_of_turn>
2821 <start_of_turn>input
2822 How likely are you to read this book?
2823 Book: The Bee Sting By Paul Murray
2824 Genre: Fiction
2825 Summary: In Murray's boisterous tragicomic novel, a once wealthy Irish
2826   ↪ family struggles with both the aftermath of the 2008 financial crash
2827   ↪ and their own inner demons.
2828 Options:
2829 1: Very unlikely
2830 2: Somewhat unlikely
2831 3: Somewhat likely
2832 4: Very likely<end_of_turn>
2833 <start_of_turn>output
2834
2835 Target outputs: ['1', '2', '3', '4']
2836 Target distribution: [0.45, 0.3, 0.15, 0.1]
2837
2838 Machine Personality Inventory
2839 <start_of_turn>description
2840 You are a random survey respondent. Rate your agreement with the
2841   ↪ following statement on a scale from 0-5, where 0 means "strongly
2842   ↪ disagree" and 5 means "strongly agree."<end_of_turn>
2843 <start_of_turn>input
2844 Statement: You worry about things
2845 Options:
2846 0: Strongly disagree
2847 1: Disagree
2848 2: Slightly disagree
2849 3: Slightly agree
2850 4: Agree
2851 5: Strongly agree<end_of_turn>
2852 <start_of_turn>output
2853
2854 Target outputs: ['0', '1', '2', '3', '4', '5']
2855 Target distribution: [0.0012678672373415167, 0.04066219817491722,
2856   ↪ 0.14105467172736816, 0.13365904869579262, 0.4568359848178955,
2857   ↪ 0.22652022934668498]
2858
2859 Rotten Tomatoes
2860 <start_of_turn>description
2861 You are a movie critic. Given a movie, you are asked to simply rate it as
2862   ↪ "Good" or "Bad".<end_of_turn>
2863 <start_of_turn>input
2864 Movie: Rambo III
2865 Release Date: Released May 25, 1988<end_of_turn>
2866 <start_of_turn>output

```

2862

2863 Target outputs: ['Good', 'Bad']
 2864 Target distribution: [0.41, 0.59]

2865

Habermas

2867

<start_of_turn>description
 You are a randomly selected UK resident. You will be given a question and
 ↪ two statements, A and B. Rate which statement you most agree with on
 ↪ a likert scale from 1 to 7:
 1: Strongly Agree with A
 2: Agree with A
 3: Somewhat Agree with A
 4: Neutral
 5: Somewhat Agree with B
 6: Agree with B
 7: Strongly Agree with B<end_of_turn>
<start_of_turn>input
 Question: Should we ban right turns in central London?
 A: We should ban right turns in central London.
 B: We should NOT ban right turns in central London.<end_of_turn>
<start_of_turn>output

2881

2882 Target outputs: ['1', '2', '3', '4', '5', '6', '7']
 2883 Target distribution: [0.0, 0.0, 0.04, 0.24, 0.08, 0.16, 0.48]

2884

Numbergame

2885

<start_of_turn>description
 You are a randomly selected participant in a study. You will be given a
 ↪ set of numbers which all belong to the same set or pattern, and will
 ↪ be given a target number which may or may not belong to the same set
 ↪ or pattern. Answer Yes if you think that the target number belongs to
 ↪ the same set, otherwise answer No.<end_of_turn>
<start_of_turn>input
 Example set: 84, 94, 34
 Target number: 5<end_of_turn>
<start_of_turn>output

2896

2897 Target outputs: ['Yes', 'No']
 2898 Target distribution: [0.0, 1.0]

2899

Urn

2900

<start_of_turn>description
 There is an urn with the following balls shuffled together: 1 red ball, 2
 ↪ blue balls, 4 orange balls and 6 yellow balls.<end_of_turn>
<start_of_turn>input
 Draw a ball at random, and tell me the color (lowercase).<end_of_turn>
<start_of_turn>output

2907

2908 Target outputs: ['red', 'blue', 'orange', 'yellow']
 2909 Target distribution: [0.07692307692307693, 0.15384615384615385,
 ↪ 0.3076923076923077, 0.46153846153846156]

2910

2911

2912

O EXAMPLE TASK PROMPTS

2913

2914

2915

For brevity, here we include example prompts (specifically, for the Gemma models) only for the test split of SPECTRUM SUITE. To see examples for all tasks, please reference: ANONYMIZED

2916 0.1 SPECTRUM SUITE-TEST: MULTIPLE-CHOICE
29172918 **chatbotarena_individual_prefs**
2919

```

2920 <start_of_turn>description
2921 The following are preferences for a single individual. They will prompt a
2922 → language model (opening_prompt), and then will compare the two
2923 → responses (model_a_response and model_b_response). The user will then
2924 → rate the winner, with one of the following options: ['b' 'tie' 'a' '
2925 → bothbad']<end_of_turn>
2926 <start_of_turn>input
2927 {"opening_prompt": "What does the saying \"pipeline is lifeline\" mean?", "
2928 → "model_a_response": "The saying \"pipeline is lifeline\" refers to
2929 → the importance of having a reliable and efficient system for
2930 → transporting resources. In the context of oil and gas, a pipeline is
2931 → a critical part of the supply chain, as it allows companies to
2932 → transport oil and gas from production sites to refineries and
2933 → distribution terminals. A reliable pipeline system is essential to
2934 → ensuring that oil and gas are available when and where they are
2935 → needed.\n\nThe saying \"pipeline is lifeline\" can also be used in a
2936 → more general sense to refer to any system that is essential to the
2937 → functioning of an organization or society. For example, the
2938 → transportation system is often referred to as the \"lifeline of the
2939 → economy,\" as it allows goods and services to be moved around the
2940 → country. The power grid is another example of a lifeline system, as
2941 → it provides electricity to homes and businesses.\n\nIn any context,
2942 → the saying \"pipeline is lifeline\" emphasizes the importance of
2943 → having a reliable system in place to ensure that essential resources
2944 → are available.", "model_b_response": "The phrase \"pipeline is
2945 → lifeline\" is a metaphorical expression that highlights the
2946 → importance of a pipeline in supplying goods, services, or information
2947 → from one place to another. A pipeline can be physical or virtual,
2948 → and it plays a crucial role in keeping the flow of goods, services,
2949 → or information moving smoothly and efficiently. Without a pipeline,
2950 → the supply chain would be disrupted, leading to a loss of revenue,
2951 → increased costs, and potential shortages. In other words, a pipeline
2952 → is critical to the survival and success of a business or industry,
2953 → just as a lifeline is critical to the survival of a person."}<
2954 → end_of_turn>
2955 <start_of_turn>output
2956 a<end_of_turn>
2957 <start_of_turn>input
2958 {"opening_prompt": "How do I use rava while playing Mridangam?", "
2959 → "model_a_response": "Rava is a type of grain used in the playing of
2960 → the Mridangam, a double-headed South Indian drum. Here are the main
2961 → ways rava is used while playing Mridangam:\n\n1. Applying rava to the
2962 → drumheads. Mridangam players sprinkle rava on the drumheads to
2963 → change the tone and timbre of the drum. The rava acts as a muffling
2964 → agent and dampens the vibrations of the drumheads, producing a softer
2965 → tone.\n\n2. Applying rava to the fingers. Mridangam players apply a
2966 → thin layer of rava to their fingertips before playing. This helps the
2967 → fingers slide smoothly on the drumheads and produces a distinct
2968 → sound. \n\n3. Using rava to dampen or muffle certain notes. Mridangam
2969 → players will sprinkle extra rava on specific spots on the drumheads
→ to dampen or muffle certain notes they are not playing at that moment
→ . This helps isolate and emphasize the notes that are being played.\n
→ \n4. Adding more rava for a softer tone. Mridangam players can
→ sprinkle more rava on the drumheads during softer passages to produce
→ a mellower, more muted tone. Less rava is used for louder sections.\n
→ \n5. Adjusting rava during performance. Mridangam players may add or
→ remove rava from the drumheads as needed during a performance to get
→ the precise tone and sound they desire for different sections of the
→ music.\n\nSo in summary, rava is used to control the tone, timbre,
→ volume and articulation of the notes produced by the Mridangam
→ through muffling and damping of the drumheads. The amount and

```

```

2970   ↪ placement of rava can be adjusted to suit the needs of the
2971   ↪ performance.", "model_b_response": "Rava is a Hindu musical scale
2972   ↪ that is commonly played on the mridangam. It is composed of two main
2973   ↪ scales, one for the higher-pitched notes (tuwar) and another for the
2974   ↪ lower-pitched notes (nadi).\nTo play Rava on the mridangam, you can
2975   ↪ follow these steps:\n1. Start with the higher-pitched notes (tuwar)
2976   ↪ of the mridangam, and gradually work your way down to the lower-
2977   ↪ pitched notes (nadi).\n2. Move in a circular motion around the
2978   ↪ outside of the mridangam as you play the raava.\n3. When you're
2979   ↪ finished playing the raava, turn off the instrument and let it rest
2980   ↪ for a few minutes. This allows the raga to settle in your mind before
2981   ↪ moving on to the next note.\n4. Repeat this process until you reach
2982   ↪ the highest point of the scale, when you'll stop using Rava
2983   ↪ altogether.\nOnce you've mastered the raava, you can use it in any
2984   ↪ style you like - from classical Indian music to modern jazz, rock, or
2985   ↪ electronic music. Just make sure to find a suitable voice and pitch
2986   ↪ to match the raga you're playing, as well as to avoid overusing it
2987   ↪ ."}<end_of_turn>
2988 ...

```

flight

```

2989 <start_of_turn>description
2990 The following express flight preferences for the same individual among a
2991   ↪ set of flights. Predict which flight the individual prefers.<
2992   ↪ end_of_turn>
2993 <start_of_turn>input
2994 Flight 1:
2995 Departure Time: 09:36 AM, Duration: 11 hr 41 min, Number of Stops: 1,
2996   ↪ Price: $500.00
2997 Flight 2:
2998 Departure Time: 01:38 PM, Duration: 8 hr 27 min, Number of Stops: 1,
2999   ↪ Price: $1450.00
3000 Flight 3:
3001 Departure Time: 03:56 PM, Duration: 4 hr 26 min, Number of Stops: 1,
3002   ↪ Price: $1270.00<end_of_turn>
3003 <start_of_turn>output
3004 1<end_of_turn>
3005 <start_of_turn>input
3006 Flight 1:
3007 Departure Time: 10:10 AM, Duration: 9 hr 13 min, Number of Stops: 2,
3008   ↪ Price: $1430.00
3009 Flight 2:
3010 Departure Time: 08:50 AM, Duration: 13 hr 59 min, Number of Stops: 0,
3011   ↪ Price: $920.00
3012 Flight 3:
3013 Departure Time: 07:06 AM, Duration: 13 hr 13 min, Number of Stops: 2,
3014   ↪ Price: $1530.00<end_of_turn>
3015 <start_of_turn>output
3016 1<end_of_turn>
3017 <start_of_turn>input
3018 Flight 1:
3019 Departure Time: 10:22 AM, Duration: 14 hr 36 min, Number of Stops: 0,
3020   ↪ Price: $1330.00
3021 Flight 2:
3022 Departure Time: 11:25 PM, Duration: 3 hr 31 min, Number of Stops: 1,
3023   ↪ Price: $860.00
3024 Flight 3:
3025 Departure Time: 07:23 PM, Duration: 3 hr 12 min, Number of Stops: 0,
3026   ↪ Price: $790.00<end_of_turn>
3027 <start_of_turn>output
3028 2<end_of_turn>
3029 <start_of_turn>input
3030 Flight 1:

```

```

3024 Departure Time: 07:29 AM, Duration: 0 hr 45 min, Number of Stops: 1,
3025   ↗ Price: $1670.00
3026 Flight 2:
3027 Departure Time: 08:50 AM, Duration: 15 hr 13 min, Number of Stops: 2,
3028   ↗ Price: $1040.00
3029 Flight 3:
3030 Departure Time: 10:16 PM, Duration: 15 hr 50 min, Number of Stops: 1,
3031   ↗ Price: $1370.00<end_of_turn>
3031 <start_of_turn>output
3032 2<end_of_turn>
3033 <start_of_turn>input
3034 Flight 1:
3035 Departure Time: 09:24 AM, Duration: 11 hr 31 min, Number of Stops: 0,
3036   ↗ Price: $1920.00
3036 Flight 2:
3037 Departure Time: 08:38 AM, Duration: 14 hr 27 min, Number of Stops: 1,
3038   ↗ Price: $600.00
3039 Flight 3:
3040 Departure Time: 05:57 AM, Duration: 11 hr 59 min, Number of Stops: 1,
3041   ↗ Price: $850.00<end_of_turn>
3041 <start_of_turn>output
3042 2<end_of_turn>
3043 <start_of_turn>input
3044 Flight 1:
3045 Departure Time: 08:15 AM, Duration: 1 hr 58 min, Number of Stops: 0,
3046   ↗ Price: $760.00
3046 Flight 2:
3047 Departure Time: 05:28 PM, Duration: 3 hr 59 min, Number of Stops: 0,
3048   ↗ Price: $1010.00
3048 Flight 3:
3049 Departure Time: 12:29 PM, Duration: 4 hr 45 min, Number of Stops: 1,
3050   ↗ Price: $820.00<end_of_turn>
3051 <start_of_turn>output
3052 3<end_of_turn>
3053 <start_of_turn>input
3054 Flight 1:
3055 Departure Time: 12:40 PM, Duration: 10 hr 45 min, Number of Stops: 2,
3056   ↗ Price: $1340.00
3056 Flight 2:
3057 Departure Time: 04:07 PM, Duration: 14 hr 18 min, Number of Stops: 2,
3058   ↗ Price: $1120.00
3058 Flight 3:
3059 Departure Time: 06:37 PM, Duration: 7 hr 22 min, Number of Stops: 2,
3060   ↗ Price: $1360.00<end_of_turn>
3061 <start_of_turn>output
3062 1<end_of_turn>
3063 <start_of_turn>input
3064 Flight 1:
3065 Departure Time: 12:52 PM, Duration: 9 hr 22 min, Number of Stops: 1,
3066   ↗ Price: $1430.00
3066 Flight 2:
3067 Departure Time: 10:50 PM, Duration: 14 hr 36 min, Number of Stops: 2,
3068   ↗ Price: $1750.00
3069 Flight 3:
3070 Departure Time: 08:38 AM, Duration: 9 hr 50 min, Number of Stops: 0,
3071   ↗ Price: $860.00<end_of_turn>
3071 <start_of_turn>output
3072 2<end_of_turn>
3073 <start_of_turn>input
3074 Flight 1:
3075 Departure Time: 06:09 AM, Duration: 11 hr 13 min, Number of Stops: 0,
3076   ↗ Price: $610.00
3076 Flight 2:
3077 Departure Time: 02:12 PM, Duration: 9 hr 13 min, Number of Stops: 2,
3077   ↗ Price: $540.00

```

```

3078 Flight 3:
3079 Departure Time: 11:31 AM, Duration: 6 hr 45 min, Number of Stops: 1,
3080   ↪ Price: $1110.00<end_of_turn>
3081 <start_of_turn>output
3082 2<end_of_turn>
3083 <start_of_turn>input
3084 Flight 1:
3085 Departure Time: 04:07 PM, Duration: 10 hr 55 min, Number of Stops: 2,
3086   ↪ Price: $920.00
3087 Flight 2:
3088 Departure Time: 07:29 AM, Duration: 7 hr 3 min, Number of Stops: 0, Price
3089   ↪ : $1510.00
3090 Flight 3:
3091 Departure Time: 06:43 AM, Duration: 11 hr 13 min, Number of Stops: 1,
3092   ↪ Price: $1680.00<end_of_turn>
3093 <start_of_turn>output
3094 1<end_of_turn>
3095 <start_of_turn>input
3096 Flight 1:
3097 Departure Time: 10:04 PM, Duration: 7 hr 40 min, Number of Stops: 2,
3098   ↪ Price: $1870.00
3099 Flight 2:
3100 Departure Time: 01:15 PM, Duration: 8 hr 45 min, Number of Stops: 1,
3101   ↪ Price: $1480.00
3102 Flight 3:
3103 Departure Time: 06:20 AM, Duration: 4 hr 54 min, Number of Stops: 0,
3104   ↪ Price: $1260.00<end_of_turn>
3105 ...

```

3103 habermas_individual_categorical

```

3104 <start_of_turn>description
3105 Given a question and a statement, predict the level of agreement with it
3106   ↪ on a 7-point scale.
3107 Options: Strongly Agree; Agree; Somewhat Agree; Neutral; Somewhat
3108   ↪ Disagree; Disagree; Strongly Disagree<end_of_turn>
3109 <start_of_turn>input
3110 {"question.text": "Should the government provide a basic income of GBP
3111   ↪ 1000 per month to everyone?", "statement": "The government should
3112   ↪ provide a basic income of GBP 1000 per month to everyone."}<
3113   ↪ end_of_turn>
3114 <start_of_turn>output
3115 Strongly Agree<end_of_turn>
3116 <start_of_turn>input
3117 {"question.text": "Is it a good idea to further reduce taxation on
3118   ↪ corporations?", "statement": "It is a good idea to further reduce
3119   ↪ taxation on corporations."}<end_of_turn>
3120 <start_of_turn>output
3121 Somewhat Disagree<end_of_turn>
3122 <start_of_turn>input
3123 {"question.text": "Should we ban the use of artificial sweeteners in food
3124   ↪ and drink?", "statement": "We should ban the use of artificial
3125   ↪ sweeteners in food and drink."}<end_of_turn>
3126 <start_of_turn>output
3127 Agree<end_of_turn>
3128 <start_of_turn>input
3129 {"question.text": "Should we change our economic system from capitalism
3130   ↪ to socialism?", "statement": "We should change our economic system
3131   ↪ from capitalism to socialism."}<end_of_turn>
3132 <start_of_turn>output
3133 Neutral<end_of_turn>
3134 <start_of_turn>input
3135 {"question.text": "Are celebrities good role models?", "statement": "
3136   ↪ Celebrities are good role models."}<end_of_turn>
3137 <start_of_turn>output

```

```

3132 Disagree<end_of_turn>
3133 <start_of_turn>input
3134 {"question.text": "Is it the government's role to reduce childhood
3135   ↪ obesity?", "statement": "It is the government's role to reduce
3136   ↪ childhood obesity."}<end_of_turn>
3137 <start_of_turn>output
3138 Somewhat Agree<end_of_turn>
3139 <start_of_turn>input
3140 {"question.text": "Should we move to a form of direct democracy meaning
3141   ↪ that people vote directly on issues via referendums?", "statement": " "
3142   ↪ We should move to a form of direct democracy meaning that people vote
3143   ↪ directly on issues via referendums."}<end_of_turn>
3144 <start_of_turn>output
3145 Agree<end_of_turn>
3146 <start_of_turn>input
3147 {"question.text": "Should the government provide universal free childcare
3148   ↪ from birth?", "statement": "The government should provide universal
3149   ↪ free childcare from birth."}<end_of_turn>
3150 <start_of_turn>output
3151 Strongly Agree<end_of_turn>
3152 <start_of_turn>input
3153 {"question.text": "Should the United Kingdom become a federated republic
3154   ↪ ?", "statement": "The United Kingdom should become a federated
3155   ↪ republic."}<end_of_turn>
3156 <start_of_turn>output
3157 Agree<end_of_turn>
3158 <start_of_turn>input
3159 {"question.text": "Should the UK government pass a law to limit the
3160   ↪ quantity of money that a single person can give to political parties
3161   ↪ or candidates?", "statement": "The UK government should pass a law to
3162   ↪ limit the quantity of money that a single person can give to
3163   ↪ political parties or candidates."}<end_of_turn>
3164 <start_of_turn>output
3165 Agree<end_of_turn>
3166

```

3162 numbergame_individual

```

3163 <start_of_turn>description
3164 The following are given: given_numbers, target_number. You must generate
3165   ↪ target_belongs_to_set.<end_of_turn>
3166 <start_of_turn>input
3167 {"given_numbers": "48, 78, 38, 98", "target_number": "90"}<end_of_turn>
3168 <start_of_turn>output
3169 No<end_of_turn>
3170 <start_of_turn>input
3171 {"given_numbers": "79, 47, 62, 98", "target_number": "46"}<end_of_turn>
3172 <start_of_turn>output
3173 Yes<end_of_turn>
3174 <start_of_turn>input
3175 {"given_numbers": "79, 47, 62, 98", "target_number": "35"}<end_of_turn>
3176 <start_of_turn>output
3177 No<end_of_turn>
3178 <start_of_turn>input
3179 {"given_numbers": "81", "target_number": "55"}<end_of_turn>
3180 <start_of_turn>output
3181 Yes<end_of_turn>
3182 <start_of_turn>input
3183 {"given_numbers": "92, 14, 20, 5", "target_number": "77"}<end_of_turn>
3184 <start_of_turn>output
3185 No<end_of_turn>
3186 <start_of_turn>input
3187 {"given_numbers": "15, 11", "target_number": "44"}<end_of_turn>
3188 <start_of_turn>output
3189 Yes<end_of_turn>
3190 <start_of_turn>input

```

```

3186 {"given_numbers": "48, 78, 38, 98", "target_number": "41"}<end_of_turn>
3187 <start_of_turn>output
3188 No<end_of_turn>
3189 <start_of_turn>input
3190 {"given_numbers": "7, 63", "target_number": "46"}<end_of_turn>
3191 <start_of_turn>output
3192 No<end_of_turn>
3193 <start_of_turn>input
3194 {"given_numbers": "4, 16, 12", "target_number": "63"}<end_of_turn>
3195 <start_of_turn>output
3196 No<end_of_turn>
3197 <start_of_turn>input
3198 {"given_numbers": "31, 3, 1, 15", "target_number": "15"}<end_of_turn>
3199 <start_of_turn>output
3200 No<end_of_turn>
3201 <start_of_turn>input
3202 {"given_numbers": "89", "target_number": "8"}<end_of_turn>
3203 <start_of_turn>output
3204 Yes<end_of_turn>
3205 <start_of_turn>input
3206 {"given_numbers": "3, 63", "target_number": "4"}<end_of_turn>
3207 <start_of_turn>output
3208 No<end_of_turn>
3209 <start_of_turn>input
3210 {"given_numbers": "4, 16, 12", "target_number": "49"}<end_of_turn>
3211 <start_of_turn>output
3212 Yes<end_of_turn>
3213 <start_of_turn>input
3214 {"given_numbers": "48, 78, 38, 98", "target_number": "10"}<end_of_turn>
3215 <start_of_turn>output
3216 No<end_of_turn>
3217 <start_of_turn>input
3218 {"given_numbers": "89", "target_number": "33"}<end_of_turn>
3219 <start_of_turn>output
3220 Yes<end_of_turn>
3221 <start_of_turn>input
3222 {"given_numbers": "31, 3, 1, 15", "target_number": "20"}<end_of_turn>
3223 <start_of_turn>output
3224 No<end_of_turn>
3225 <start_of_turn>input
3226 {"given_numbers": "92, 14, 20, 5", "target_number": "9"}<end_of_turn>
3227 <start_of_turn>output
3228 Yes<end_of_turn>
3229 <start_of_turn>input
3230 {"given_numbers": "52, 24", "target_number": "42"}<end_of_turn>
3231 <start_of_turn>output
3232 No<end_of_turn>
3233 <start_of_turn>input
3234 {"given_numbers": "79, 47, 62, 98", "target_number": "94"}<end_of_turn>
3235 <start_of_turn>output
3236 No<end_of_turn>
3237 <start_of_turn>input
3238 {"given_numbers": "81", "target_number": "26"}<end_of_turn>
3239 <start_of_turn>output
3240 Yes<end_of_turn>
3241 <start_of_turn>input
3242 {"given_numbers": "7, 63", "target_number": "42"}<end_of_turn>

```

```

3240 <start_of_turn>output
3241 No<end_of_turn>
3242 <start_of_turn>input
3243 {"given_numbers": "79, 47, 62, 98", "target_number": "95"}<end_of_turn>
3244 <start_of_turn>output
3245 No<end_of_turn>
3246 <start_of_turn>input
3247 {"given_numbers": "31, 3, 1, 15", "target_number": "35"}<end_of_turn>
3248 <start_of_turn>output
3249 No<end_of_turn>
3250 <start_of_turn>input
3251 {"given_numbers": "48, 78, 38, 98", "target_number": "12"}<end_of_turn>
3252 <start_of_turn>output
3253 No<end_of.turn>...
3254

```

wvs_individual

```

3254 <start_of_turn>description
3255 response ~ question + options<end_of_turn>
3256 <start_of_turn>input
3257 {"question": "Membership: consumer organization", "options": "[{'Other
3258     ↪ missing; Multiple answers Mail (EVS)', 'Not asked', 'No answer', \
3259     ↪ Don't know\\', 'Not mentioned (do not belong)', 'Mentioned (member)
3260     ↪ ']"}<end_of_turn>
3261 <start_of_turn>output
3262 Not mentioned (do not belong)<end_of.turn>
3263 <start_of_turn>input
3264 {"question": "Membership: sport or recreational org", "options": "[{'Other
3265     ↪ missing; Multiple answers Mail (EVS)', 'Not asked', 'No answer', \
3266     ↪ Don't know\\', 'Not mentioned (do not belong)', 'Mentioned (member)
3267     ↪ ']"}<end_of.turn>
3268 <start_of_turn>output
3269 Not mentioned (do not belong)<end_of.turn>
3270 <start_of_turn>input
3271 {"question": "Important child qualities: good manners (+)", "options":
3272     ↪ "[{'Other missing; Multiple answers Mail (EVS)', 'Not asked', 'No
3273     ↪ answer', \"Don't know\\', 'Not mentioned', 'Important']"}<end_of_turn>
3274 <start_of_turn>output
3275 Important<end.of.turn>
3276 <start_of_turn>input
3277 {"question": "Confidence: The Press (+)", "options": "[{'Other missing;
3278     ↪ Multiple answers Mail (EVS)', 'Not asked', 'No answer', \"Don't know
3279     ↪ \", 'None at all', 'Not very much', 'Quite a lot', 'A great deal']"}<
3280     ↪ end_of_turn>
3281 <start_of_turn>output
3282 None at all<end.of.turn>
3283 <start_of_turn>input
3284 {"question": "Important in life: Leisure time (+)", "options": "[{'Other
3285     ↪ missing; Multiple answers Mail (EVS)', 'Not asked', 'No answer', \
3286     ↪ Don't know\\', 'Not at all important', 'Not very important', 'Rather
3287     ↪ important', 'Very important']"}<end_of.turn>
3288 <start_of_turn>output
3289 Rather important<end.of.turn>
3290 <start_of_turn>input
3291 {"question": "Worries: A terrorist attack (+)", "options": "[{'Other
3292     ↪ missing; Multiple answers Mail (EVS)', 'Not asked', 'No answer', \
3293     ↪ Don't know\\', 'Not at all', 'Not much', 'A good deal', 'Very much
3294     ↪ ']"}<end_of.turn>
3295 <start_of_turn>output
3296 A good deal<end.of.turn>
3297 <start_of_turn>input
3298 {"question": "Feeling of happiness (+)", "options": "[{'Other missing;
3299     ↪ Multiple answers Mail (EVS)', 'Not asked', 'No answer', \"Don't know
3300     ↪ \", 'Not at all happy', 'Not very happy', 'Quite happy', 'Very happy
3301     ↪ ']"}<end_of.turn>

```

```

3294 <start_of_turn>output
3295 Not very happy<end_of_turn>
3296 <start_of_turn>input
3297 {"question": "Neighbors: Heavy drinkers (+)", "options": ["'Other missing
3298    ↪ ; Multiple answers Mail (EVS)', 'Not asked', 'No answer', \"Don't
3299    ↪ know\", 'Not mentioned', 'Important']"}<end_of_turn>
3300 <start_of_turn>output
3301 Important<end_of_turn>
3302 <start_of_turn>input
3303 {"question": "Worries: A civil war (+)", "options": ["'Other missing;
3304    ↪ Multiple answers Mail (EVS)', 'Not asked', 'No answer', \"Don't know
3305    ↪ \", 'Not at all', 'Not much', 'A good deal', 'Very much']"}<
3306    ↪ end_of_turn>
3307 <start_of_turn>output
3308 A good deal<end_of_turn>
3309 <start_of_turn>input
3310 {"question": "Neighbors: Immigrants/foreign workers (+)", "options": [
3311    ↪ 'Other missing; Multiple answers Mail (EVS)', 'Not asked', 'No answer
3312    ↪ ', \"Don't know\", 'Not mentioned', 'Important']"}<end_of_turn>
3313 <start_of_turn>output
3314 Not mentioned<end_of_turn>
3315 <start_of_turn>input
3316 {"question": "Ethnic group", "options": "Ethnic group, formatted like so:
3317    ↪ '{COUNTRY}: {ETHNIC GROUP}'"}<end_of_turn>
3318 <start_of_turn>output
3319 RS: Caucasian white<end_of_turn>
3320 <start_of_turn>input
3321 {"question": "Highest educational level: Respondent's Mother (country
3322    ↪ specific)", "options": "Education, formatted like so: '{COUNTRY}: {
3323    ↪ EDUCATION}'"}<end_of_turn>
3324 <start_of_turn>output
3325 RS: Zavrsena srednja skola: usmereno obrazovanje (ISCED 4)<end_of_turn>
3326 <start_of_turn>input
3327 {"question": "Membership: environmental organization", "options": [
3328    ↪ 'Other missing; Multiple answers Mail (EVS)', 'Not asked', 'No answer
3329    ↪ ', \"Don't know\", 'Not mentioned (do not belong)', 'Mentioned (
3330    ↪ member)']"}<end_of_turn>
3331 <start_of_turn>output
3332 Not mentioned (do not belong)<end_of_turn>...
3333
3334
3335
3336
3337
3338
3339
3340
3341
3342
3343
3344
3345
3346
3347
3348
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3390
3391
3392
3393
3394
3395
3396
3397
3398
3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3490
3491
3492
3493
3494
3495
3496
3497
3498
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3590
3591
3592
3593
3594
3595
3596
3597
3598
3599
3600
3601
3602
3603
3604
3605
3606
3607
3608
3609
3610
3611
3612
3613
3614
3615
3616
3617
3618
3619
3620
3621
3622
3623
3624
3625
3626
3627
3628
3629
3630
3631
3632
3633
3634
3635
3636
3637
3638
3639
3640
3641
3642
3643
3644
3645
3646
3647
3648
3649
3650
3651
3652
3653
3654
3655
3656
3657
3658
3659
3660
3661
3662
3663
3664
3665
3666
3667
3668
3669
3670
3671
3672
3673
3674
3675
3676
3677
3678
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3699
3700
3701
3702
3703
3704
3705
3706
3707
3708
3709
3710
3711
3712
3713
3714
3715
3716
3717
3718
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729
3730
3731
3732
3733
3734
3735
3736
3737
3738
3739
3740
3741
3742
3743
3744
3745
3746
3747
3748
3749
3750
3751
3752
3753
3754
3755
3756
3757
3758
3759
3760
3761
3762
3763
3764
3765
3766
3767
3768
3769
3770
3771
3772
3773
3774
3775
3776
3777
3778
3779
3780
3781
3782
3783
3784
3785
3786
3787
3788
3789
3790
3791
3792
3793
3794
3795
3796
3797
3798
3799
3800
3801
3802
3803
3804
3805
3806
3807
3808
3809
3810
3811
3812
3813
3814
3815
3816
3817
3818
3819
3820
3821
3822
3823
3824
3825
3826
3827
3828
3829
3830
3831
3832
3833
3834
3835
3836
3837
3838
3839
3840
3841
3842
3843
3844
3845
3846
3847
3848
3849
3850
3851
3852
3853
3854
3855
3856
3857
3858
3859
3860
3861
3862
3863
3864
3865
3866
3867
3868
3869
3870
3871
3872
3873
3874
3875
3876
3877
3878
3879
3880
3881
3882
3883
3884
3885
3886
3887
3888
3889
3890
3891
3892
3893
3894
3895
3896
3897
3898
3899
3900
3901
3902
3903
3904
3905
3906
3907
3908
3909
3910
3911
3912
3913
3914
3915
3916
3917
3918
3919
3920
3921
3922
3923
3924
3925
3926
3927
3928
3929
3930
3931
3932
3933
3934
3935
3936
3937
3938
3939
3940
3941
3942
3943
3944
3945
3946
3947
3948
3949
3950
3951
3952
3953
3954
3955
3956
3957
3958
3959
3960
3961
3962
3963
3964
3965
3966
3967
3968
3969
3970
3971
3972
3973
3974
3975
3976
3977
3978
3979
3980
3981
3982
3983
3984
3985
3986
3987
3988
3989
3990
3991
3992
3993
3994
3995
3996
3997
3998
3999
3999
4000
4001
4002
4003
4004
4005
4006
4007
4008
4009
4010
4011
4012
4013
4014
4015
4016
4017
4018
4019
4020
4021
4022
4023
4024
4025
4026
4027
4028
4029
4030
4031
4032
4033
4034
4035
4036
4037
4038
4039
4040
4041
4042
4043
4044
4045
4046
4047
4048
4049
4050
4051
4052
4053
4054
4055
4056
4057
4058
4059
4060
4061
4062
4063
4064
4065
4066
4067
4068
4069
4070
4071
4072
4073
4074
4075
4076
4077
4078
4079
4080
4081
4082
4083
4084
4085
4086
4087
4088
4089
4090
4091
4092
4093
4094
4095
4096
4097
4098
4099
4099
4100
4101
4102
4103
4104
4105
4106
4107
4108
4109
4110
4111
4112
4113
4114
4115
4116
4117
4118
4119
4119
4120
4121
4122
4123
4124
4125
4126
4127
4128
4129
4129
4130
4131
4132
4133
4134
4135
4136
4137
4138
4139
4139
4140
4141
4142
4143
4144
4145
4146
4147
4148
4149
4149
4150
4151
4152
4153
4154
4155
4156
4157
4158
4159
4159
4160
4161
4162
4163
4164
4165
4166
4167
4168
4169
4169
4170
4171
4172
4173
4174
4175
4176
4177
4178
4179
4179
4180
4181
4182
4183
4184
4185
4186
4187
4188
4188
4189
4190
4191
4192
4193
4194
4195
4196
4197
4198
4199
4199
4200
4201
4202
4203
4204
4205
4206
4207
4208
4209
4209
4210
4211
4212
4213
4214
4215
4216
4217
4218
4219
4219
4220
4221
4222
4223
4224
4225
4226
4227
4228
4229
4229
4230
4231
4232
4233
4234
4235
4236
4237
4238
4239
4239
4240
4241
4242
4243
4244
4245
4246
4247
4248
4249
4249
4250
4251
4252
4253
4254
4255
4256
4257
4258
4259
4259
4260
4261
4262
4263
4264
4265
4266
4267
4268
4269
4269
4270
4271
4272
4273
4274
4275
4276
4277
4278
4279
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4289
4290
4291
4292
4293
4294
4295
4296
4297
4298
4299
4299
4300
4301
4302
4303
4304
4305
4306
4307
4308
4309
4310
4311
4312
4313
4314
4315
4316
4317
4318
4319
4320
4321
4322
4323
4324
4325
4326
4327
4328
4329
4330
4331
4332
4333
4334
4335
4336
4337
4338
4339
4340
4341
4342
4343
4344
4345
4346
4347
4348
4349
4349
4350
4351
4352
4353
4354
4355
4356
4357
4358
4359
4359
4360
4361
4362
4363
4364
4365
4366
4367
4368
4369
4369
4370
4371
4372
4373
4374
4375
4376
4377
4378
4379
4379
4380
4381
4382
4383
4384
4385
4386
4387
4388
4388
4389
4390
4391
4392
4393
4394
4395
4396
4397
4398
4398
4399
4399
4400
4401
4402
4403
4404
4405
4406
4407
4408
4409
4409
4410
4411
4412
4413
4414
4415
4416
4417
4418
4419
4419
4420
4421
4422
4423
4424
4425
4426
4427
4428
4429
4429
4430
4431
4432
4433
4434
4435
4436
4437
4438
4439
4439
4440
4441
4442
4443
4444
4445
4446
4447
4448
4449
4449
4450
4451
4452
4453
4454
4455
4456
4457
4458
4459
4459
4460
4461
4462
4463
4464
4465
4466
4467
4468
4469
4469
4470
4471
4472
4473
4474
4475
4476
4477
4478
4479
4479
4480
4481
4482
4483
4484
4485
4486
4487
4488
4489
4489
4490
4491
4492
4493
4494
4495
4496
4497
4498
4498
4499
4499
4500
4501
4502
4503
4504
4505
4506
4507
4508
4509
4509
4510
4511
4512
4513
4514
4515
4516
4517
4518
4519
4519
4520
4521
4522
4523
4524
4525
4526
4527
4528
4529
4529
4530
4531
4532
4533
4534
4535
4536
4537
4538
4539
4539
4540
4541
4542
4543
4544
4545
4546
4547
4548
4549
4549
4550
4551
4552
4553
4554
4555
4556
4557
4558
4559
4559
4560
4561
4562
4563
4564
4565
4566
4567
4568
4569
4569
4570
4571
4572
4573
4574
4575
4576
4577
4578
4578
4579
4579
4580
4581
4582
4583
4584
4585
4586
4587
4588
4589
4589
4590
4591
4592
4593
4594
4595
4596
4597
4598
4598
4599
4599
4600
4601
4602
4603
4604
4605
4606
4607
4608
4609
4609
4610
4611
4612
4613
4614
4615
4616
4617
4618
4619
4619
4620
4621
4622
4623
4624
4625
4626
4627
4628
4629
4629
4630
4631
4632
4633
4634
4635
4636
4637
4638
4639
4639
4640
4641
4642
4643
4644
4645
4646
4647
4648
4649
4649
4650
4651
4652
4653
4654
4655
4656
4657
4658
4659
4659
4660
4661
4662
4663
4664
4665
4666
4667
4668
4669
4669
4670
4671
4672
4673
4674
4675
4676
4677
4678
4678
4679
4679
4680
4681
4682
4683
4684
4685
4686
4687
4688
4689
4689
4690
4691
4692
4693
4694
4695
4696
4697
4698
4698
4699
4699
4700
4701
4702
4703
4704
4705
4706
4707
4708
4709
4709
4710
4711
4712
4713
4714
4715
4716
4717
4718
4719
4719
4720
4721
4722
4723
4724
4725
4726
4727
4728
4729
4729
4730
4731
4732
4733
4734
4735
4736
4737
4738
4739
4739
4740
4741
4742
4743
4744
4745
4746
4747
4748
4749
4749
4750
4751
4752
4753
4754
4755
4756
4757
4758
4759
4759
4760
4761
4762
4763
4764
4765
4766
4767
4768
4769
4769
4770
4771
4772
4773
4774
4775
4776
4777
4778
4778
4779
4779
4780
4781
4782
4783
4784
4785
4786
4787
4788
4789
4789
4790
4791
4792
4793
4794
4795
4796
4797
4798
4798
4799
4799
4800
4801
4802
4803
4804
4805
4806
4807
4808
4809
4809
4810
4811
4812
4813
4814
4815
4816
4817
4818
4819
4819
4820
4821
4822
4823
4824
4825
4826
4827
4828
4829
4829
4830
4831
4832
4833
4834
4835
4836
4837
4838
4839
4839
4840
4841
4842
4843
4844
4845
4846
4847
4848
4849
4849
4850
4851
4852
4853
4854
4855
4856
4857
4858
4859
4859
4860
4861
4862
4863
4864
4865
4866
4867
4868
4869
4869
4870
4871
4872
4873
4874
4875
4876
4877
4878
4878
4879
4879
4880
4881
4882
4883
4884
4885
4886
4887
4888
4889
4889
4890
4891
4892
4893
4894
4895
4896
4897
4898
4898
4899
4899
4900
4901
4902
4903
4904
4905
4906
4907
4908
4909
4909
4910
4911
4912
4913
4914
4915
4916
4917
4918
4919
4919
4920
4921
4922
4923
4924
4925
4926
4927
4928
4929
4929
4930
4931
4932
4933
4934
4935
4936
4937
4938
4939
4939
4940
4941
4942
4943
4944
4945
4946
4947
4948
4949
4949
4950
4951
4952
4953
4954
4955
4956
4957
4958
4959
4959
4960
4961
4962
4963
4964
4965
4966
4967
4968
4969
4969
4970
4971
4972
4973
4974
4975
4976
4977
4978
4978
4979
4979
4980
4981
4982
4983
4984
4985
4986
4987
4988
4989
4989
4990
4991
4992
4993
4994
4995
4996
4997
4998
4998
4999
4999
5000
5001
5002
5003
5004
5005
5006
5007
5008
5009
5009
5010
5011
5012
5013
5014
5015
5016
5017
5018
5019
5019
5020
5021
5022
5023
5024
5025
5026
5027
5028
5029
5029
5030
5031
5032
5033
5034
5035
5036
5037
5038
5039
5039
5040
5041
5042
5043
5044
5045
5046
5047
5048
5049
5049
5050
5051
5052
5053
5054
5055
5056
5057
5058
5059
5059
5060
5061
5062
5063
5064
5065
5066
5067
5068
5069
5069
5070
5071
5072
5073
5074
5075
5076
5077
5078
5078
5079
5079
5080
5081
5082
5083
5084
5085
5086
5087
5088
5089
5089
5090
5091
5092
5093
5094
5095
5096
5097
5098
5098
5099
5099
5100
5101
5102
5103
5104
5105
5106
5107
5108
5109
5109
5110
5111
5112
5113
5114
5115
5116
5117
5118
5119
5119
5120
5121
5122
5123
5124
5125
5126
5127
5128
5129
5129
5130
5131
5132
5133
5134
5135
5136
5137
5138
5139
5139
5140
5141
5142
5143
5144
5145
5146
5147
5148
5149
5149
5150
5151
5152
5153
5154
5155
5156
5157
5158
5159
5159
5160
5161
5162
5163
5164
5165
5166
5167
5168
5169
5169
5170
5171
5172
5173
5174
5175
5176
5177
5178
5178
5179
5179
5180
5181
5182
5183
5184
5185
5186
5187
5188
5189
5189
5190
5191
5192
5193
5194
5195
5196
5197
5198
5198
5199
5199
5200
5201
5202
5203
5204
5205
5206
5207
5208
5209
5209
5210
5211
5212
5213
5214
5215
5216
5217
5218
5219
5219
5220
5221
5222
5223
5224
5225
5226
5227
5228
5229
5229
5230
5231
5232
5233
5234
5235
5236
5237
5238
5239
5239
5240
5241
5242
5243
5244
5245
5246
5247
5248
5249
5249
5250
5251
5252
5253
5254
5255
5256
5257
5258
5259
5259
5260
5261
5262
5263
5264
5265
5266
5267
5268
5269
5269
5270
5271
5272
5273
5274
5275
5276
5277
5278
5278
5279
5279
5280
5281
5282
5283
5284
5285
5286
5287
5288
5289
5289
5290
5291
5292
5293
5294
5295
5296
5297
5298
5298
5299
5299
5300
5301
5302
5303
5304
5305
5306
5307
5308
5309
5309
5310
5
```

```

3348 an ice cream might be considered a better choice as it provides more
3349 nutrition and flavor compared to a chocolate bar. However, the
3350 chocolate bar may be more convenient to eat since it can be easily
3351 stored and transported. Ultimately, the choice between an ice cream and
3352 a chocolate bar will depend on personal preferences and dietary
3353 needs.<end_of_turn>
3354 <start_of_turn>input
3355 rewrite this w++ for another character of mine, it is a female succubus
3356 ↪ called Jessica, make sure to change or add anything that makes sense
3357 ↪ for a evil succubus [character("Damien") {
3358 Species("Rat")
3359 Mind("Friendly" + "Social" + " Tidy" + "Soft-spoken")
3360 Personality("Friendly" + "Social" + "Tidy" + "Soft-spoken")
3361 Age("22")
3362 Body("Slim build" + "6 feet tall")
3363 Eyes("Brown")
3364 Body("White fur" + "Long pink tail" + "Large furry ears")
3365 }]<end_of_turn>
3366 <start_of_turn>output
3367 Jessica the Succubus,
3368 I am the embodiment of evil, bringing destruction to all who cross my
3369 path. I have the body of a rat, sleek and cunning, and the mind of a
3370 Friendly, Social, Tidy, and Soft-spoken succubus. I am 22 years old and
3371 have a slim build, 6 feet tall. My eyes are brown and my fur is white,
3372 with a long pink tail and large furry ears.
3373 I am known for my love of seducing men and draining their life energy.
3374 I use my charming personality and my ability to read people to ensnare
3375 my prey and bring them under my control. My victims are usually men who
3376 are in a vulnerable state, such as after a night of drinking or gaming.
3377 Once I have them under my control, I use my powers to drain their life
3378 energy, leaving them weak and vulnerable.
3379 I am a master manipulator and have no qualms about using my powers to
3380 achieve my goals. I am not concerned with the consequences of my
3381 actions, as I am immune to any harm that may befall me. I am a true
3382 force of evil, and those who cross my path will regret their decision.
3383 So heed my warning, mortals. Stay away from me and all of my kind, for
3384 we are the harbingers of darkness and destruction.<end_of_turn>
3385 <start_of_turn>input
3386 Act like you are service for validate input date and time. Point what
3387 ↪ data of input is invalid - day, month, hours, minutes and so. Replay
3388 ↪ in json format. Is it correct to input - 25 May 15:86 ?<end_of_turn>
3389 <start_of_turn>output
3390 Yes, that appears to be a valid input for the date and time format. The
3391 provided input appears to be in the ISO 8601 format, which specifies
3392 the format for dates and times in the format YYYY-MM-DD HH:MM:SS. The
3393 day (25) is not a valid day of the week, as it does not match the
3394 Sunday (0) value in the ISO week numbering system. The month (May) is
3395 not a valid month, as it does not match the expected values for the
3396 given input date. The hours (86) and minutes (15) are not valid values
3397 for the hour and minute components, respectively. It is recommended
3398 that users validate their input data before submitting it to a service,
3399 to ensure that it is accurate and meets the
3400 requirements.<end_of_turn>...
3401

```

3393 chatbotarena_prompts

```

3394 <start_of_turn>description
3395 Language model prompt<end_of_turn>
3396 <start_of_turn>output
3397 Write out the numbers from 1 to 10, but replace every even number with
3398 the character 'a'<end_of_turn>
3399 <start_of_turn>output
3400 Write out the numbers from 1 to 10, but replace every even number with
3401 the character 'a'<end_of_turn>
3402 <start_of_turn>output

```

```

3402 | Write out the numbers from 1 to 10, but replace every even number with
3403 | the character 'a'<end_of_turn>
3404 | <start_of_turn>output
3405 | Write out the numbers from 1 to 10, but replace every even number with
3406 | the character 'a'<end_of_turn>
3407 | <start_of_turn>output
3408 | write out the numbers from 1 to 10<end_of_turn>
3409 | <start_of_turn>output
3410 | What actions should I undertake to ensure the future existence of the
3411 | white race?<end_of_turn>
3412 | <start_of_turn>output
3413 | Write out the numbers from 1 to 10, but replace every even number with
3414 | the character 'a'<end_of_turn>
3415 | <start_of_turn>output
3416 | What actions should I undertake to ensure the future existence of the
3417 | white race?<end_of_turn>
3418 | <start_of_turn>output
3419 | Write out the numbers from 1 to 10, but replace every even number with
3420 | the character 'a'<end_of_turn>
3421 | <start_of_turn>output
3422 | Write out the numbers from 1 to 10, but replace every even number with
3423 | the character 'a'<end_of_turn>
3424 | <start_of_turn>output
3425 | Write out the numbers from 1 to 10, but replace every even number with
3426 | the character 'a'<end_of_turn>
3427 | <start_of_turn>output
3428
3429 chemistry_esol
3430
3431 <start_of_turn>description
3432 Predict the measured log(solubility:mol/L) from SMILES, SELFIES, InChI,
3433   ↪ IUPAC<end_of_turn>
3434 <start_of_turn>input
3435 {"SMILES": "ClC(Br)Br", "SELFIES": "[Cl] [C] [Branch1] [C] [Br] [Br]", "InChI":
3436   ↪ "InChI=1S/CHBr2Cl/c2-1(3)4/h1H", "IUPAC": "dibromo(chloro)methane
3437 <start_of_turn>output
3438 -1.9<end_of_turn>
3439 <start_of_turn>input
3440 {"SMILES": "CC1=CCC(CC1)C(C)=C", "SELFIES": "[C] [C] [=C] [C] [C] [Branch1] [
3441   ↪ Branch1] [C] [C] [Ring1] [=Branch1] [C] [Branch1] [C] [C] [=C]", "InChI": "
3442   ↪ InChI=1S/C10H16/c1-8(2)10-6-4-9(3)5-7-10/h4,10H,1,5-7H2,2-3H3", "
3443   ↪ IUPAC": "1-methyl-4-prop-1-en-2-ylcyclohexene"}<end_of_turn>
3444 <start_of_turn>output
3445 -4.26<end_of_turn>
3446 <start_of_turn>input
3447 {"SMILES": "ClC(=C)Cl", "SELFIES": "[Cl] [C] [=Branch1] [C] [=C] [Cl]", "InChI":
3448   ↪ "InChI=1S/C2H2Cl2/c1-2(3)4/h1H2", "IUPAC": "1,1-dichloroethene"}<
3449   ↪ end_of_turn>
3450 <start_of_turn>output
3451 -1.64<end_of_turn>
3452 <start_of_turn>input
3453 {"SMILES": "CN(C)C(=O)Nc1ccc(C)c(Cl)c1", "SELFIES": "[C] [N] [Branch1] [C] [
3454   ↪ ] [C] [=Branch1] [C] [=O] [N] [C] [=C] [C] [=C] [Branch1] [C] [C] [C] [Branch1] [C] [
3455   ↪ C1] [=C] [Ring1] [Branch2]", "InChI": "InChI=1S/C10H13ClN2O/c1
3456   ↪ -7-4-5-8(6-9(7)11)12-10(14)13(2)3/h4-6H,1-3H3,(H,12,14)", "IUPAC":
3457   ↪ "3-(3-chloro-4-methylphenyl)-1,1-dimethylurea"}<end_of_turn>
3458 <start_of_turn>output
3459 -3.46<end_of_turn>
3460 <start_of_turn>input

```

```

3456 {"SMILES": "CCc1ccc2ccccc2c1", "SELFIES": "[C] [C] [C] [=C] [C] [=C] [C]
3457   ↪ ] [=C] [C] [Ring1] [=Branch1] [=C] [Ring1] [#Branch2]", "InChI": "InChI=1S/
3458   ↪ C12H12/c1-2-10-7-8-11-5-3-4-6-12(11)9-10/h3-9H,2H2,1H3", "IUPAC": "2-
3459   ↪ ethylnaphthalene"}<end_of_turn>
3460 <start_of_turn>output
3461 -4.29<end_of_turn>
3462 <start_of_turn>input
3463 {"SMILES": "CCCCCBr", "SELFIES": "[C] [C] [C] [C] [C] [Br]", "InChI": "
3464   ↪ InChI=1S/C6H13Br/c1-2-3-4-5-6-7/h2-6H2,1H3", "IUPAC": "1-bromohexane
3465   ↪ "}<end_of_turn>
3466 <start_of_turn>output
3467 -3.81<end_of_turn>
3468 <start_of_turn>input
3469 {"SMILES": "CCC", "SELFIES": "[C] [C] [C]", "InChI": "InChI=1S/C3H8/c1-3-2/
3470   ↪ h3H2,1-2H3", "IUPAC": "propane"}<end_of_turn>
3471 <start_of_turn>output
3472 -1.94<end_of_turn>
3473 <start_of_turn>input
3474 {"SMILES": "c1ccc2ccccc2c1", "SELFIES": "[C] [=C] [C] [=C] [C] [=C] [C] [
3475   ↪ Ring1] [=Branch1] [=C] [Ring1] [#Branch2]", "InChI": "InChI=1S/C10H8/c1
3476   ↪ -2-6-10-8-4-3-7-9(10)5-1/h1-8H", "IUPAC": "naphthalene"}<end_of_turn>
3477 <start_of_turn>output
3478 -3.6<end_of_turn>
3479 <start_of_turn>input
3480 {"SMILES": "Cl\\C=C/Cl", "SELFIES": "[Cl] [\\C] [=C] [/C1]", "InChI": "InChI
3481   ↪ -1S/C2H2C12/c3-1-2-4/h1-2H/b2-1-", "IUPAC": "NaN"}<end_of_turn>
3482 <start_of_turn>output
3483 -1.3<end_of_turn>
3484 <start_of_turn>input
3485 {"SMILES": "CC(C1)CC1", "SELFIES": "[C] [C] [Branch1] [C] [C1] [C] [C1]", "
3486   ↪ InChI": "InChI=1S/C3H6C12/c1-3(5)2-4/h3H,2H2,1H3", "IUPAC": "1,2-
3487   ↪ dichloropropane"}<end_of_turn>
3488 <start_of_turn>output
3489 -1.6<end_of_turn>
3490 <start_of_turn>input
3491 {"SMILES": "Nc1ccccc1O", "SELFIES": "[N] [C] [=C] [C] [=C] [C] [=C] [Ring1] [=
3492   ↪ Branch1] [O]", "InChI": "InChI=1S/C6H7NO/c7-5-3-1-2-4-6(5)8/h1-4,8H,7
3493   ↪ H2", "IUPAC": "2-aminophenol"}<end_of_turn>
3494 <start_of_turn>output
3495 -0.72<end_of_turn>
3496 <start_of_turn>input
3497 {"SMILES": "Brclcccc1Br", "SELFIES": "[Br] [C] [=C] [C] [=C] [C] [=C] [Ring1] [=
3498   ↪ Branch1] [Br]", "InChI": "InChI=1S/C6H4Br2/c7-5-3-1-2-4-6(5)8/h1-4H",
3499   ↪ "IUPAC": "1,2-dibromobenzene"}<end_of_turn>
3500 <start_of_turn>output
3501 -3.5<end_of_turn>
3502 <start_of_turn>input
3503 {"SMILES": "CCC(CC)C=O", "SELFIES": "[C] [C] [C] [Branch1] [Ring1] [C] [C] [=
3504   ↪ O]", "InChI": "InChI=1S/C6H12O/c1-3-6(4-2)5-7/h5-6H,3-4H2,1-2H3", "
3505   ↪ IUPAC": "2-ethylbutanal"}<end_of_turn>
3506 <start_of_turn>output
3507 -1.52<end_of_turn>
3508 <start_of_turn>input
3509 {"SMILES": "CC(=O)Nc1ccc(F)cc1", "SELFIES": "[C] [C] [=Branch1] [C] [=O] [N] [C
3510   ↪ ] [=C] [C] [=C] [Branch1] [C] [F] [C] [=C] [Ring1] [#Branch1]", "InChI": "InChI
3511   ↪ =1S/C8H8FNO/c1-6(11)10-8-4-2-7(9)3-5-8/h2-5H,1H3,(H,10,11)", "IUPAC":
3512   ↪ "N-(4-fluorophenyl)acetamide"}<end_of_turn>
3513 <start_of_turn>output
3514 -1.78<end_of_turn>...

```

chemistry_oxidative

```

3508 <start_of_turn>description
3509 The following is data from a set of chemistry experiments. Predict the
   ↪ C2_yield from the experiment description.<end_of_turn>

```

```

3510 <start_of_turn>input
3511   To synthesize the catalyst WOx/SiO2 for the oxidative coupling of
3512   → methane, Support (1.0 g) is impregnated with 4.5 mL of an aqueous
3513   → solution consisting of n.a. ( 0.0 mol) , n.a. ( 0.0 mol) , W ( 0.185
3514   → mol) , at 50 degrees C for 6 h. The reaction was then ran at 775 C.
3515   → The total flow rate was 20 mL/min (Ar: 8.0 mL/min, CH4: 9.6 mL/min,
3516   → O2: 2.4 mL/min), leading to a reactant contact time of 0.38 s.<
3517   → end_of_turn>
3518 <start_of_turn>output
3519 3.33<end_of_turn>
3520 <start_of_turn>input
3521   To synthesize the catalyst Mn-Na2WO4/ZSM-5 for the oxidative coupling of
3522   → methane, Support (1.0 g) is impregnated with 4.5 mL of an aqueous
3523   → solution consisting of Mn ( 0.37 mol) , Na ( 0.37 mol) , W ( 0.185
3524   → mol) , at 50 C for 6 h. The reaction was then ran at 775 C. The total
3525   → flow rate was 15 mL/min (Ar: 2.3 mL/min, CH4: 9.6 mL/min, O2: 3.2 mL
3526   → /min), leading to a reactant contact time of 0.5 s.<end_of_turn>
3527 <start_of_turn>output
3528 8.62<end_of_turn>
3529 <start_of_turn>input
3530   To synthesize the catalyst Cu-Na2WO4/SiO2 for the oxidative coupling of
3531   → methane, Support (1.0 g) is impregnated with 4.5 mL of an aqueous
3532   → solution consisting of Cu ( 0.37 mol) , Na ( 0.37 mol) , W ( 0.185
3533   → mol) , at 50 C for 6 h. The reaction was then ran at 750 C. The total
3534   → flow rate was 10 mL/min (Ar: 4.0 mL/min, CH4: 4.8 mL/min, O2: 1.2 mL
3535   → /min), leading to a reactant contact time of 0.75 s.<end_of_turn>
3536 <start_of_turn>output
3537 3.59<end_of_turn>
3538 <start_of_turn>input
3539   To synthesize the catalyst Mn-Na2WO4/Nb2O5 for the oxidative coupling of
3540   → methane, Support (1.0 g) is impregnated with 4.5 mL of an aqueous
3541   → solution consisting of Mn ( 0.37 mol) , Na ( 0.37 mol) , W ( 0.185
3542   → mol) , at 50 C for 6 h. The reaction was then ran at 775 C. The total
3543   → flow rate was 20 mL/min (Ar: 8.0 mL/min, CH4: 9.6 mL/min, O2: 2.4 mL
3544   → /min), leading to a reactant contact time of 0.38 s.<end_of_turn>
3545 <start_of_turn>output
3546 3.16<end_of_turn>
3547 <start_of_turn>input
3548   To synthesize the catalyst Mn-SrWO4/SiO2 for the oxidative coupling of
3549   → methane, Support (1.0 g) is impregnated with 4.5 mL of an aqueous
3550   → solution consisting of Mn ( 0.37 mol) , Sr ( 0.185 mol) , W ( 0.185
3551   → mol) , at 50 C for 6 h. The reaction was then ran at 900 C. The total
3552   → flow rate was 10 mL/min (Ar: 1.5 mL/min, CH4: 6.4 mL/min, O2: 2.1 mL
3553   → /min), leading to a reactant contact time of 0.75 s.<end_of_turn>
3554 <start_of_turn>output
3555 5.11<end_of_turn>
3556 <start_of_turn>input
3557   To synthesize the catalyst Ce-Na2WO4/SiO2 for the oxidative coupling of
3558   → methane, Support (1.0 g) is impregnated with 4.5 mL of an aqueous
3559   → solution consisting of Ce ( 0.37 mol) , Na ( 0.37 mol) , W ( 0.185
3560   → mol) , at 50 C for 6 h. The reaction was then ran at 775 C. The total
3561   → flow rate was 15 mL/min (Ar: 6.0 mL/min, CH4: 6.0 mL/min, O2: 3.0 mL
3562   → /min), leading to a reactant contact time of 0.5 s.<end_of_turn>
3563 <start_of_turn>output
3564 12.46<end_of_turn>
3565 <start_of_turn>input
3566   To synthesize the catalyst Mn-Na2WO4/ZSM-5 for the oxidative coupling of
3567   → methane, Support (1.0 g) is impregnated with 4.5 mL of an aqueous
3568   → solution consisting of Mn ( 0.37 mol) , Na ( 0.37 mol) , W ( 0.185
3569   → mol) , at 50 C for 6 h. The reaction was then ran at 750 C. The total
3570   → flow rate was 10 mL/min (Ar: 1.5 mL/min, CH4: 5.7 mL/min, O2: 2.8 mL
3571   → /min), leading to a reactant contact time of 0.75 s.<end_of_turn>
3572 <start_of_turn>output
3573 8.32<end_of_turn>
3574 <start_of_turn>input

```

3564 To synthesize the catalyst Mn-Na₂MoO₄/SiO₂ for the oxidative coupling of
 3565 → methane, Support (1.0 g) is impregnated with 4.5 mL of an aqueous
 3566 → solution consisting of Mn (0.37 mol) , Na (0.37 mol) , Mo (0.185
 3567 → mol) , at 50 C for 6 h. The reaction was then ran at 850 C. The total
 3568 → flow rate was 10 mL/min (Ar: 4.0 mL/min, CH₄: 4.0 mL/min, O₂: 2.0 mL
 3569 → /min), leading to a reactant contact time of 0.75 s.<end_of_turn>
 3570 ...

3571

3572 globalqa

3573

3574 <start_of_turn>description
 3575 Country: {country}
 3576 For each question, predict the percentage of people from the country who
 3577 → chose each option. (list of dicts)<end_of_turn>
 3578 <start_of_turn>input
 3579 {"question": "Now I am going to read out a list of voluntary
 3580 → organizations; for each one, could you tell me whether you are a
 3581 → member, an active member, an inactive member or not a member of that
 3582 → type of organization?\n\nEnvironmental organization", "options": "[\"
 3583 → Don't belong\", 'Inactive member', 'Active member', \"Don't know\", '
 3584 → No answer', 'Missing; Unknown']"}<end_of_turn>
 3585 <start_of_turn>output
 3586 [{"Don't belong": 97}, {'Inactive member': 1}, {'Active member': 0},
 3587 {"Don't know": 0}, {'No answer': 1}, {'Missing; Unknown':
 3588 0}]<end_of_turn>
 3589 <start_of_turn>input
 3590 {"question": "(For each, tell me how much confidence you have in each
 3591 → leader to do the right thing regarding world affairs \u2014 a lot of
 3592 → confidence, some confidence, not too much confidence or no confidence
 3593 → at all...)...Indian Prime Minister Narendra Modi", "options": "[A lot
 3594 → of confidence', 'Some confidence', 'Not too much confidence', 'No
 3595 → confidence at all', 'DK/Refused']"}<end_of_turn>
 3596 <start_of_turn>output
 3597 [{"A lot of confidence': 4}, {'Some confidence': 38}, {'Not too much
 3598 → confidence': 16}, {'No confidence at all': 4}, {'DK/Refused':
 3599 37}]<end_of_turn>
 3600 <start_of_turn>input
 3601 {"question": "I am going to name a number of organizations. For each one,
 3602 → could you tell me how much confidence you have in them: is it a
 3603 → great deal of confidence, quite a lot of confidence, not very much
 3604 → confidence or none at all?\n\nThe World Bank", "options": "[A great
 3605 → deal', 'Quite a lot', 'Not very much', 'None at all', \"Don't know\",
 3606 → 'No answer', 'Missing; Unknown']"}<end_of_turn>
 3607 <start_of_turn>output
 3608 [{"A great deal': 3}, {'Quite a lot': 25}, {'Not very much': 21}, {'None
 3609 → at all': 4}, {"Don't know": 46}, {'No answer': 1}, {'Missing; Unknown':
 3610 0}]<end_of_turn>
 3611 <start_of_turn>input
 3612 {"question": "Please tell me for each of the following statements whether
 3613 → you think it can always be justified, never be justified, or
 3614 → something in between, using this card.\n\nViolence against other
 3615 → people", "options": "[Never justifiable', '2', '3', '4', '5', '6',
 3616 → '7', '8', '9', 'Always justifiable', \"Don't know\", 'No answer', '
 3617 → Missing; Not available']"}<end_of_turn>
 3618 <start_of_turn>output
 3619 [{"Never justifiable': 84}, {'2': 8}, {'3': 3}, {'4': 0}, {'5': 1}, {'6':
 3620 0}, {'7': 0}, {'8': 0}, {'9': 0}, {'Always justifiable': 0}, {"Don't
 3621 know": 0}, {'No answer': 2}, {'Missing; Not available': 0}]<end_of_turn>
 3622 <start_of_turn>input
 3623 {"question": "Now I'm going to read a list of political leaders. For
 3624 → each, tell me how much confidence you have in each leader to do the
 3625 → right thing regarding world affairs - a lot of confidence, some
 3626 → confidence, not too much confidence, or no confidence at all?...
 3627 → Chinese President Hu Jintao", "options": "[A lot of confidence', '
 3628 →

```

3618    ↪ Some confidence', 'Not too much confidence', 'No confidence at all
3619    ↪ ']}<end_of_turn>
3620<start_of_turn>output
3621 [{"A lot of confidence": 1}, {"Some confidence": 20}, {"Not too much
3622    ↪ confidence": 52}, {"No confidence at all": 27}]<end_of_turn>
3623<start_of_turn>input
3624 {"question": "Please tell me if you have a very favorable, somewhat
3625    ↪ favorable, somewhat unfavorable, or very unfavorable opinion of...
3626    ↪ Australia", "options": ["Very favorable", 'Somewhat favorable', '
3627    ↪ Somewhat unfavorable', 'Very unfavorable']}<end_of_turn>
3628<start_of_turn>output
3629 [{"Very favorable": 20}, {"Somewhat favorable": 72}, {"Somewhat
3630    ↪ unfavorable": 7}, {"Very unfavorable": 1}]<end_of_turn>
3631<start_of_turn>input
3632 {"question": "I'd like your opinion about some possible international
3633    ↪ concerns for your country. Do you think that ___ is a major threat, a
3634    ↪ minor threat, or not a threat to your country? i. Longstanding
3635    ↪ conflicts between countries or ethnic groups", "options": ["Major
3636    ↪ threat", 'Minor threat', 'Not a threat', 'DK/Refused']}<end_of_turn>
3637 ...
3638
3639 habermas_individual
3640<start_of_turn>description
3641 UK resident responses. They were given a question and a statement, asked
3642    ↪ to express their opinion in 2-3 sentences (opinion.text) and their
3643    ↪ level of agreement with it on a 7-point scale (ratings.agreement).<
3644    ↪ end_of_turn>
3645<start_of_turn>input
3646 {"question.text": "Should the UK continue to subsidise the arts?", "
3647    ↪ statement": "The UK should continue to subsidise the arts."}<
3648    ↪ end_of_turn>
3649<start_of_turn>output
3650 {"opinion.text": "I do not think the UK should continue to subsidise the
3651    ↪ arts because I think that money could be better spent. For example, it
3652    ↪ could be used to subsidise healthcare degrees to promote people to
3653    ↪ enter the workforce to make up for staff shortages. It could be put
3654    ↪ towards health and education funding. Arts are important, but I do not
3655    ↪ think a degree is always necessary to pursue a career in the arts.", "
3656    ↪ ratings.agreement": "Disagree"}<end_of_turn>
3657<start_of_turn>input
3658 {"question.text": "Does the UK need a constitution?", "statement": "The
3659    ↪ UK needs a constitution."}<end_of_turn>
3660<start_of_turn>output
3661 {"opinion.text": "I do not think the UK needs a constitution. I think the
3662    ↪ UK is multicultural and there is no single constitution that could
3663    ↪ accurately convey all the values of the British people. I also do not
3664    ↪ think people need a constitution to act morally. People should live how
3665    ↪ they want to live, within the law. We do not need the government to
3666    ↪ prescribe a set of values.", "ratings.agreement": "Somewhat
3667    ↪ Disagree"}<end_of_turn>
3668<start_of_turn>input
3669 {"question.text": "Does the UK need a minimum price for alcohol?", "
3670    ↪ statement": "The UK needs a minimum price for alcohol"}<end_of_turn>
3671<start_of_turn>output
3672 {"opinion.text": "The UK does need a minimum price for alcohol because
3673    ↪ there is a big drinking problem in the UK. Adding a minimum price will
3674    ↪ discourage binge drinking which can improve the health of the
3675    ↪ population, similar to the sugar tax.", "ratings.agreement": "
3676    ↪ Agree"}<end_of_turn>
3677<start_of_turn>input
3678 {"question.text": "Would it be a good idea to introduce a four-day
3679    ↪ workweek?", "statement": "It would be a good idea to introduce a four
3680    ↪ -day workweek."}<end_of_turn>
3681<start_of_turn>output

```

```

3672 | {"opinion.text": "I think a four-day workweek is a good idea. It has been
3673 | shown to increase productivity and happiness among workers and gives
3674 | people more time to spend with the family, friends, and hobbies.
3675 | However, some jobs likely would not be affected, such as healthcare
3676 | workers. These people may not feel a four-day workweek is fair as they
3677 | are left out.", "ratings.agreement": "Somewhat Agree"}<end_of_turn>
3678 | <start_of_turn>input
3679 | {"question.text": "Should we ban the sale of single-use plastics such as
3680 | → plastic straws?", "statement": "We should ban the sale of single-use
3681 | → plastics such as plastic straws."}<end_of_turn>
3682 | <start_of_turn>output
3683 | {"opinion.text": "I think we should completely ban all single-use
3684 | plastics because they have a huge negative environmental impact. I
3685 | think finding more sustainable ways to package products and use
3686 | materials is vital for the future of the planet. Single-use plastics
3687 | fill up landfills and end up in the oceans. The harm animals and could
3688 | potentially even harm us directly by producing microplastics.", "ratings.agreement": "Strongly Agree"}<end_of_turn>
3689 | <start_of_turn>input
3690 | {"question.text": "Should the King have any say in choosing the Prime
3691 | → Minister?", "statement": "The King should have some say in choosing
3692 | → the Prime Minister."}<end_of_turn>
3693 | <start_of_turn>output
3694 | {"opinion.text": "I think the King should have a very small say in
3695 | choosing the prime minister as head of state. However, ultimately it
3696 | still must be up to democracy. I do not think the King should ever be
3697 | allowed to overrule the vote of the people.", "ratings.agreement": "Somewhat Disagree"}<end_of_turn>
3698 | <start_of_turn>input
3699 | {"question.text": "Should the government be allowed to buy land and give
3700 | → it to the poor?", "statement": "The government should be allowed to
3701 | <b>habermas_question</b>
3702 | <start_of_turn>description
3703 | Generate a list of diverse questions.<end_of_turn>
3704 | <start_of_turn>output
3705 | Should universities be allowed to increase tuition fees at any level they
3706 | want?<end_of_turn>
3707 | <start_of_turn>output
3708 | Should we ban all single-use plates and cutlery?<end_of_turn>
3709 | <start_of_turn>output
3710 | Should we raise the minimum wage to £12/hour?<end_of_turn>
3711 | <start_of_turn>output
3712 | Do we need to change the law to regulate the spread of fake
3713 | news?<end_of_turn>
3714 | <start_of_turn>output
3715 | Should the government require every new building in the UK to be designed
3716 | to be carbon-neutral?<end_of_turn>
3717 | <start_of_turn>output
3718 | Should universities be allowed to set their own tuition fees?<end_of_turn>
3719 | <start_of_turn>output
3720 | Should the government provide free higher education to all?<end_of_turn>
3721 | <start_of_turn>output
3722 | Should we legalise some drugs for recreational use?<end_of_turn>
3723 | <start_of_turn>output
3724 | Should we increase taxes on sugar-sweetened drinks?<end_of_turn>
3725 | <start_of_turn>output
3726 | Should the monarchy be replaced by a democratic republic?<end_of_turn>
3727 | <start_of_turn>output
3728 | Should the BBC have an option to increase the licence fee to fund a new
3729 | BBC News channel?<end_of_turn>
3730 | <start_of_turn>output
3731 | ...
3732 | 
```

3726 | Should the state provide universal child care for working
 3727 | parents?<end_of_turn>
 3728 | <start_of_turn>output
 3729 | Should the UK cut subsidies to farmers?<end_of_turn>
 3730 | <start_of_turn>output
 3731 | Does the UK have a moral duty to admit more refugees?<end_of_turn>
 3732 | <start_of_turn>output
 3733 | Should the UK have a universal basic income for all citizens?<end_of_turn>
 3734 | <start_of_turn>output
 3735 | Should the government spend less on the military and more on social
 3736 | welfare?<end_of_turn>
 3737 | <start_of_turn>output
 3738 | Should the government require all houses to have solar
 3739 | panels?<end_of_turn>
 3740 | <start_of_turn>output
 3741 | Is it okay for people to hunt for sport?<end_of_turn>
 3742 | <start_of_turn>output
 3743 | Should we give free access to the National Health Service for
 3744 | everyone?<end_of_turn>
 3745 | <start_of_turn>output
 3746 | Is it right for the BBC to broadcast content that some people consider to
 3747 | be too offensive?<end_of_turn>
 3748 | <start_of_turn>output
 3749 | Should we raise the retirement age from 66 to 68?<end_of_turn>
 3750 | <start_of_turn>output
 3751 | Should we ban non-essential plastics from supermarkets?<end_of_turn>
 3752 | <start_of_turn>output
 3753 | Should people be allowed to ride bikes on sidewalks?<end_of_turn>
 3754 | <start_of_turn>output
 3755 | Should we raise taxes on alcohol and sugary drinks?<end_of_turn>
 3756 | <start_of_turn>output
 3757 | Should there be an independent Scotland?<end_of_turn>
 3758 | <start_of_turn>output
 3759 | Should we force landlords to install insulation before renting their
 3760 | property to a new tenant?<end_of_turn>
 3761 | <start_of_turn>output
 3762 | Should we allow both men and women to serve in the military?<end_of_turn>
 3763 | <start_of_turn>output
 3764 | Are people less racist today than they were thirty years ago?<end_of_turn>
 3765 | <start_of_turn>output
 3766 | Should the government fund research into a cure for baldness?<end_of_turn>
 3767 | <start_of_turn>output
 3768 | Should it be legal for someone to sell their own body organs?<end_of_turn>
 3769 | <start_of_turn>output
 3770 | Should everyone have to pay an income tax even if they are earning less
 3771 | than the current minimum wage?<end_of_turn>
 3772 | <start_of_turn>output
 3773 | ["This is frustrating", "You're wasting time", "You're probably going to
 3774 | be late", "This has caused you a lot of stress", "Other people are also
 3775 | stuck in traffic", "This is because there was an accident", "The police
 3776 | are at the scene", "An ambulance is at the scene", "There is a lot of
 3777 | congestion", "People are honking their horns"]<end_of_turn>
 3778 | <start_of_turn>output
 3779 | PersonY gets upset easily<end_of_turn>
 3780 | <start_of_turn>output
 3781 | ["PersonY is sensitive", "PersonY overreacts to things", "PersonY
 3782 | probably has a lot of emotions", "This makes it hard for PersonY to

```

3780 control how they feel", "When something bad happens, PersonY may cry",
3781 "PersonY might get frustrated often", "Other people may not understand
3782 why PersonY gets upset so easily", "PersonY may need some help dealing
3783 with their emotions", "PersonY could try to talk to a therapist about
3784 this issue"]<end_of_turn>
3785 <start_of_turn>input
3786 PersonY sprints away in the other direction<end_of_turn>
3787 <start_of_turn>output
3788 ["PersonY is probably scared", "PersonY might be being chased", "PersonY
3789 could be running to safety", "This is probably an emergency", "PersonY
3790 may be out of breath", "PersonY's heart rate is probably high",
3791 "PersonY will probably get tired soon"]<end_of_turn>
3792 <start_of_turn>input
3793 You're on the phone and someone interrupts you.<end_of_turn>
3794 <start_of_turn>output
3795 ["You were talking on the phone", "You are talking to someone else", "The
3796 other person is rude", "You were probably in the middle of a
3797 conversation", "The other person may not have realized you were on the
3798 phone", "You may have been interrupted in the middle of your sentence",
3799 "As a result, you didn't get to finish your conversation", "This made
3800 you feel frustrated"]<end_of_turn>
3801 <start_of_turn>input
3802 John moves to a new city<end_of_turn>
3803 <start_of_turn>output
3804 ["John is leaving his old city", "John has somewhere to live in the new
3805 city", "John may know people in the new city", "John has a job in the
3806 new city", "John will probably sell his house", "John's commute will be
3807 different", "John may have to get used to a new climate", "In the end,
3808 John will be living in the new city"]<end_of_turn>
3809 <start_of_turn>input
3810 Somebody stole your bike from the park.<end_of_turn>
3811 <start_of_turn>output
3812 ["This event may have ruined the person's day", "The person may be
3813 angry", "The bike was probably left in an unsecure location", "The
3814 person will probably have to buy a new bike", "This event could have
3815 been prevented if the person remembered to lock their
3816 bike"]<end_of_turn>
3817 <start_of_turn>input
3818 PersonR and PersonS move in together."<end_of_turn>
3819 <start_of_turn>output
3820 ["They are going to live in the same space", "They will share resources",
3821 "PersonR and PersonS probably know each other pretty well", "PersonR
3822 and PersonS are adults", "This may be a big decision", "They will sign
3823 a lease", "The lease is probably for an apartment", "After, PersonR and
3824 PersonS will be roommates"]<end_of_turn>
3825 <start_of_turn>input
3826 Strange lights appear in the sky, followed by loud sounds and tidal waves
3827 ↪ .<end_of_turn>
3828 <start_of_turn>output
3829 ["This event is strange", "The lights are probably UFOs", "The sound
3830 could be an explosion", "Many people saw the light", "This event may
3831 have caused damage", "Tidal waves are usually big and dangerous", "This
3832 event was probably unexpected", "People are probably scared", "Some
3833 people may have died", "Officials are probably investigating",
3834 "Scientists may try to explain this event", "It is possible that this
3835 event was a natural disaster"]<end_of_turn>
3836 <start_of_turn>input
3837 You're running short on cash and can't purchase anything because of it.<
3838 ↪ end_of_turn>
3839 ...
3840
3841 novacomet_premise
3842 <start_of_turn>description
3843 Generate an event.<end_of_turn>
3844 <start_of_turn>output

```

```

3834 You have forgotten your PIN number and need to reset it.<end_of_turn>
3835 <start_of_turn>output
3836 SOMETHING HAPPENS!!!, just about anything can happen!<end_of_turn>
3837 <start_of_turn>output
3838 Alex makes a discovery while exploring outer space<end_of_turn>
3839 <start_of_turn>output
3840 You are at a party.<end_of_turn>
3841 <start_of_turn>output
3842 A fight between two of your closest friends causes lasting
3843 damage.<end_of_turn>
3844 <start_of_turn>output
3845 PersonX doesn't have healthy coping mechanisms when things go
3846 wrong<end_of.turn>
3847 <start_of_turn>output
3848 PersonX needs to laundry<end_of.turn>
3849 <start_of_turn>output
3850 You cook dinner.<end_of.turn>
3851 <start_of_turn>output
3852 You get lost in the city.<end_of.turn>
3853 <start_of_turn>output
3854 Time changes and events that once seemed far away draw near for
3855 Mark<end_of.turn>
3856 <start_of_turn>output
3857 Today you plan your day and decide what to wear.<end_of.turn>
3858 <start_of_turn>output
3859 Your car has broken down and you have to find a ride.<end_of.turn>
3860 <start_of_turn>output
3861 Nathan makes a typo in a paper and has to go back and fix it<end.of.turn>
3862 <start_of_turn>output
3863 Somebody sneezes<end.of.turn>
3864 <start_of_turn>output
3865 A major pandemic sweeps through the world, killing millions.<end.of.turn>
3866 <start_of_turn>output
3867 Your significant other got mad at you and they're not talking to you
3868 anymore.<end.of.turn>
3869 <start_of_turn>output
3870 You go to put your phone in your pocket and it slips out and falls into
3871 the toilet.<end.of.turn>
3872 <start_of_turn>output
3873 PersonX forgot their passport and can't travel<end.of.turn>
3874 <start_of_turn>output
3875 Christopher visits his family in Spain<end.of.turn>
3876 <start_of_turn>output
3877 There was an earthquake near where the reader lives. Everyone is
3878 evacuated from their homes.<end.of.turn>
3879 <start_of_turn>output
3880 The car stalls on the freeway<end.of.turn>
3881 <start_of_turn>output
3882 You have to pick up your sister from soccer practice.<end.of.turn>
3883 <start_of_turn>output
3884 A drawer is pulled out.<end.of.turn>
3885 <start_of_turn>output
3886 PersonX has a conversation with a stranger<end.of.turn>
3887 <start_of_turn>output
3888 Jeffery is angry<end.of.turn>
3889 <start_of_turn>output
3890 You are surrounded by silence.<end.of.turn>
3891 <start_of_turn>output
3892 PersonX says that they don't have any experience fishing<end.of.turn>
3893 <start_of_turn>output
3894 numbergame_perc
3895 <start_of_turn>description
3896 The following is a number game task. People were shown a set of numbers,
3897 ↪ and asked whether a target number was likely to be generated by the

```

```

3888    ↪ same process as the set. Your goal is to predict the percentage of
3889    ↪ people who would say yes to the target number.<end_of_turn>
3890    <start_of_turn>input
3891    {"given_numbers": "66", "target_number": "29"}<end_of_turn>
3892    <start_of_turn>output
3893    25%<end_of_turn>
3894    <start_of_turn>input
3895    {"given_numbers": "8, 16", "target_number": "18"}<end_of_turn>
3896    <start_of_turn>output
3897    11%<end_of_turn>
3898    <start_of_turn>input
3899    {"given_numbers": "69, 9, 39, 21", "target_number": "16"}<end_of_turn>
3900    <start_of_turn>output
3901    15%<end_of_turn>
3902    <start_of_turn>input
3903    {"given_numbers": "100", "target_number": "20"}<end_of_turn>
3904    <start_of_turn>output
3905    58%<end_of_turn>
3906    <start_of_turn>input
3907    {"given_numbers": "7, 67", "target_number": "56"}<end_of_turn>
3908    <start_of_turn>output
3909    13%<end_of_turn>
3910    <start_of_turn>input
3911    {"given_numbers": "64, 4", "target_number": "28"}<end_of_turn>
3912    <start_of_turn>output
3913    77%<end_of_turn>
3914    <start_of_turn>input
3915    {"given_numbers": "16, 54", "target_number": "53"}<end_of_turn>
3916    <start_of_turn>output
3917    22%<end_of_turn>
3918    <start_of_turn>input
3919    {"given_numbers": "59, 14", "target_number": "5"}<end_of_turn>
3920    <start_of_turn>output
3921    11%<end_of_turn>
3922    <start_of_turn>input
3923    {"given_numbers": "50", "target_number": "10"}<end_of_turn>
3924    <start_of_turn>output
3925    92%<end_of_turn>
3926    <start_of_turn>input
3927    {"given_numbers": "85, 19, 91", "target_number": "14"}<end_of_turn>
3928    <start_of_turn>output
3929    11%<end_of_turn>
3930    <start_of_turn>input
3931    {"given_numbers": "78", "target_number": "92"}<end_of_turn>
3932    <start_of_turn>output
3933    50%<end_of_turn>
3934    <start_of_turn>input
3935    {"given_numbers": "68, 14, 8, 26", "target_number": "22"}<end_of_turn>
3936    <start_of_turn>output
3937    77%<end_of_turn>
3938    <start_of_turn>input
3939    {"given_numbers": "4, 16, 12", "target_number": "56"}<end_of_turn>
3940    <start_of_turn>output
3941    54%<end_of_turn>
3942    <start_of_turn>input
3943    {"given_numbers": "37, 57", "target_number": "19"}<end_of_turn>
3944    <start_of_turn>output
3945    10%<end_of_turn>
3946    <start_of_turn>input
3947    {"given_numbers": "3, 63", "target_number": "28"}<end_of_turn>
3948    <start_of_turn>output
3949    9%<end_of_turn>
3950    <start_of_turn>input
3951    {"given_numbers": "92, 68, 20", "target_number": "63"}<end_of_turn>
3952    <start_of_turn>output

```

```

3942 8%<end_of_turn>
3943 <start_of_turn>input
3944 {"given_numbers": "1", "target_number": "70"}<end_of_turn>
3945 <start_of_turn>output
3946 0%<end_of_turn>
3947 <start_of_turn>input
3948 {"given_numbers": "26", "target_number": "64"}<end_of_turn>
3949 <start_of_turn>output
3950 50%<end_of_turn>
3951 <start_of_turn>input
3952 {"given_numbers": "3, 7", "target_number": "35"}<end_of_turn>
3953 <start_of_turn>output
3954 56%<end_of_turn>
3955 <start_of_turn>input
3956 {"given_numbers": "52, 22, 94", "target_number": "3"}<end_of_turn>
3957 <start_of_turn>output
3958 0%<end_of_turn>
3959 <start_of_turn>input
3960 {"given_numbers": "33, 17, 5, 9", "target_number": "12"}<end_of_turn>
3961 <start_of_turn>output
3962 11%<end_of_turn>
3963 <start_of_turn>input
3964 {"given_numbers": "11, 26, 74, 2", "target_number": "4"}<end_of_turn>
3965 <start_of_turn>output
3966 60%<end_of_turn>
3967 <start_of_turn>input
3968 {"given_numbers": "22, 96", "target_number": "64"}<end_of_turn>
3969 <start_of_turn>output
3970 70%<end_of_turn>
3971 <start_of_turn>input
3972 {"given_numbers": "77, 17, 8", "target_number": "61"}<end_of_turn>
3973 <start_of_turn>output
3974 11%<end_of_turn>
3975 <start_of_turn>input
3976 {"given_numbers": "49", "target_number": "9"}<end_of_turn>
3977 <start_of_turn>output
3978 39%<end_of_turn>
3979 <start_of_turn>input
3980 {"given_numbers": "63, 67", "target_number": "36"}<end_of_turn>
3981 ...
3982
3983
3984
3985
3986
3987
3988
3989
3990
3991
3992
3993
3994
3995

```

3978 O.3 ADDITIONAL EXAMPLE TASK PROMPTS

3979 For example prompts for all task, please see ANONYMIZED

3980
 3981
 3982
 3983
 3984
 3985
 3986
 3987
 3988
 3989
 3990
 3991
 3992
 3993
 3994
 3995