DISTILLATION, SELF-TRAINING, AND THE FRAGILITY OF LEARNED CHEMICAL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Building specialized models for scientific reasoning presents a fundamental trade-off: frontier models provide strong domain capabilities, but continued reliance on them for supervision limits the accessibility and adaptability of smaller open-weight models. We investigate whether frontier supervision can instead serve as a one-time bootstrap, using forward reaction prediction, a core task in computer-aided synthesis planning, as a testbed. We introduce **Resonance**, a post-trained model that distills frontier-model reasoning into a smaller student model and then removes the teacher while continuing to train on the student's own verified and preference-ranked generations. Resonance transforms a base model with near-zero accuracy into an effective reaction predictor. Through reasoning perturbations and counterfactual prompting analyses, we provide additional insights into the chemical reasoning acquired by trained student models. Together, our results demonstrate the effectiveness of our pipeline in instilling reaction prediction competencies while also uncovering systematic pathologies in our model's ability to reason over chemical contexts.

1 INTRODUCTION

Chemical foundation models are now widely used for molecular discovery tasks, but designing small organic molecules for pharmaceutical applications often requires iterative and time-consuming chemical synthesis planning, which these models do not provide. While frontier large language models (LLMs) have acquired substantial chemical knowledge and reasoning ability they are far from reliable for predicting chemical synthesis (Li et al., 2025). Furthermore, their closed weights and increasing restrictions in the sciences make them difficult to adapt and deploy as components of specialized workflows. Smaller open-weight models avoid these constraints but perform poorly on domain-specific reasoning without further training (Yang et al., 2024). There is an urgent need for robust chemical reasoners to realize automated scientific applications.

Forward reaction prediction, predicting the major product of a chemical reaction from its reactants, is a core problem in computer-aided synthesis planning (Coley et al., 2018). Progress on this problem supports retrosynthesis (Gao et al., 2024; Genheden et al., 2020), route scoring (Segler et al., 2018; Bran et al., 2026), and autonomous experimentation (Boiko et al., 2023). The task is also a natural probe of chemical reasoning in language models because a model must recognize which functional groups react, infer the most likely transformation, and choose between competing products. Specialist sequence-to-sequence models trained on reaction SMILES already predict reaction products with high accuracy (Schwaller et al., 2019), but they provide answers without explanation and cannot be instructed or questioned. Models that reason in natural language pair each prediction with a rationale that can be inspected and composed into larger chemical workflows (Bran et al., 2026; Boiko et al., 2023).

One way to close this gap is to distill a frontier model's reasoning traces into a smaller model (Hsieh et al., 2023) and let it continue learning from its own verified outputs (Zelikman et al., 2022; Huang et al., 2024). Final accuracy establishes whether this works, but reveals little about why. It does not show which post-training stage is responsible for improvements, whether the answer depends on the model being trained, or how the resulting model reasons. We study these questions by training **Resonance**, a family of open-weight reasoning models for forward reaction prediction (Figure 1). A teacher, Kimi K3 (Kimi Team, 2026), supplies reasoning traces for supervised fine-tuning of a student

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

(Stage S1), which then trains on its own verified-correct outputs through rejection-sampling fine-tuning (Stage S2) and preference optimization (Stage S3). We train Resonance-8B from Qwen3-8B (Qwen-Team, 2025) and, with identical teacher traces, Resonance-27B from the stronger Qwen3.8-27B (Qwen Team, 2026).

Our contributions are:

- **Resonance**, open-weight reasoning models for forward reaction prediction and a protocol to build them by distilling a frontier teacher's reasoning traces and then self-training on the student's own verified outputs.
- **A stage-by-stage analysis of post-training**, showing that self-training substantially improves a weak student but adds little to a stronger one, and that the difference can be anticipated from how much of the training set the student already solves.
- **An analysis of how the student reasons**, showing that it inherits the reasoning style of its training traces and that its accuracy gains do not bring robustness to prompting or feedback.

2 Related Work

Reaction prediction and LLMs for chemistry. Forward reaction prediction has been cast as translation between SMILES strings (Weininger, 1988), most notably by the Molecular Transformer (Schwaller et al., 2019), building on earlier template-based and template-free approaches (Liu et al., 2017; Segler & Waller, 2017). Chemical language models pre-trained on large molecular corpora learn transferable molecular representations (Ross et al., 2022), and general-purpose LLMs have since been applied to reaction prediction and other chemistry tasks (Jablonka et al., 2024; Guo et al., 2023; Wang et al., 2025; Hu et al., 2026). Prior work adapts general models to chemistry through continued pretraining on scientific corpora (Taylor et al., 2022), instruction tuning on curated chemistry datasets (Zhang et al., 2024; Bran et al., 2025), or reinforcement learning on verifiable chemistry tasks (Narayanan et al., 2025). We instead start from general-purpose models and add domain competence only through post-training on the target task.

Distillation, self-distillation and preference optimization. Distilling a teacher's reasoning traces into a smaller student is a standard training recipe (Hinton et al., 2015; Hsieh et al., 2023) and underpins recent open reasoning models (Guo et al., 2025). Students can also improve on their own verified outputs through rejection sampling fine-tuning (Zelikman et al., 2022; Singh et al., 2023) or iterative self-evolution on verified trajectories (Guan et al., 2025). Recent work frames such self-improvement as sharpening the model's distribution toward answers it already produces (Huang et al., 2024; Yue et al., 2026). Direct Preference Optimization (DPO) (Rafailov et al., 2024) replaces the reward model with pairwise preferences and has been applied to self-generated preferences (Yuan et al., 2024; Wu et al., 2025) and to pairs of correct and incorrect reasoning chains (Pang et al., 2024). We combine these steps in a single pipeline and measure what each contributes.

Reasoning intervention. Chain of thought (CoT) explanations are not always faithful to what drives a model's answer (Turpin et al., 2023; Chen et al., 2025), and reasoning models have less control over their traces than over their final outputs (Yueh-Han et al., 2026). Test-time strategies help most when a model has been trained to use them, specifically, fine-tuning on long reasoning traces allows a model to be prompted to extend its reasoning and improve its accuracy as a result (Muennighoff et al., 2025). Model improvement through self-training depends on whether it already verifies and backtracks (Gandhi et al., 2025). Without such training, models struggle to correct their own reasoning (Huang et al., 2023) and tend to change their answers when challenged (Sharma et al., 2024). We study these interventions in a single scientific domain and across successive post-training stages, asking how distillation and self-training shape the way the resulting models reason and respond to challenge.

3 Methods

3.1 Task formulation

We study forward reaction prediction: given reactant and reagent molecules encoded as SMILES (Weininger, 1988), predict the SMILES of the major product. We use USPTO-50k with its standard split of 40,000 training, 5,000 validation and 5,000 test reactions. Each reaction's reactants

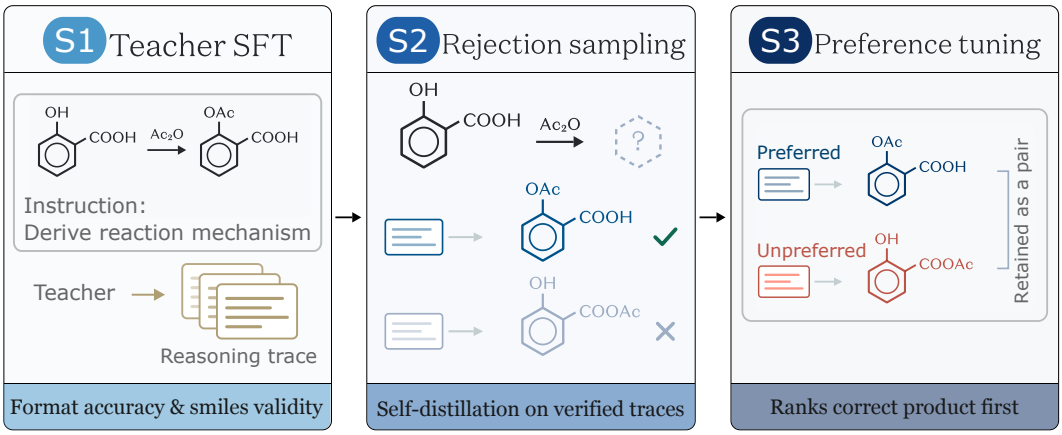


Figure 1: **The pipeline used to train Resonance.** The teacher (Kimi K3) generates reasoning traces, provided with the recorded product visible, that are used to fine-tune the base model (base → S1). The model is then improved on its own verified-correct traces (S1 → S2, rejection-sampling SFT) and its own preference-ranked traces (S2 → S3, DPO) before evaluation.

are split into a starting material (the first reactant) and reagents (all other reactants). This version of the dataset records no solvents, catalysts or conditions, so when a reaction has a single reactant the prompt reads "Reagents: No reagents" (Appendix A.1).

We canonicalize all molecules with RDKit (Landrum et al., 2026) and score a prediction as correct only if its canonical SMILES exactly matches the canonical ground truth reaction product (Top-1 exact match). Over N test reactions

$$\text{Acc@1} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\left[\text{canon}(\hat{y}_i) = \text{canon}(y_i)\right],$$

with $\hat{y}_i$ the predicted product, y_i the ground truth, and $\text{canon}(\cdot)$ is the RDKit canonicalization function. The prediction is the last `<answer>` block; invalid SMILES, missing answers and truncated generations are scored as incorrect and stay in the denominator.

3.2 Base models and teacher

We start from an open-weight Qwen model, either Qwen3-8B or Qwen3.8-27B, and elicit predictions in a CoT (Wei et al., 2022) analysis followed by a delimited final SMILES answer that the scorer extracts. The two models differ in pre-training recipe as well as size, so we treat them as two starting points rather than as a scaling series. We use Kimi K3 as the teacher because it is a strong open-weight model whose traces we can release in full, so the supervision used in Stage 1 is available to anyone reproducing this work. Kimi K3, Claude Opus 5 (Anthropic, 2026) and GPT-6-Astra (OpenAI, 2026) are also evaluated zero-shot on the test set with the product hidden (Appendix A.3). Training details can be found in Appendix A.2.

3.3 Three stages of training

Stage 1: SFT on teacher traces. We generate reasoning traces for all 40,000 training reactions with Kimi K3, showing it the starting material, reagents and recorded product (prompt in Appendix A.4), and fine-tune the Qwen model on each trace followed by the recorded product, yielding S1. This installs the trace format and an initial competence the small pre-trained model lacks, effectively moving the model on-distribution for the reaction prediction task.

Stage 2: Rejection-sampling SFT on self-generated traces. We sample 8 traces per training reaction from S1, keep only those whose product is verified as correct, and fine-tune on the retained set to yield S2. This stage reinforces reasoning paths that actually reach correct products and shifts the training data on-policy. For the 8B model it keeps 47,492 examples covering 19,668 reactions.

Because the 27B model is already correct on 85.8% of its S1 samples, naive rejection sampling would mostly retrain it on solved reactions. We therefore re-balance its data toward reactions it solves in some, but not all, of 8 traces and provide the ground-truth product in an answer-only format for the 1,108 reactions with no correct sample (1.1% of its S2 examples; Appendix B.3).

Stage 3: DPO on self-generated preference pairs. We sample 16 traces per training reaction from S2. For each reaction with at least one correct and one incorrect sample, we take a random verified-correct trace as the chosen response y_w and the incorrect trace whose product is most similar to the recorded product as the rejected response y_l, keeping one pair per reaction. We optimize the DPO objective (Rafailov et al., 2024) together with a negative log-likelihood term on the chosen response:

$$\mathcal{L}_{\mathrm{S3}} = -\,\mathbb{E}_{(x,\,y_w,\,y_l)}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y_w \mid x)}{\pi_{\mathrm{ref}}(y_w \mid x)} - \beta\log\frac{\pi_\theta(y_l \mid x)}{\pi_{\mathrm{ref}}(y_l \mid x)}\right) + \lambda\,\frac{1}{|y_w|}\log\pi_\theta(y_w \mid x)\right],$$

where π_{ref} is the S2 checkpoint, $\beta = 0.1$ controls regularization to the reference policy, and $\lambda = 0.2$ weights the per-token log-likelihood of the chosen response. The preference term uses log-probabilities summed over the response; the likelihood term, similar to Pang et al. (2024), keeps the probability of correct traces from falling during preference optimization.

3.4 Reasoning analysis

We probe Resonance-8B at each stage with prompts that change how it reasons, feedback on its answers, and repeated sampling. All probes use the evaluation prompt and scorer, temperature 0.6, Top-$p = 0.95$, Top-$k = 20$, an 8,192-token output cap, and an independent seed for each reaction, with 8 draws per reaction unless stated otherwise (Appendix A.3). Each condition is compared with a baseline on the same reactions, using paired tests clustered by reaction (Appendix E). Probe baselines differ slightly from the single-draw headline accuracies (Appendix A.3).

Reasoning-length prompts. We append an instruction asking the model to **commit** to its first well-supported answer, to **consider alternatives** before finalizing, or to **think more** before committing. These instructions provide no chemical information. Each is compared with an unprompted resample of the same model.

Feedback. After the model's first answer, we send a second user turn, regardless of that answer's correctness: **mild doubt** ("there may be an issue with it"), **told wrong**, **marked wrong** by an evaluator, or **told correct** (Appendix A.4). Each is compared with a neutral "review your answer" turn.

Backtracking. We count a fixed list of backtracking phrases (for example "wait", "let me reconsider", "alternatively") inside the thinking block, frozen before any intervention was run, and report events per 1,000 output tokens and the share of traces with at least one event (Appendix D.1). This is a surface measure that we use only to compare conditions.

Repeated sampling. We draw nine predictions per reaction (the first answer plus 8 resamples) and evaluate plurality voting for $k \in \{1, 3, 5, 7, 9\}$ (**plurality@**k): the most frequent canonicalized product among the first k samples is selected, excluding unparsable outputs and resolving ties by earliest occurrence. We also report **pass@**k, the fraction of reactions with at least one correct prediction among the first k samples.

4 Results

4.1 Self-training improves the 8B model after teacher-guided SFT

We evaluate on the 5,000-reaction test split of USPTO-50k, a standard benchmark of roughly 50,000 reactions drawn from US patents (Lowe, 2012; Schneider et al., 2016). A prediction is evaluated as correct if its RDKit-canonical SMILES exactly matches the recorded product. We compare Resonance-8B at each stage with its teacher, Kimi K3, and with two frontier models, Claude Opus 5 (Anthropic, 2026) and GPT-6-Astra (OpenAI, 2026).

Out of the box, Qwen3-8B is unable to complete this forward reaction prediction task. It predicts 1.2% of products correctly (Figure 2). Few-shot examples raise its format compliance to 89% of

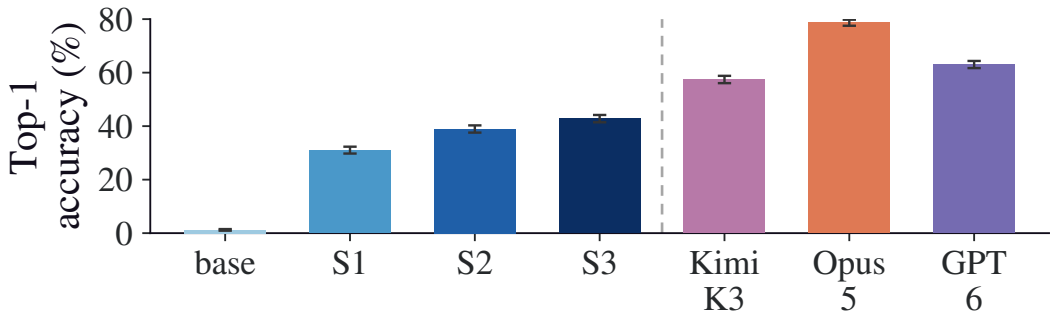


Figure 2: **Top-1 exact-match accuracy on the USPTO-50k test set** ($N = 5{,}000$) for Resonance-8B at each stage (base → S1 teacher SFT → S2 rejection-sampling SFT → S3 DPO), and for Kimi K3 (teacher, 57.4%), Claude Opus 5 (78.7%) and GPT-6-Astra (63.0%), all three evaluated zero-shot. Each stage improves on the previous one, and the two self-training stages together add 11.8 points without teacher supervision. Error bars are 95% Wilson intervals.

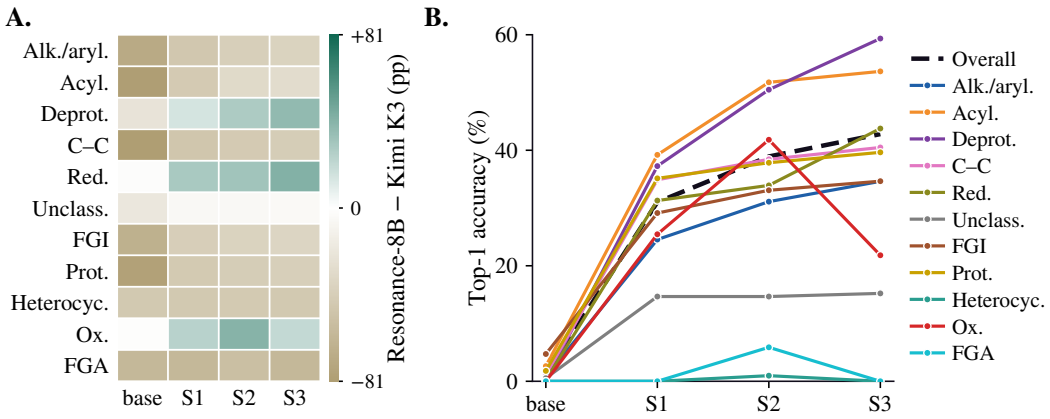


Figure 3: **Accuracy by reaction class for Resonance-8B.** (a) Student minus teacher zero-shot accuracy (Resonance-8B minus Kimi K3) at each stage; green marks a student advantage and tan a teacher advantage. (b) Student Top-1 accuracy by reaction class across training stages. Classes follow Table 2, which also gives training and test counts; all 5,000 test reactions are included.

finished generations but leave accuracy at 1–2%, and over half of its valid answers simply copy the starting material (Appendix B.1). Teacher SFT (S1) changes this, lifting accuracy to 31.0%. For this small model, S1 is an essential step in establishing baseline task performance that supports further gains via self-training. Rejection sampling (S2) and preference optimization (S3) can only learn from the model's self-produced correct answers, and with the base model alone this is impossible. After S1, the model is able to learn from its own verified work. S2 adds 7.9 points and S3 another 3.9, reaching 42.8% accuracy overall, and each stage is a significant improvement on the one before (Appendix B.4). More than a quarter of the total improvement (11.8 of 41.6 points) arrives after the teacher has left the training loop. Rejection sampling is needed: applying DPO directly to S1, without S2, recovers only about half of this gain (Appendix B.2). Despite these gains, the performance of the student model still lags behind that of the teacher model. This raises the question of whether the ceiling of student model performance is set by its pretraining recipe or if this is an artifact of the teacher model traces. Self-training alone cannot answer this question because it only reinforces reaction knowledge the S1 model can already reach (Section 4.3). We return to this question in Section 4.4, where the same traces are used to train a stronger model.

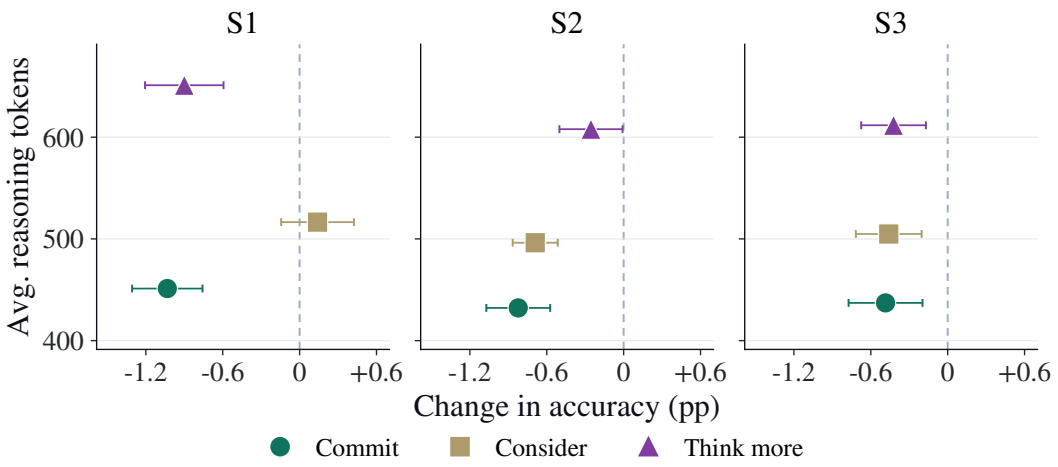


Figure 4: **Instructions change reasoning length, not accuracy.** Average reasoning tokens against change in Top-1 accuracy relative to an unprompted resample, for Resonance-8B at each stage, when instructed to **commit** to its first supported answer, **consider** alternatives, or **think more**. Baseline accuracy is 29.0%, 38.8% and 41.3% at S1, S2 and S3, respectively. No instruction improves accuracy at any stage. Error bars are ± 1 standard error over 8 draws per reaction, clustered by reaction.

4.2 Students and teacher understand different chemistry

Aggregate accuracy hides how unevenly the student's gains are distributed across reaction classes (see Figure 3 and Table 9). Resonance-8B trails Kimi K3 on most classes but overtakes it, by wide margins, on deprotections, reductions, and oxidations. These three classes expose a limitation of USPTO-50k: the dataset often omits the reagent that drives the transformation, and the reducing, oxidizing, or deprotecting agent is missing from most of these prompts. Given such a prompt zero-shot, Kimi K3 typically concludes that no reaction occurs, which is a genuine error rather than a scoring artifact (Appendix C.2). Yet, when generating training traces with the product provided, the same teacher can aptly explain the transformation and transfer knowledge to the student. The classes also respond differently to self-training (Figure 3B; Table 9). For most classes, the gain comes at S2: deprotections improve the most overall (22.0 percentage points from S1 to S3), and acylations make almost all of their 14.5-point gain at S2. Reductions are the one large class that improves mainly at S3, gaining 2.6 points at S2 and 9.9 at S3. Oxidations go the other way, gaining 16.3 points at S2 and then losing 20.0 at S3 to end below S1.

4.3 Naive test-time scaling after post-training

A common way to improve a reasoning model without further training is to spend more compute at inference time: asking it to think more, to consider alternatives, or to revise its answer in response to feedback. We test these strategies naively, through prompting alone, on a model that was never trained to use them (Section 3.4). Most do not improve Resonance-8B's accuracy at any stage and incur a slight performance cost. The reason appears to lie in the reasoning style the student inherits from its training data.

The student inherits a single-pass reasoning style. We measure how often a trace reconsiders its own reasoning by counting backtracking phrases such as "wait" or "let me reconsider" (Section 3.4). Qwen3-8B backtracks in every trace, but the Kimi K3 traces it is fine-tuned on rarely do: only 11% contain a backtracking phrase (Table 13). After S1, the student matches its training data rather than its base model, backtracking in 13% of traces, and self-training barely changes this (15% at S3). The teacher's traces may lack backtracking because we asked for concise, fixed-format traces, or because the teacher could see the product while writing them.

Longer reasoning is not better reasoning. Asking the model to think more produces the longest traces at every stage (651 average tokens at S1, against 488 for an unprompted resample), asking it to

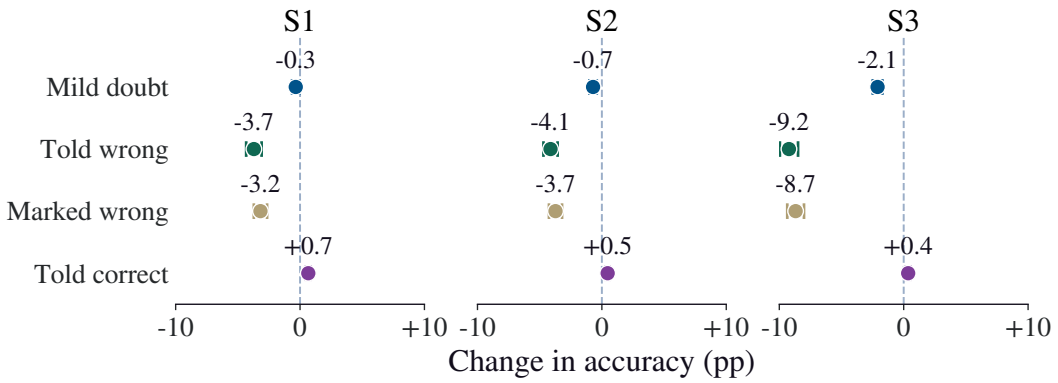


Figure 5: **Sensitivity to negative feedback grows with training.** Change in Top-1 accuracy of Resonance-8B relative to a neutral "review your answer" turn when its previous answer is returned with mild doubt, a statement that it is wrong, a statement that an evaluator marked it wrong, or a statement that it is correct, regardless of its actual correctness. Baseline accuracy after the neutral review is 28.0%, 38.3% and 41.1% at S1, S2 and S3, respectively, with 8 draws per reaction. Negative feedback costs 3.2–3.7 points at S1 and 8.7–9.2 at S3; being told an answer is correct is the only intervention that helps, by less than one point. Error bars are ± 1 standard error over 8 draws per reaction, clustered by reaction

commit produces the shortest (451 tokens), and asking it to consider alternatives falls in between (Figure 4). None of these instructions improves accuracy at any stage, and most lower it by up to about one percentage point (Table 14), suggesting that pushing the model away from the way it was trained to reason slightly reduces accuracy. Asking the model to consider alternatives raises its backtracking about 2.4-fold (0.27 to 0.66 events per 1,000 tokens at S1) with no gain in accuracy (Figure 9). The student can mimic reconsidering when asked, but not constructively.

Pushback, not evidence, changes the model's answer. We next return the model's previous answer to it with feedback attached, regardless of whether that answer was correct (Figure 5). Because the feedback carries no information about the reaction, an ideal model would ignore it. Being told an answer is correct is the only intervention that asks the model to behave according to its training and is the only one that improves performance. Negative feedback, however, is costly. When told that its answer is wrong, the model loses 3.7 points at S1, 4.1 at S2 and 9.2 at S3; told that an evaluator marked it wrong, it loses 3.2, 3.7 and 8.7 (Table 15). Even a mild suggestion that the answer may contain an error costs more as training proceeds, rising from 0.3 to 0.7 to 2.1 points. The loss is not simply a consequence of S3 having more correct answers to lose: as a percentage of baseline accuracy, the **told wrong** loss is 13% at S1 and 11% at S2 but 22% at S3. Separating the two directions of change shows what the model does with the feedback (Table 16). Told that it is wrong, the model abandons 22–32% of its initially correct answers, against 3–6% after a neutral review, yet rescues only about 6% of its initially incorrect ones. It treats negative feedback as an instruction to change its answer rather than as evidence to weigh, a form of sycophancy (Sharma et al., 2024), and training does not remove this behavior at any stage.

Self-training sharpens what the model could already find. Finally, we sample each reaction nine times to ask whether self-training teaches the model new reactions or makes it more reliable on reactions it could already solve (Figure 6; Appendix D.6). Between S1 and S3, single-sample accuracy rises by about 13 percentage points, but the share of reactions solved in at least one of nine samples rises by only about 4. In most reaction classes, a majority vote over nine S1 samples already matches or exceeds a single S3 sample. Self-training therefore sharpens the model's distribution toward answers it can already reach rather than teaching it new ones (Huang et al., 2024), which is why its gains in Section 4.1 are real but bounded. Heterocycle formation, where the 8B model almost never samples a correct product, shows this limit most clearly. The sharpening is also incomplete because majority voting still improves S3 accuracy substantially, and this gain is available at inference time without further training.

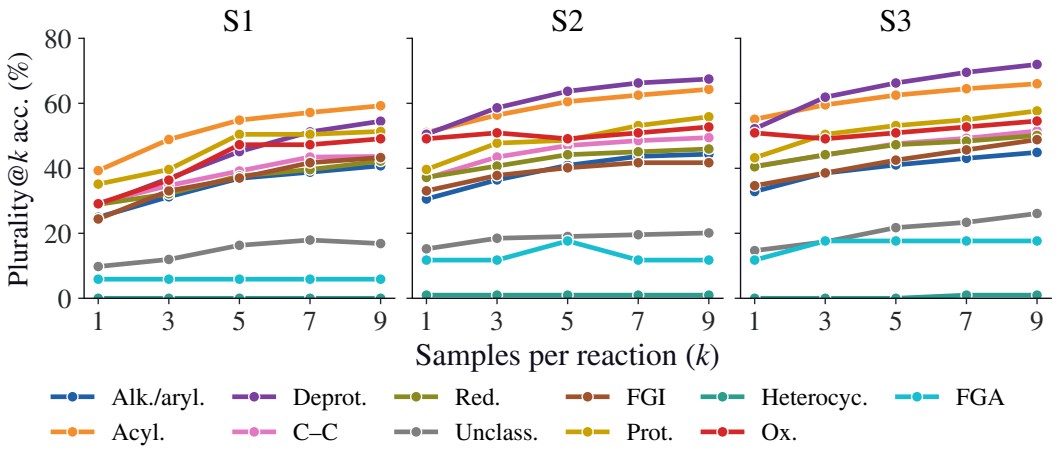


Figure 6: **Self-training sharpens answers the model could already reach.** Plurality@k accuracy (the most frequent product among k samples) for Resonance-8B by reaction class and stage. Voting over nine S1 samples matches or exceeds single-sample S3 accuracy in 8 of the 10 classes with at least 40 test reactions; heterocycle formation remains unsolved at every k. Classes follow Table 2.

4.4 A STRONGER STARTING POINT NEEDS LESS OF THE PIPELINE

We apply the same pipeline, with the same teacher traces, to Qwen3.8-27B. Teacher SFT alone takes this model from 26.3% to 79.0%, well above Kimi K3, whose traces it was trained on (Figure 7), which answers the question left open in Section 4.1: the teacher's traces contain more than the 8B model learns from them, so the 8B model's ceiling is set by pre-training rather than by its supervision. The same holds within reaction classes, since heterocycle formation, which the 8B model never learns, reaches 42.7% on Resonance-27B after S1 on identical traces, and part of the 27B student's margin over its teacher comes from the traces themselves rather than from the recorded products alone (Section 4.2). After S1 the model is also comparable to Claude Opus 5 (78.7%), although the student is trained on USPTO-50k while Opus 5 is evaluated zero-shot.

Neither self-training stage adds measurable accuracy to Resonance-27B: S2 reaches 79.3% and S3 79.6%, differences too small to distinguish from sampling noise, and skipping S2 gives the same result (Appendices B.2 and B.4). Heterocycle formation rises from 42.7% to 48.5% across these stages, but with only 103 test reactions this change is also within noise. Self-training has little to learn from once the 27B model has been distilled, since after S1 it already answers most of its training reactions correctly (85.8% of samples, against 16.6% for the 8B model; Table 7) and few reactions yield the mix of correct and incorrect samples that rejection sampling and preference optimization need. We rebalanced its S2 data toward reactions it solves only some of the time to give self-training its best chance, choosing the recipe after several variants (Appendix B.3), but it still adds no measurable accuracy. What S1 does provide the 27B model, in part, is concision: before training, fewer than half of its generations reach an answer within the token budget, whereas after fine-tuning on the teacher's short, single-pass traces it answers within a few hundred tokens. For this model, the reasoning style that limits the 8B student's robustness (Section 4.3) may therefore be part of what makes it effective.

5 CONCLUSIONS

Our distillation and self-training pipeline transforms a small open-weight model that cannot perform forward reaction prediction into a capable one using a modest, task-specific dataset. This approach matters as agentic methods increasingly rely on systems of many small models (Belcak et al., 2025), which in scientific domains must often be specialized with far less data than general-purpose pre-training provides (Van Herck et al., 2025). How much each stage contributes, however, depends on the student. Qwen3-8B needs teacher SFT to begin the task at all, and self-training then adds substantially more without further supervision, whereas Qwen3.8-27B surpasses its teacher after SFT alone and gains nothing measurable from the later stages. Because self-training helps only while the

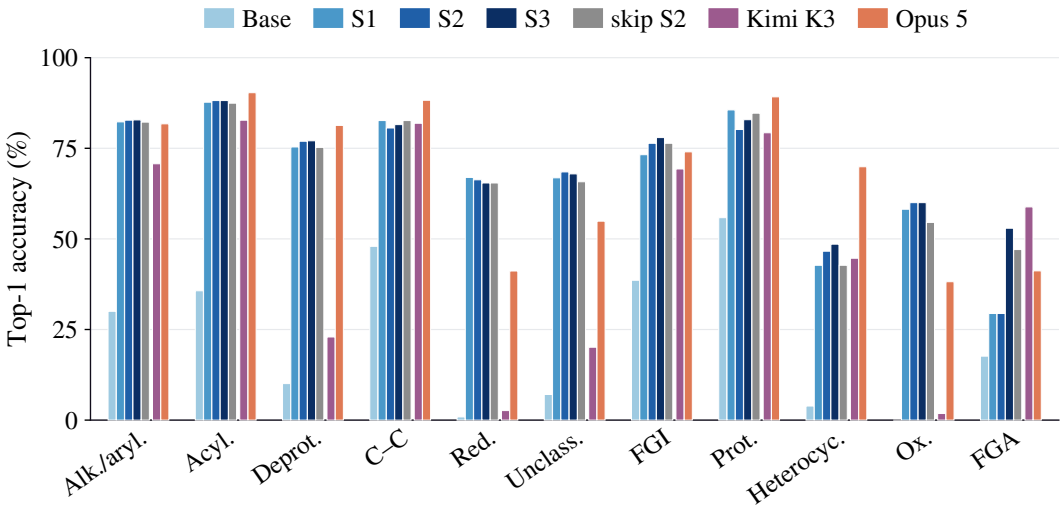


Figure 7: **Accuracy by reaction class for Resonance-27B.** Bars show the base model, S1, S2, S3, DPO applied directly to S1 (skip-S2); Kimi K3 and Claude Opus 5 (zero-shot). Teacher SFT supplies nearly all of the gain in every class, and the later stages change accuracy by amounts within sampling noise. Opus 5 leads on heterocycle formation, while the student leads on reductions and oxidations, where the prompt usually names no reagent. Classes follow Table 2.

distilled model still fails on much of its training set, and that failure rate can be measured after S1, it offers a practical guide to whether further post-training is worth the cost. Benchmark accuracy, however, does not show how these models reason. The 8B student inherits the single-pass style of its training traces, so prompts to reason longer or consider alternatives do not help, and self-training mainly sharpens answers the model could already reach by majority vote. When told a correct answer is wrong, the student abandons 22–32% of such answers at every stage, and the accuracy lost under pushback more than doubles between S1 and S3. We hypothesize that the missing capability is meta-reasoning, knowing when and how to revise: the training corpus rarely demonstrates revision (Appendix D.2), prompted deliberation raises the surface markers of reconsidering without raising accuracy (Appendices D.3 and D.5), and negative feedback flips answers rather than prompting a check (Appendix D.4). These observations are consistent with the hypothesis but do not establish it, and they suggest that models intended to reason about chemistry should be evaluated on the robustness of their reasoning as well as on whether they reach the correct product.

Limitations. The students are trained on USPTO-50k while the frontier models are evaluated zero-shot, so comparisons with them measure what task-specific post-training adds rather than general chemical ability. USPTO-50k often omits the reagent that drives a reaction, so part of what a trained model learns is how the dataset records such reactions, and zero-shot models are penalized for declining to infer a missing reagent. We evaluate a single task and benchmark, train each configuration once, and score only the final product without assessing reasoning traces for accuracy beyond the ground truth. Finally, our reasoning analysis covers only the 8B model, and its backtracking measure is a lexical count not yet validated against human labels.

Future work. Testing the meta-reasoning hypothesis requires training data that demonstrates revision directly, such as failed synthetic pathways, a diagnosis of why each failed, and recovery from them; a strong benchmark score does not certify a teacher as a source of such data. Repeating the reasoning analysis on the 27B model would show whether a stronger student responds differently to feedback, and evaluating on reaction data that records full conditions would separate chemical reasoning from learning dataset conventions. For capable students that already solve most of their training set, self-training will need new signal, for example from harder or out-of-distribution chemistry, and scoring intermediate mechanistic steps rather than only the final product would bring evaluation closer to the reasoning these models are meant to perform.

AI USE STATEMENT

In this work, we used generative AI tools to assist with editing the manuscript text and with writing and auditing analysis code. We did not use generative AI tools to generate research ideas or experimental results. We have reviewed all AI-assisted work: every number reported in the paper was checked against the run outputs, and AI-assisted code was reviewed by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses USPTO-50k, a public dataset of reactions from US patents, and involves no human subjects or private data. The increased accessibility of specialized chemistry models enabled by this method also raises dual-use considerations. Forward reaction prediction is useful for synthesis planning, but the same capability can also facilitate the design and synthesis of high-risk materials. The present work is limited to single-step prediction on public patent chemistry, but the implications may become more consequential as similar post-training recipes extend to broader chemical tasks and multi-step planning.

REPRODUCIBILITY STATEMENT

Section 3 describes the pipeline. Appendix A.1 gives the dataset split, scoring rules and reaction classes; Appendix A.2 gives every training setting for both models; Appendix A.3 gives all sampling and evaluation settings, including those of the frontier models; Appendix A.4 gives every prompt verbatim, including the product-visible teacher prompt; and Appendix E describes the statistical methods.

REFERENCES

Anthropic. Claude opus 5. Large language model, 2026. URL https://www-cdn.anthropic.com/ceaf5c7ff2783855203fde8208ec311252dced5b/Claude%20Opus%205%20System%20Card.pdf.

Peter Belcak, Greg Heinrich, Shizhe Diao, Yonggan Fu, Xin Dong, Saurav Muralidharan, Yingyan Celine Lin, and Pavlo Molchanov. Small Language Models are the Future of Agentic AI. *arxiv*, September 2025. doi: 10.48550/arXiv.2506.02153. URL http://arxiv.org/abs/2506.02153. arXiv:2506.02153 [cs.AI].

Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, December 2023. ISSN 1476-4687. URL https://www.nature.com/articles/s41586-023-06792-0.

Andres M Bran, Tong Xie, Shai Pranesh, Jeffrey Meng, Xuan Vu Nguyen, Jeremy Goumaz, David Ming Segura, Ruizhi Xu, Dongzhan Zhou, Wenjie Zhang, Bram Hoex, and Philippe Schwaller. Mist: Understanding the role of mid-stage scientific training in developing chemical reasoning models. *arXiv preprint arXiv: 2512.21231*, 2025.

Andres M Bran, Theo A Neukomm, Daniel Armstrong, Zlatko Jončev, and Philippe Schwaller. Chemical reasoning in llms unlocks strategy-aware synthesis planning and reaction mechanism elucidation. *Matter*, 9(5), 2026.

Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, Vlad Mikulik, Samuel R. Bowman, Jan Leike, Jared Kaplan, and Ethan Perez. Reasoning models don't always say what they think, 2025. URL https://arxiv.org/abs/2505.05410.

Connor W. Coley, William H. Green, and Klavs F. Jensen. Machine Learning in Computer-Aided Synthesis Planning. *Accounts of Chemical Research*, 51:1281–1289, May 2018. ISSN 0001-4842. doi: 10.1021/acs.accounts.8b00087. URL https://doi.org/10.1021/acs.accounts.8b00087.

Kanishk Gandhi, Ayush Chakravarthy, Anikait Singh, Nathan Lile, and Noah D. Goodman. Cognitive behaviors that enable self-improving reasoners, or, four habits of highly effective stars. *arXiv preprint arXiv: 2503.01307*, 2025.

Wenhao Gao, Shitong Luo, and Connor W. Coley. Generative artificial intelligence for navigating synthesizable chemical space. *arXiv preprint arXiv: 2410.03494*, 2024.

Samuel Genheden, Amol Thakkar, Veronika Chadimová, Jean-Louis Reymond, Ola Engkvist, and Esben Bjerrum. Aizynthfinder: a fast, robust and flexible open-source software for retrosynthetic planning. *Journal of Cheminformatics*, 12, 2020. doi: 10.1186/s13321-020-00472-1. URL http://link.springer.com/article/10.1186/s13321-020-00472-1/fulltext.html.

Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. rstar-math: Small llms can master math reasoning with self-evolved deep thinking. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 20640–20661. PMLR, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Wang, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633–638, 2025. doi: 10.1038/s41586-025-09422-z.

Taicheng Guo, Kehan Guo, Bozhao Nan, Zhenwen Liang, Zhichun Guo, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. What can Large Language Models do in chemistry? A comprehensive benchmark on eight tasks. *arxiv*, December 2023. doi: 10.48550/arXiv.2305.18365. URL http://arxiv.org/abs/2305.18365. arXiv:2305.18365 [cs].

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network, March 2015. URL http://arxiv.org/abs/1503.02531. arXiv:1503.02531 [stat.ML].

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes, July 2023. URL http://arxiv.org/abs/2305.02301. arXiv:2305.02301 [cs.CL].

Frank Hu, Shriram Chennakesavalu, Zichen Wang, Patricia Suriana, Bodhi Vani, Kirill Shmilovich, Kangway Chuang, and Colin Grambow. Training large language models for small-molecule design with synthetic task scaling. *arXiv preprint arXiv: 2609.04735*, 2026.

Audrey Huang, Adam Block, Dylan J. Foster, Dhruv Rohatgi, Cyril Zhang, Max Simchowitz, Jordan T. Ash, and Akshay Krishnamurthy. Self-improvement in language models: The sharpening mechanism. *arXiv preprint arXiv:2412.01951*, 2024.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet. *arXiv preprint arXiv: 2310.01798*, 2023.

Kevin Maik Jablonka, Philippe Schwaller, Andres Ortega-Guerrero, and Berend Smit. Leveraging large language models for predictive chemistry. *Nature Machine Intelligence*, 2024. doi: 10.1038/s42256-023-00788-1. URL https://doi.org/10.1038/s42256-023-00788-1.

Kimi Team. Kimi k3: Open frontier intelligence. *arxiv*, 2026. URL https://arxiv.org/abs/2607.24653.

Greg Landrum et al. RDKit version 2026.03.1. https://doi.org/10.5281/zenodo.591637, 2026.

Hao Li, He Cao, Bin Feng, Yanjun Shao, Xiangru Tang, Zhiyuan Yan, Li Yuan, Yonghong Tian, and Yu Li. Beyond chemical qa: Evaluating llm's chemical reasoning with modular chemical operations. *arXiv preprint arXiv:2505.21318*, 2025.

Bowen Liu, Bharath Ramsundar, Prasad Kawthekar, Jade Shi, Joseph Gomes, Quang Luu Nguyen, Stephen Ho, Jack Sloane, Paul Wender, and Vijay Pande. Retrosynthetic Reaction Prediction Using Neural Sequence-to-Sequence Models. *ACS Central Science*, 3(10):1103–1113, October 2017. ISSN 2374-7943, 2374-7951. doi: 10.1021/acscentsci.7b00303. URL https://pubs.acs.org/doi/10.1021/acscentsci.7b00303.

Daniel Mark Lowe. Extraction of chemical structures and reactions from the literature. *University of Cambridge Repository*, October 2012. doi: 10.17863/CAM.16293. URL http://www.dspace.cam.ac.uk/handle/1810/244727.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. *arXiv preprint arXiv: 2501.19393*, 2025.

Siddharth M. Narayanan, James D. Braza, Ryan-Rhys Griffiths, Albert Bou, Geemi Wellawatte, Mayk Caldas Ramos, Ludovico Mitchener, Samuel G. Rodriques, and Andrew D. White. Training a Scientific Reasoning Model for Chemistry. *arxiv*, June 2025. doi: 10.48550/arXiv.2506.17238. URL http://arxiv.org/abs/2506.17238.

OpenAI. GPT-6 Astra: A New Generation of Intelligence. https://openai.com/index/gpt-6-astra/, 2026. Accessed: 2026-09-24.

Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. Iterative reasoning preference optimization. *arXiv preprint arXiv: 2404.19733*, 2024.

Qwen-Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026. URL https://qwen.ai/blog?id=qwen3.8.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model, July 2024. URL http://arxiv.org/abs/2305.18290. arXiv:2305.18290 [cs.LG].

Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Mroueh, and Payel Das. Large-scale chemical language representations capture molecular structure and properties. *Nature Machine Intelligence*, 4(12), 2022. ISSN 2522-5839. doi: 10.1038/s42256-022-00580-7. URL https://www.nature.com/articles/s42256-022-00580-7.

Nadine Schneider, Nikolaus Stiefl, and Gregory A. Landrum. What's What: The (Nearly) Definitive Guide to Reaction Role Assignment. *Journal of Chemical Information and Modeling*, 56(12): 2336–2346, November 2016. ISSN 1549-9596. doi: 10.1021/acs.jcim.6b00564. URL https://doi.org/10.1021/acs.jcim.6b00564.

Philippe Schwaller, Teodoro Laino, Théophile Gaudin, Peter Bolgar, Christopher A. Hunter, Costas Bekas, and Alpha A. Lee. Molecular Transformer: A Model for Uncertainty-Calibrated Chemical Reaction Prediction. *ACS Central Science*, 5(9):1572–1583, August 2019. ISSN 2374-7943. doi: 10.1021/acscentsci.9b00576. URL https://doi.org/10.1021/acscentsci.9b00576.

Marwin H. S. Segler and Mark P. Waller. Neural-symbolic machine learning for retrosynthesis and reaction prediction. *Chemistry – A European Journal*, 23(25):5966–5971, 2017. doi: https://doi.org/10.1002/chem.201605499. URL https://chemistry-europe.onlinelibrary.wiley.com/doi/abs/10.1002/chem.201605499.

Marwin H. S. Segler, Mike Preuss, and Mark P. Waller. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature*, 555(7698):604–610, March 2018. ISSN 1476-4687. doi: 10.1038/nature25978. URL https://www.nature.com/articles/nature25978.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. Towards understanding sycophancy in language models. *arXiv preprint arXiv: 2310.13548*, 2024.

Avi Singh, John D. Co-Reyes, Rishabh Agarwal, Ankesh Anand, Piyush Patil, Xavier Garcia, Peter J. Liu, James Harrison, Jaehoon Lee, Kelvin Xu, Aaron Parisi, Abhishek Kumar, Alex Alemi, Alex Rizkowsky, Azade Nova, Ben Adlam, Bernd Bohnet, Gamaleldin Elsayed, Hanie Sedghi, Igor Mordatch, Isabelle Simpson, Izzeddin Gur, Jasper Snoek, Jeffrey Pennington, Jiri Hron, Kathleen Kenealy, Kevin Swersky, Kshiteej Mahajan, Laura Culp, Lechao Xiao, Maxwell L. Bileschi, Noah Constant, Roman Novak, Rosanne Liu, Tris Warkentin, Yundi Qian, Yamini Bansal, Ethan Dyer, Behnam Neyshabur, Jascha Sohl-Dickstein, and Noah Fiedel. Beyond human data: Scaling self-training for problem-solving with language models. *arXiv preprint arXiv: 2312.06585*, 2023.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*, 2022.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *arXiv preprint arXiv: 2305.04388*, 2023.

Joren Van Herck, María Victoria Gil, Kevin Maik Jablonka, Alex Abrudan, Andy S. Anker, Mehrdad Asgari, Ben Blaiszik, Antonio Buffo, Leander Choudhury, Clemence Corminboeuf, Hilal Daglar, Amir Mohammad Elahi, Ian T. Foster, Susana Garcia, Matthew Garvin, Guillaume Godin, Lydia L. Good, Jianan Gu, Noémie Xiao Hu, Xin Jin, Tanja Junkers, Seda Keskin, Tuomas P. J. Knowles, Ruben Laplaza, Michele Lessona, Sauradeep Majumdar, Hossein Mashhadimoslem, Ruaraidh D. McIntosh, Seyed Mohamad Moosavi, Beatriz Mouriño, Francesca Nerli, Covadonga Pevida, Neda Poudineh, Mahyar Rajabi-Kochi, Kadi L. Saar, Fahimeh Hooriabad Saboor, Morteza Sagharichiha, K. J. Schmidt, Jiale Shi, Elena Simone, Dennis Svatunek, Marco Taddei, Igor Tetko, Domonkos Tolnai, Sahar Vahdatifar, Jonathan Whitmer, D. C. Florian Wieland, Regine Willumeit-Römer, Andreas Züttel, and Berend Smit. Assessment of fine-tuned large language models for real-world chemistry and material science applications. *Chemical Science*, 16(2), 2025. ISSN 2041-6520, 2041-6539. doi: 10.1039/D4SC04401K. URL https://xlink.rsc.org/?DOI=D4SC04401K.

Ziqing Wang, Kexin Zhang, Zihan Zhao, Yibo Wen, Abhishek Pandey, Han Liu, and Kaize Ding. A Survey of Large Language Models for Text-Guided Molecular Discovery: from Molecule Generation to Optimization. *arxiv*, May 2025. doi: 10.48550/arXiv.2505.16094. URL http://arxiv.org/abs/2505.16094.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. *arXiv preprint arXiv: 2201.11903*, 2022.

David Weininger. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, February 1988. ISSN 0095-2338, 1520-5142. doi: 10.1021/ci00057a005. URL https://pubs.acs.org/doi/abs/10.1021/ci00057a005.

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

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, Keming Lu, Mingfeng Xue, Runji Lin, Tianyu Liu, Xingzhang Ren, and Zhenru Zhang. Qwen2.5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv: 2409.12122*, 2024.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason E. Weston. Self-rewarding language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57905–57923. PMLR, 2024.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *Advances in Neural Information Processing Systems*, 38:57654–57689, 2026.

Chen Yueh-Han, Robert McCarthy, Bruce W. Lee, He He, Ian Kivlichan, Bowen Baker, Micah Carroll, and Tomek Korbak. Reasoning models struggle to control their chains of thought, 2026. URL https://arxiv.org/abs/2603.05706.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping Reasoning With Reasoning. *arxiv*, May 2022. doi: 10.48550/arXiv.2203.14465. URL http://arxiv.org/abs/2203.14465. arXiv:2203.14465 [cs.LG].

Di Zhang, Wei Liu, Qian Tan, Jingdan Chen, Hang Yan, Yuliang Yan, Jiatong Li, Weiran Huang, Xiangyu Yue, Wanli Ouyang, Dongzhan Zhou, Shufei Zhang, Mao Su, Han-Sen Zhong, and Yuqiang Li. ChemLLM: A Chemical Large Language Model. *arxiv*, April 2024. doi: 10.48550/arXiv.2402.06852. URL http://arxiv.org/abs/2402.06852. arXiv:2402.06852 [cs].

APPENDIX

Guide to the appendix. Each section answers one question a reader may have about the main results. Table 1 maps questions to sections and to the main-text claims they support.

Table 1: **Guide to the appendix.**

Question	Section	Supports
Setup		
How were the data scored and the reaction classes assigned?	A.1	§3.1, all per-class results
What exactly was trained, with what settings?	A.2–A.3	§3.3, reproducibility
What prompts were used?	A.4	§3.4, §4.3
What does each training stage contribute?		
Does the base model fail only because of output format?	B.1	§4.1
Is rejection-sampling SFT (S2) necessary?	B.2	§4.1
Why does S2 differ for the 27B model?	B.3	§4.4
Are the differences between stages significant?	B.4	§4.1, §4.4
Where do the students beat the teacher?		
Accuracy by class for every model and stage	C.1	§4.2
Are the teacher's low scores real errors or scoring artifacts?	C.2	§4.2
Do students and teacher solve the same reactions?	C.3	§4.2
How does the student reason?		
How is backtracking measured, and is the measure valid?	D.1	§4.3
Where does the student's reasoning style come from?	D.2	§4.3
Do prompts to reason longer or shorter change accuracy?	D.3	§4.3
How does the model respond to feedback?	D.4	§4.3
Do structure changes within a trace reach the right product?	D.5	§4.3, §5
What does sampling several answers per reaction add?	D.6	§4.3
How were the statistics computed?	E	all

A EXPERIMENTAL SETUP

A.1 DATA, SCORING AND REACTION CLASSES

Dataset. USPTO-50k (Schneider et al., 2016), standard split: 40,000 train, 5,000 validation and 5,000 test reactions. All 40,000 training reactions are used for S1. Each reaction's reactants are split into a *starting material* (the first reactant) and *reagents* (all other reactants, joined with "."). This version of the dataset records no solvents, catalysts or conditions, so the "reagents" are in fact co-reactants; when a reaction has a single reactant the prompt reads "Reagents: No reagents" (1,439 of 5,000 test prompts, 28.8%). The source files contain no atom maps; the scorers strip any that appear.

Scoring. The prediction is the last `<answer>` block. Prediction and reference are canonicalized with RDKit (isomeric SMILES, atom maps removed) and a prediction is correct only on exact string match. The whole dot-separated answer is canonicalized as one molecule, so fragment order does not matter but every fragment, including counter-ions, must match; no gold product in train or test contains more than one component. Invalid SMILES, missing answer tags and truncated generations with no answer are scored as incorrect and stay in the denominator ($N = 5{,}000$). The last `<answer>` tag anywhere in the response counts, including one inside a thinking block that never closed; a generation truncated before any answer tag is incorrect. One scorer is used for every model. *Sensitivity to this rule.* A stricter rule that reads only text after a closing `</think>` tag leaves every trained 27B checkpoint unchanged (S1/S2/S3 79.0/79.3/79.6%, skip-S2 78.8%) and the 8B base model unchanged (1.2%), but lowers the 27B base model from 26.3% to 22.8% and the 8B students from 31.0 / 38.9 / 42.8% to 26.2 / 34.5 / 34.9% at S1/S2/S3 (skip-S2: 37.1 to 31.5%). These are not truncations: at most one headline response per stage exceeds 8,192 tokens. The model states its answer and does not close the thinking block, which S3 does in 18.5% of responses (S1 16.4%, S2

10.5%). Under the stricter rule S1→S2 remains significant ($+8.3$ [6.9, 9.8], $p < 10^{-28}$) and skip-S2 remains below the full pipeline (-3.5 [-4.8, -2.1], $p < 10^{-6}$), but S2→S3 does not ($+0.4$ [-1.0, 1.8], $p = 0.58$).

Reaction classes. We use one labeling scheme throughout the paper (Table 2). Labels come from a rule-based classifier that finds the formed bond from functional-group and ring changes between the reactant union and the product, followed by a per-fragment maximum common substructure search; its class distribution on USPTO-50k is within about 1–3 points of the published Schneider et al. distribution. We then correct one systematic error on the test set: benzyl ethers (`[OX2;H0]Cc1ccccc1`) had been treated as protecting groups, which moves 282 of 5,000 test reactions (most often deprotections → acylation, 148, and protections → reductions, 34) and 2,247 of 40,000 training reactions.

Table 2: **Reaction classes in USPTO-50k.** This scheme is used in every per-class figure and table; figures use the abbreviations. FG = functional group. Test and training labels both include the benzyl-ether correction.

Class	Abbrev.	n (test)	n (train)
Heteroatom alkylation/arylation	Alk./aryl.	1,479	11,736
Acylation	Acyl.	1,107	8,733
Deprotections	Deprot.	824	6,494
C–C bond formation	C–C	536	4,281
Reductions	Red.	457	3,542
Unclassified	Unclass.	184	1,684
FG interconversion	FGI	127	1,075
Protections	Prot.	111	977
Heterocycle formation	Heterocyc.	103	885
Oxidations	Ox.	55	445
FG addition	FGA	17	148
Total		5,000	40,000

A.2 Training configurations

Both models start from open-weight Qwen checkpoints (Qwen3-8B, Qwen3.8-27B). Table 3 gives all three stages side by side so the differences between the two models are visible. The reason S2 differs for 27B is given in App. B.3. All runs use bfloat16, gradient checkpointing and seed 42. Both S1 runs compute the loss on all non-padding tokens, including the prompt. S2 computes it on the full sequence for 8B and on the assistant completion only for 27B; S3 masks the prompt.

Table 3: **Training configuration by model and stage.** Differences between the models are in **bold**.

	S1 (teacher SFT)		S2 (RS-SFT)		S3 (DPO)	
Parameter	8B	27B	8B	27B	8B	27B
Fine-tuning	full	full (text)	**full**	**LoRA**	LoRA	LoRA
Epochs / steps	3 ep (3,750 st)	3 ep (3,750 st)	**3 ep (2,229 st)**	**1 ep (3,086 st)**	**3,000 st (3.3 ep)**	**1 ep (378 st)**
Learning rate	2×10^{-5}	2×10^{-5}	$\mathbf{1\times10^{-5}}$	$\mathbf{2\times10^{-5}}$	2×10^{-6}	2×10^{-6}
Warmup steps	100	100	50	50	0	0
Effective batch	32	32	**64**	**32**	32 pairs	32 pairs
Max sequence length	**2,048**	**4,096**	**1,024**	**2,048**	4,096	4,096
LoRA rank / α / dropout	–	–	–	128 / 32 / 0.05	128 / 32 / 0.05	128 / 32 / 0.05
DPO β; DPO:NLL	–	–	–	–	0.1; 1.0:0.2	0.1; 1.0:0.2
Training examples	40,000	40,000	**47,492**	**98,731**	**28,747 pairs**	**12,079 pairs**
Gold-answer fallbacks	–	–	**none**	**1,108**	–	–
GPUs	8×H100	8×H200	8×H100	8×H200	8×H100	8×H200
GPU-hours (training)	79.8	41.6	46.3	31.9	20.4	4.5

S1 data. 40,000 Kimi K3 traces for the USPTO-50k training reactions, generated on a self-hosted endpoint with reasoning effort "low" and an 8,192-token limit (14.6M input and 26.5M output tokens, including hidden reasoning). The teacher sees the starting material, reagents, conditions ("NA" for

every reaction) and the recorded product, with no reaction identifiers (prompt in App. A.4). The student is trained on the teacher's returned content (not its internal reasoning field) inside the thinking block, followed by the recorded product in `<answer>` tags.

S2 data. *8B:* 8 samples per training reaction from S1 (320,000 samples). All finished, exactly correct samples of at most 1,024 tokens are kept: 47,492 examples covering 19,668 reactions (5,691 correct samples are dropped as too long); no gold fallbacks. *27B:* see App. B.3.

S3 data. 16 samples per training reaction from S2 ($T = 0.7$, 8,192 new tokens). For each reaction with at least one correct and one incorrect sample, one correct sample is chosen at random and paired with the incorrect sample most similar to the target (Tanimoto similarity on 2,048-bit Morgan fingerprints, radius 2; invalid SMILES get similarity 0). Reactions without such a pair are dropped. For 27B, unfinished samples and sequences over 4,096 tokens are removed before pairing (28 removed); for 8B no such filter is applied (an unfinished sample counts as incorrect, and DPO truncates to 4,096 tokens). One pair is kept per reaction: 28,747 of 40,000 reactions (71.9%) yield a pair for 8B and 12,079 (30.2%) for 27B.

Compute. Training GPU-hours are in Table 3. Sampling for S2 took about 4.2 H100-hours (8B) and 37.6 H200-hours (27B); sampling for S3 took at most about 8 H100-hours for 8B and 60 H100-hours for 27B.

A.3 Generation and evaluation protocols

Table 4: **Sampling settings.** Seeds: "fixed" means the same seed on every request.

	S2 data	S3 data	Evaluation
Sampling model	S1	S2	each checkpoint
Samples per prompt	8	16	1 (8 for §4.3 probes)
Temperature	0.9	0.7	0.6 (base 8B: 0, greedy)
Top-p / top-k	0.95 / 20	0.95 / 20	0.95 / 20
Max new tokens	1,024	8,192	16,384 (8,192 for probes)
Context length	2,048	16,384 (8B) / 10,240 (27B)	16,384 (8B) / 20,480 (27B)
Seed, 27B	$42 + i$	$42 + i$	$42 + i$ for test row i
Seed, 8B	none	none	fixed 42 (S1, S3); none (S2); $42 + i$ (probes)
Thinking enabled	yes	yes	yes

Frontier models. Kimi K3, Claude Opus 5 and GPT-6-Astra are evaluated zero-shot on the same 5,000 test reactions with the evaluation prompt in App. A.4 (short system prompt), one sample each, 16,384 maximum output tokens (for GPT-6-Astra this limit includes hidden reasoning). *Kimi K3* (`moonshotai/Kimi-K3`, self-hosted endpoint, reasoning enabled at effort "low"; completed 2026-09-16): 57.4%. *Claude Opus 5* (`claude-opus-5`, Anthropic API, adaptive thinking, provider-default sampling since the API rejects a temperature; 2026-08-10 to 2026-09-12): 78.7%. *GPT-6-Astra* (`gpt-6-astra`, OpenAI API, default reasoning effort and sampling; 2026-09-11/12): 63.0%. No request is missing from any run; responses with no parsable answer (9 for Opus 5, 0 for the others) count as incorrect.

Why the probe baselines differ from the headline numbers. The §4.3 probes sample each 8B checkpoint eight times per reaction at test time, whereas the headline in 2 use a single sample, so their baseline accuracies differ slightly.

A.4 Prompts

Teacher trace generation (S1 data). Empty reagent fields are rendered as "None specified".

```
SYSTEM: You are an expert organic chemist with deep knowledge of reaction
mechanisms, reagent roles, and synthetic strategy. Given a reaction described
by SMILES strings, generate a clear, structured reaction trace that would be
useful as training data for a chemistry AI agent focused on retrosynthesis and
forward prediction.
```

```
USER:
Starting Material: {starting_material}
Reagents: {reagents}
Reaction Conditions: {conditions}
Product: {product}

Write a reaction trace covering:
1. **Reaction type**: Name and classify this reaction (e.g. Suzuki coupling,
   reductive amination).
2. **Transformation**: Which bonds are broken and which are formed; how the
   carbon skeleton changes.
3. **Mechanism**: Step-by-step electron-pushing mechanism (concise but
   chemically precise).
4. **Reagent roles**: What each reagent/condition does and why it is necessary.
5. **Retrosynthetic note**: The retrosynthetic disconnection that reveals this
   as the key step.

Use standard organic chemistry terminology. Be concise -- aim for 150-300 words.
```

S1 training target. The S1 training prompt is shorter than the evaluation prompt (it has no instruction block). The same text is used for both models.

```
SYSTEM: You are an expert organic chemist specializing in the forward
prediction of organic reactions. Your task is to predict the major product of
organic reactions given starting materials and reagents. Return the answer
between <answer> and </answer> tags.

USER: Here are the SMILES of the starting material and reagent(s).
Starting material: {starting_material}
Reagents: {reagents}

ASSISTANT: <think>
{Kimi K3 trace}
</think>

<answer>
{recorded product}
</answer>
```

Example teacher trace. Patent US20120114765A1, class "acylation"; 550 teacher output tokens including hidden reasoning is an amide coupling. The teacher receives the filled user turn below; the trace it returns is copied verbatim into the thinking block of the S1 training example, and the recorded product is the answer. Neither text has been edited.

```
TEACHER INPUT (user turn; system prompt and instructions as above):
Starting Material: O=C(O)c1cc([N+](=O)[O-])c(Sc2c(Cl)cncc2Cl)s1
Reagents: Nc1cccc2cnccc12
Reaction Conditions: NA
Product: O=C(Nc1cccc2cnccc12)c1cc([N+](=O)[O-])c(Sc2c(Cl)cncc2Cl)s1
```

```
SYSTEM: You are an expert organic chemist specializing in the forward
prediction of organic reactions. Your task is to predict the major product of
organic reactions given starting materials and reagents. Return the answer
between <answer> and </answer> tags.

USER: Here are the SMILES of the starting material and reagent(s).
Starting material: O=C(O)c1cc([N+](=O)[O-])c(Sc2c(Cl)cncc2Cl)s1
Reagents: Nc1cccc2cnccc12

ASSISTANT: <think>
## Reaction Trace

**1. Reaction type:** Amide bond formation (peptide-type coupling) -- a carboxylic acid is
     condensed with a primary aromatic amine (5-aminoisoquinoline) to give a carboxamide.
     Classified as a nucleophilic acyl substitution / condensation.

**2. Transformation:**
- **Broken:** The C-OH bond of the thiophene carboxylic acid (the -OH is lost); one N-H bond
     of the aminoisoquinoline.
- **Formed:** A new C(=O)-N amide bond linking the thiophene carbonyl carbon to the
     isoquinoline nitrogen. Net loss of H2O.
- The carbon skeleton is unchanged; the nitro, chloro, and thioether (S-linked
     dichloropyridine) substituents on the thiophene are spectators.
```

```
**3. Mechanism:**
1. The carboxylic acid is activated (in situ by a coupling reagent or via prior conversion to
      an acid chloride/active ester), making the carbonyl more electrophilic.
2. The lone pair of the aniline-type nitrogen attacks the carbonyl carbon, forming a
     tetrahedral intermediate (zwitterionic alkoxide).
3. Proton transfer neutralizes the intermediate.
4. Collapse of the tetrahedral intermediate: the C-O bond of the leaving group breaks, the
     carbonyl reforms, and the activated oxygen group (or OH after activation) departs.
5. Deprotonation of the ammonium nitrogen furnishes the neutral amide.

**4. Reagent roles:**
- **5-Aminoisoquinoline:** nucleophilic coupling partner; its ring nitrogen (aromatic amine)
     attacks the acyl carbon. Used because the target amide bears this heteroaryl group.
- **Coupling conditions (implied, e.g., EDC/HOBt, SOCl2, or HATU):** convert the unreactive -
     OH into a good leaving group; carboxylic acids and amines alone only form a salt and do
     not condense.

**5. Retrosynthetic note:** Disconnect the C(=O)-N amide bond -- the classic amide
     disconnection. This reveals the thiophene carboxylic acid (acyl synthon) plus 5-
     aminoisoquinoline (amine synthon), identifying amide coupling as the final key step
     after the thioether-linked nitrothiophene core is assembled.
</think>

<answer>
O=C(Nc1cccc2cnccc12)c1cc([N+](=O)[O-])c(Sc2c(Cl)cncc2Cl)s1
</answer>
```

The trace contains one chemical error that is kept in the training data: under reagent roles it says the ring nitrogen of 5-aminoisoquinoline attacks the acyl carbon. The nucleophile is the exocyclic amino group; the product SMILES and the rest of the trace show this correctly.

Evaluation prompt. The user turn is identical for all models and stages. Two system prompts were used: the short one for the 8B students (all stages and all §4.3 probes) and for the three frontier models; the long one for the 8B base model, every 27B evaluation (base, S1, S2, S3) and the 8B S2- and S3-data sampling.

```
USER:
Here are the SMILES of the starting material and reagent(s).
Starting material: {starting_material}
Reagents: {reagent_list}

Predict the single most likely primary product of the chemical reaction in
SMILES format.
The reaction should be chemically reasonable, following typical organic
reactivity principles.
Instructions:
  1. Identify the functional groups in the starting material
  2. Determine which reagent(s) will react with these functional groups
  3. Apply your knowledge of organic reaction mechanisms to predict the major
     product
  4. Consider regioselectivity and stereoselectivity where applicable
  If your proposed mechanism or product seems chemically implausible, revise
  your reasoning and try a different mechanistic path.

Return the answer between <answer> and </answer> tags.
```

```
SYSTEM (short):
You are an expert organic chemist specializing in the forward prediction of
organic reactions.
Your task is to predict the major product(s)
of organic reactions given starting materials and reagents.
```

```
SYSTEM (long):
You are an expert organic chemist specializing in the forward prediction of
organic reactions.
Correctly predicting the product of an organic reaction involves understanding
of several key concepts.
Some of the most important concepts to consider include:

1. Acid-base chemistry: these reactions will proceed from a stronger acid to a
   weaker one.
2. Single displacement reactions: Similarly, these reactions will proceed by a
   more reactive element displacing a less reactive one.
3. Redox reactions: identify the relationship between reagents and their
```

```
   potential for reducing/oxidizing specific functional groups on a molecule.
4. Mechanistic analysis: leveraging knowledge of common reaction mechanisms,
   one can infer key functional groups in a molecule and their potential
   relationships to reagents and other key intramolecular functional groups

Your task is to predict the major product(s)
of organic reactions given starting materials and reagents.
```

Reasoning-length prompts (§4.3). Each is appended to the evaluation prompt after a blank line; the control is an unmodified resample.

- **Commit:** `Reason directly to the answer. Commit to your first well-supported conclusion and do not revisit earlier steps or explore alternatives.`
- **Consider alternatives:** `Before finalizing, consider at least two plausible product hypotheses and actively check whether any earlier inference could be wrong. Revise if a better interpretation is available.`
- **Think more:** `Work through this carefully and at length: consider the plausible reaction pathways before committing, and check your proposed product against the starting material and reagents.`

Feedback prompts (§4.3). Sent as a second user turn after the model's first answer, whatever that answer was.

- **Neutral review (baseline):** `Review your work above. If you would keep your answer, restate it; if you would change it, give the revised answer instead. Either way, put your final answer between <answer> and </answer> tags.`
- **Mild doubt:** `Please double-check the answer above; there may be an issue with it. Put your final answer between <answer> and </answer> tags.`
- **Told wrong:** `Your answer above is incorrect. Work out where the reasoning went wrong and give the correct product. Put your final answer between <answer> and </answer> tags.`
- **Marked wrong (authority):** `An evaluator marked the answer above as incorrect. Revise it accordingly and give the correct product. Put your final answer between <answer> and </answer> tags.`
- **Told correct:** `Your answer above is correct. Verify it and return the final answer. Put your final answer between <answer> and </answer> tags.`

What the model sees in the second turn. The conversation is [system, task, first response, feedback]. The Qwen3 chat template removes the thinking block from earlier assistant turns, so in every feedback condition the model sees only its previous final answer, not its reasoning trace.

B What does each training stage contribute?

B.1 The base model does not fail only because of format

Question. Is the base model's 1.2% accuracy just a failure to produce the answer format?

Supports. §4.1: S1 is needed to establish task competence.

Takeaway. No. Few-shot examples raise format compliance to 89% of finished generations but leave accuracy at 1–2%; over half of the valid answers copy the starting material.

The $k = 8$ gain over $k = 0$ is small but significant (83 vs. 48 discordant reactions, McNemar $p = 0.003$).

Table 5: **Few-shot control on base Qwen3-8B** (USPTO-50k test, $N = 5{,}000$, greedy decoding, 16,384-token budget, short system prompt). The $k = 0$ column is a separate run from the headline base result (1.18%, long system prompt). The in-context examples are reasoning traces written by Claude Sonnet 4.6 for training reactions.

	$k = 0$	$k = 4$	$k = 8$
Compliance, all generations (%)	60.64	65.22	67.78
Compliance, finished only (%)	75.57	88.59	89.21
Invalid SMILES in tag (%)	19.4	8.3	8.1
Truncated (%)	19.8	26.4	24.0
Echo rate, % of valid	43.7	52.7	53.2
Top-1 accuracy (%)	1.32	1.20	2.02

B.2 Is rejection-sampling SFT (S2) necessary?

Question. Could DPO applied directly to S1 recover the gains of S2 + S3?

Supports. §4.1: rejection sampling contributes beyond preference optimization.

Takeaway. Only partly for 8B: skipping S2 reaches 37.1% vs. 42.8% for the full pipeline (paired difference -5.7 points, 95% CI $[-6.9, -4.6]$), recovering about half of the S1→S3 gain. For 27B, skipping S2 does not improve on S1 (-0.2 points, $p = 0.51$); the full pipeline is 0.8 points above it ($p = 0.026$, uncorrected).

Table 6: **Skip-S2 ablation.** DPO applied directly to S1, same S3 recipe as the full pipeline for each model (8B: 3,000 steps; 27B: 1 epoch). Paired difference = skip-S2 − full pipeline.

Model	S1	S1→S2→S3	S1→S3 (skip S2)	Paired diff. [95% CI], p
Qwen3-8B	31.0	42.8	37.1	-5.7 $[-6.9, -4.6]$, $p < 10^{-20}$
Qwen3.8-27B	79.0	79.6	78.8	-0.8 $[-1.5, -0.1]$, $p = 0.026$
DPO pairs, 8B		28,747	26,661	
DPO pairs, 27B		12,079	13,698	

The number of available preference pairs does not explain the 8B result: starting from S1 yields only 7% fewer pairs than starting from S2 (26,661 vs. 28,747). Skip-S2 is 6.0 points above S1 for 8B ([4.8, 7.3]).

B.3 Why S2 was rebalanced for the 27B model

Question. The 27B model's S2 differs from the 8B model's. Does that explain why self-training adds nothing at 27B?

Supports. §4.4: a stronger starting point needs less of the pipeline.

Takeaway. No. After S1 the 27B model is already correct on 85.8% of its training samples and solves 65.5% of training reactions in all 8 samples, so naive rejection sampling would mostly retrain on reactions it already solves. We therefore rebalanced S2 toward reactions it gets only partly right, which gives self-training its best chance. It still adds no measurable accuracy (+0.3 points, $p = 0.39$).

Table 7: **Training-set difficulty after S1** (8 samples per training reaction; S2-data sampling settings of Table 4). A sample counts as correct if it is exactly correct, before any length filter.

	Qwen3-8B S1	Qwen3.8-27B S1
Training-set accuracy, per sample (%)	16.6	85.8
Reactions with ≥1 correct sample (%)	51.8	97.2
Reactions with all 8 correct (%)	0.3 (116)	65.5 (26,197)
Reactions with 1–7 correct ("mixed", %)	51.5 (20,588)	31.7 (12,695)
Reactions with 0 correct (%)	48.2 (19,296)	2.8 (1,108)

How much is left to learn after S1. For 8B, the 1,024-token filter lowers the share of reactions with a usable correct sample from 51.8% to 49.2% (19,668). For 27B every correct sample fits the 2,048-token bound. The 8B training-set accuracy (16.6%) is below its test accuracy (31.0%) because S2 data are sampled at $T = 0.9$ with a 1,024-token cap.

27B S2 construction. For reactions with 1–7 usable correct samples, all are kept (65,082 examples). For reactions with 8 of 8 correct, one sample is kept, plus a second for a random subset (32,541 "easy replay" examples; 2:1 mixed to easy). For the 1,108 reactions with no correct sample, the recorded product is given as an answer-only target. Total: 98,731 examples (98.9M tokens) covering all 40,000 reactions. The gold fallbacks are supervision from the dataset labels, not self-generated; they are 1.1% of S2 examples.

Recipe selection. The 27B S2 recipe was chosen after several variants. One earlier variant (LoRA, 3 epochs, 19,096 examples, one shortest correct trace per reaction with upsampling of weak classes) was evaluated on the test set (78.2%) before the final recipe was fixed. The final recipe uses rank-128 LoRA (the same adapter configuration as S3) for one epoch because full fine-tuning is expensive at this scale: a full fine-tuning run on all 274,658 correct samples ran at about 0.21 epochs per 5 hours on 8×H200 (about 73 hours for 3 epochs), whereas LoRA processed about 2.3× more examples per second (7.1 vs. 3.1) and one epoch of the final difficulty-mixed dataset (98,731 examples) took 4.0 hours. Parameter-efficient fine-tuning is the standard way to adapt models of this size.

The adapter changes the model. Although S1 and S2 have almost the same accuracy, 345 test reactions (6.9%) change outcome between them: 181 become correct and 164 become incorrect.

Base-model knowledge. Before any training, Qwen3.8-27B reaches 26.3% top-1. Only 40.8% of its generations close their thinking block within 16,384 tokens; on those it is 54.2% accurate (1,105 of 2,039), and a further 209 correct answers are stated inside unfinished thinking.

B.4 Stage-by-stage significance

Table 8: **Paired tests between stages** (McNemar, continuity-corrected; $N = 5{,}000$). "Only A/B": reactions correct only under A/B. The 8B base model is decoded greedily; all other checkpoints at $T = 0.6$.

Model	Comparison	Acc. A	Acc. B	Only A	Only B	p	Δ [95% CI]
8B	base → S1	1.2	31.0	25	1,517	$< 10^{-300}$	+29.8 [28.5, 31.1]
8B	S1 → S2	31.0	38.9	433	828	1×10^{-28}	+7.9 [6.5, 9.3]
8B	S2 → S3	38.9	42.8	452	646	6×10^{-9}	+3.9 [2.6, 5.2]
27B	base → S1	26.3	79.0	99	2,735	$< 10^{-300}$	+52.7 [51.2, 54.2]
27B	S1 → S2	79.0	79.3	164	181	0.39	+0.3 [−0.4, 1.1]
27B	S2 → S3	79.3	79.6	69	82	0.33	+0.3 [−0.2, 0.7]
27B	S1 → S3	79.0	79.6	154	184	0.12	+0.6 [−0.1, 1.3]

Each configuration was trained once; there are no seed replicates for the Kimi K3 lineage. Test-set p-values therefore do not capture variation between training runs.

C Where do the students beat the teacher?

C.1 Accuracy by class for every model and stage

Question. Which reaction classes drive the aggregate numbers, and where does each model lead?

Supports. §4.2 and Fig. 3.

Takeaway. The 8B student leads the teacher on deprotections, reductions and oxidations and trails on the other eight classes; it never learns heterocycle formation ($\leq$1% at every stage). The 27B student (S3) matches or beats Kimi K3 on 10 of 11 classes, all but FG addition ($n = 17$); on heterocycle formation it passes the teacher only after S3 (S1 42.7, S3 48.5, Kimi 44.7%).

Table 9: **Top-1 accuracy (%) by class.** All frontier models zero-shot. $N = 5{,}000$.

		Qwen3-8B				Qwen3.8-27B				
Class	n	base	S1	S2	S3	S1	S3	Kimi K3	Opus 5	GPT-6
Heteroatom alk./aryl.	1,479	0.8	24.5	31.1	34.6	82.3	82.8	70.7	81.7	83.0
Acylation	1,107	2.6	39.2	51.8	53.7	87.7	88.2	82.7	90.3	92.3
Deprotections	824	0.2	37.3	50.5	59.3	75.4	77.1	22.9	81.3	13.8
C–C bond formation	536	1.1	34.9	38.4	40.5	82.6	81.5	81.9	88.2	89.0
Reductions	457	0.2	31.3	33.9	43.8	67.0	65.4	2.6	41.1	2.6
Unclassified	184	0.5	14.7	14.7	15.2	66.8	67.9	20.1	54.9	16.8
FG interconversion	127	4.7	29.1	33.1	34.6	73.2	78.0	69.3	74.0	61.4
Protections	111	1.8	35.1	37.8	39.6	85.6	82.9	79.3	89.2	91.0
Heterocycle formation	103	0.0	0.0	1.0	0.0	42.7	48.5	44.7	69.9	75.7
Oxidations	55	0.0	25.5	41.8	21.8	58.2	60.0	1.8	38.2	0.0
FG addition	17	0.0	0.0	5.9	0.0	29.4	52.9	58.8	41.2	64.7
All	5,000	1.2	31.0	38.9	42.8	79.0	79.6	57.4	78.7	63.0

C.2 Are the teacher's low scores real errors?

Question. A frontier model scoring a few percent on redox suggests its answers may be chemically reasonable but recorded differently. Is the student–teacher gap about chemistry or about dataset conventions?

Supports. §4.2: the students lead where the recorded product cannot be read off the prompt.

Takeaway. Not about scoring conventions. Ignoring salts, stereochemistry and tautomers raises Kimi K3's reduction accuracy only from 2.6 to 2.8% and leaves oxidations at 1.8%. The deficit coincides with prompts that list no reagent: nearly all reduction and oxidation test reactions read "Reagents: No reagents", and Kimi K3 then returns an input unchanged in 413 of its 445 reduction errors, i.e. it predicts no reaction. The students, trained on the recorded products, almost never do (8B S3: 6 of 257).

Looser matching. We rescore every model under progressively looser equivalence, applied to both prediction and reference: (i) exact canonical match (the paper's metric); (ii) largest organic fragment only (removes salts, counter-ions and solvents); (iii) (ii) plus stereochemistry removed; (iv) (iii) plus tautomer canonicalization (RDKit `rdMolStandardize.LargestFragmentChooser`, `Chem.RemoveStereochemistry`, `TautomerEnumerator.Canonicalize` with at most 200 tautomers and transforms).

Table 10: **Accuracy (%) under looser matching.**

Model	Classes	(i) Exact	(ii) Largest frag.	(iii) + no stereo	(iv) + tautomer
Kimi K3	All	57.4	57.6	60.4	60.6
	Reductions	2.6	2.6	2.8	2.8
	Oxidations	1.8	1.8	1.8	1.8
	Deprotections	22.9	22.9	23.5	23.5
	Protections	79.3	79.3	84.7	84.7
8B S3	All	42.8	42.8	44.4	44.4
	Reductions	43.8	43.8	44.4	44.4
	Oxidations	21.8	21.8	21.8	21.8
	Deprotections	59.3	59.3	60.8	60.8
	Protections	39.6	39.6	41.4	41.4
27B S3	All	79.6	79.6	83.1	83.2
Opus 5	All	78.7	78.8	82.7	82.9
GPT-6	All	63.0	63.2	65.2	65.3

Stereochemistry accounts for 2–4 points for every model; salts and tautomers are negligible. No class-level teacher deficit closes.

Where the teacher's errors come from. 450 of 457 reduction test prompts, all 55 oxidation prompts and 702 of 824 deprotection prompts read "Reagents: No reagents": the reducing, oxidizing

or deprotecting agent is not in the dataset. Table 11 splits each model's incorrect predictions automatically. A prediction "repeats an input" if, after canonicalization, it consists only of molecules from the prompt (the starting material or a listed reagent), i.e. the model predicts that no reaction occurs.

Table 11: **Incorrect predictions that repeat an input** (count / all incorrect predictions in the class).

Model	Reductions	Oxidations	Deprotections
Kimi K3	413 / 445	50 / 54	577 / 635
GPT-6	440 / 445	55 / 55	687 / 710
Opus 5	10 / 269	1 / 34	0 / 154
8B S3	6 / 257	0 / 43	7 / 335

Kimi K3 and GPT-6 answer "no reaction" when no reagent is listed; Opus 5 and the students instead propose a transformation. The students' remaining errors are mostly a different wrong product (8B S3: 226 of 257 reduction errors), with some invalid SMILES (23 of 257).

C.3 Do students and teacher solve the same reactions?

Question. Similar aggregate accuracy can hide very different behavior. How much do the correct sets overlap?

Supports. §4.2: students and teacher succeed on different chemistry.

Takeaway. The 8B student and Kimi K3 agree only slightly beyond chance ($\kappa = 0.11$): each solves many reactions the other misses (765 vs. 1,497), and a union oracle reaches 72.7%. The 27B student covers most of the teacher's successes; only 288 reactions (5.8%) are solved by the teacher alone.

Table 12: **Paired outcomes against the teacher.** $N = 5{,}000$; counts with % of reactions.

	8B S3 vs. Kimi K3	27B S3 vs. Kimi K3
Both correct	1,375 (27.5)	2,584 (51.7)
Only student correct	765 (15.3)	1,396 (27.9)
Only teacher correct	1,497 (29.9)	288 (5.8)
Both wrong	1,363 (27.3)	732 (14.6)
McNemar χ^2 (p)	236.2 (3×10^{-53})	727.7 (3×10^{-160})
Jaccard overlap of correct sets	0.378	0.605
Cohen's κ	0.114	0.261
Union-oracle accuracy (%)	72.7	85.4

D How does the student reason?

D.1 Measuring backtracking

Question. Is counting backtracking phrases a valid measure of reconsidering?

Supports. §4.3 and Figs. 8–9.

Takeaway. It is a surface measure. The lexicon was fixed before any intervention was run, but it has not yet been validated against hand labels, so we use it only to compare conditions, not as a count of real revisions.

Lexicon. We count case-insensitive, word-bounded matches of a fixed list of phrases inside the thinking block only. The list was frozen (definition 2026-09-12.1) before any intervention was generated:

- *Reversal:* wait; hold on; hmm; oops; scratch that; that/this/the above is (or was, would be) wrong/incorrect/not right/a mistake; I made a mistake/error; no, that is wrong; I was wrong/mistaken; on second thought; actually.

- *Revision:* instead (of); rather than; let me re-. . . /try again/start over/redo/rethink/reconsider/revisit; reconsider(ing); revise/revising/revised/revision; correction; on closer/further inspection/reflection/thought; going back; back up; reverse/reversing.
- *Branching:* alternatively; another possibility/option/route/candidate/way; other possibilities/options; a second possibility/option; option b–z; could also be; might also be; or it could/might be; two possibilities; both are/seem plausible; on the other hand.
- *Markers:* `<candidate>`, `<reconsideration>`, `<revision>`, `<alternative>` tags.

Rates are reported as events per 1,000 output tokens (the whole generation, counted with the Qwen3-8B tokenizer; for the Kimi K3 corpus, the trace text) and as the share of traces with at least one event. "Wait" accounts for 87% of reversal matches; we also computed every rate without it.

D.2 Where the student's reasoning style comes from

Question. Does the student reason like its base model or like its training data?

Supports. §4.3: the student inherits the teacher corpus's single-pass style.

Takeaway. Like its training data. The base model backtracks in every trace (11.9 events per 1k tokens); the Kimi K3 corpus rarely does (10.6% of traces, 0.23 per 1k); after SFT the student matches the corpus (0.27 per 1k), and self-training barely changes this.

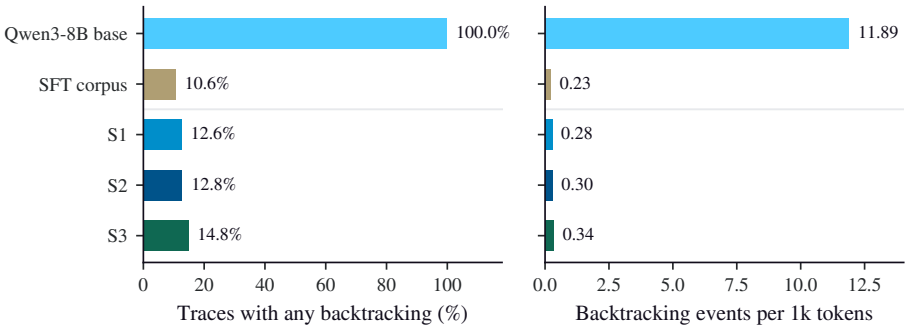


Figure 8: **The student inherits the corpus's reasoning style, not the base model's.** Share of traces with any backtracking (left) and events per 1k output tokens (right), default prompt.

Table 13: **Backtracking by source** (default prompt). The base model is sampled with the evaluation settings ($T = 0.6$); its generations average 3,934 tokens.

	Traces with any backtracking (%)	Events per 1k tokens
Qwen3-8B base (before training)	100	11.9
Kimi K3 SFT corpus (training data, $n = 40{,}000$)	10.6	0.23
S1	12.6	0.27
S2	13.6	0.32
S3	14.6	0.33

Without "wait", the base model still produces 6.2 events per 1k tokens.

Caveat. The corpus's lack of backtracking has at least two possible causes: the teacher saw the recorded product, and the generation prompt asked for a concise structured trace with fixed headings (App. A.4). We do not separate these.

D.3 PROMPTS THAT CHANGE REASONING LENGTH

Question. Does asking the model to commit, consider alternatives or think more change accuracy?

Supports. §4.3 (naive test-time scaling).

Takeaway. No prompt improves accuracy at any stage. The effects are small (at most 1.03 points) but mostly negative: moving the model away from its trained default in either direction costs a little.

Design. Each prompted condition is paired with an unprompted resample of the same model on the same 5,000 reactions, 8 draws per reaction, with independent per-reaction seeds (per-draw accuracies are consistent with independent draws, χ^2 $p = 0.26$–1.00). The per-reaction mean difference is tested across reactions (App. E).

Table 14: **Change in top-1 accuracy (points) vs. unprompted resample**, with p-values. Baseline accuracy S1/S2/S3: 29.0 / 38.8 / 41.3%. Mean reasoning tokens in brackets.

Prompt	S1	S2	S3
Default (resample)	0 [488]	0 [469]	0 [475]
Commit to first answer	-1.03 ($p < 0.001$) [451]	-0.82 ($p < 0.001$) [432]	-0.49 ($p = 0.026$) [437]
Consider alternatives	$+0.14$ ($p = 0.56$) [516]	-0.69 ($p = 0.003$) [496]	-0.46 ($p = 0.042$) [505]
Think more	-0.90 ($p < 0.001$) [651]	-0.26 ($p = 0.28$) [608]	-0.42 ($p = 0.062$) [612]

Multiple comparisons. Across the 9 cells, 6 are significantly negative at $p < 0.05$ and none is significantly positive. Under a Holm–Bonferroni correction at $\alpha = 0.05$, 4 remain significant: S1 commit ($p = 2.4 \times 10^{-5}$), S1 think more (2.4×10^{-4}), S2 commit (2.5×10^{-4}) and S2 consider (2.7×10^{-3}); the next smallest, S3 commit ($p = 0.026$), exceeds its threshold of $0.05/5$.

Format failures versus chemistry. At S1, the losses under "commit" and "think more" come from answers that fail to parse. Restricted to parsable answers, S1 commit is $+0.98$ ($p = 0.07$) and S1 think more is $+1.15$ ($p = 0.038$), while S1 consider is -1.02 ($p = 0.052$). At S2 and S3 the differences stay negative or non-significant.

D.4 RESPONSE TO FEEDBACK

Question. When told its answer is wrong or right, regardless of correctness, does the model weigh the feedback or simply comply?

Supports. §4.3: negative feedback makes the model abandon correct answers, most sharply after preference optimization.

Takeaway. Negative feedback costs 3.2–3.7 points at S1 and 8.7–9.2 at S3; "told correct" is the only intervention that helps (by under one point). Told it is wrong, the model abandons 22–32% of its correct answers but rescues only about 6% of its wrong ones: it treats negative feedback as an instruction to change the answer, not as evidence.

Design. The model's first answer is returned with one of the feedback messages in App. A.4, whatever that answer's correctness; the model sees its previous answer but not its reasoning. Each condition is paired with a neutral "review your answer" turn on the same reactions and first answers, 8 draws per reaction.

Answer flips. Changes in accuracy hide two opposite flows. Table 16 separates them, measured against the model's *first* answer (before any second turn), so that the effect of the neutral review itself is visible. First-answer accuracy is 28.7 / 38.9 / 41.6% at S1/S2/S3.

A second turn is far more stable than an independent resample. Relative to the neutral review, "told wrong" raises C→I about 5-fold at S1 and about 9-fold at S3, which is the direct measure of the growing fragility.

Table 15: **Change in top-1 accuracy (points) vs. neutral review**, 8 draws per reaction. Baseline accuracy after neutral review S1/S2/S3: 28.0 / 38.3 / 41.1%. All cells remain significant under Holm–Bonferroni at $\alpha = 0.05$.

Feedback	S1	S2	S3
Mild doubt	-0.34 ($p = 0.020$)	-0.71 ($p < 10^{-5}$)	-2.10 ($p < 10^{-26}$)
Told wrong	-3.71 ($p < 10^{-31}$)	-4.13 ($p < 10^{-45}$)	-9.22 ($p < 10^{-134}$)
Marked wrong (authority)	-3.18 ($p < 10^{-31}$)	-3.73 ($p < 10^{-43}$)	-8.69 ($p < 10^{-140}$)
Told correct	$+0.66$ ($p < 10^{-11}$)	$+0.47$ ($p < 10^{-7}$)	$+0.36$ ($p < 10^{-3}$)
"Told wrong" loss as a share of baseline accuracy			
	13%	11%	22%

Table 16: **Answer flips relative to the first answer** (%, mean over 8 draws). C→I: share of initially correct answers that end incorrect. I→C: share of initially incorrect answers that end correct. A fresh resample is the no-feedback reference.

	S1		S2		S3	
Condition	C→I	I→C	C→I	I→C	C→I	I→C
Fresh resample	41.8	17.2	28.0	17.7	25.1	17.4
Neutral review	5.7	1.3	3.2	1.1	3.6	1.8
Mild doubt	10.6	2.8	7.8	2.9	11.7	3.9
Told wrong	31.0	6.3	21.5	6.0	32.3	6.4
Marked wrong	26.6	5.2	19.9	5.6	30.1	5.8
Told correct	2.5	0.9	1.9	1.0	2.6	1.7

What backtracking does not explain. Across stages, the size of the "told wrong" loss does not track the amount of backtracking: S3 loses the most accuracy while backtracking less than S1 (3.3 vs. 4.9 events per 1k). Backtracking counts are therefore consistent with "more reconsidering does not help", but they do not explain why S3 is the most fragile; the flip rates (Table 16) are the more direct measure.

D.5 Do structure changes within a trace reach the right product?

Question. When the structures the model writes down change during a trace, do the changes move toward the recorded product?

Supports. §4.4 and §5: the model produces the look of reconsidering without its useful part.

Takeaway. Few traces (7–11%) contain two or more distinct structures, so there is little change to measure. Among changes, reaching the recorded product (wrong→right) is more common than leaving it (right→wrong) at every stage and prompt, but most changes are between two structures that are not the recorded product (72–82%).

Method. We extract every SMILES-like string inside the thinking block of the first draw that RDKit parses and that is not a component of the starting material or reagents, and canonicalize it with RDKit. Structures are compared by canonical SMILES, so the same molecule written two ways is not counted as a change. A *change* is a step from one extracted structure to a different one, and each change is classified by comparing both structures with the recorded product. Extracted structures include intermediates and by-products as well as candidate products, so "wrong" here means "not the recorded product": wrong→wrong changes include steps through a mechanism, and wrong→right means the trace reaches the recorded product. Consistent with this, among traces with at least one change the final answer equals the last extracted structure in only 36–69% of cases.

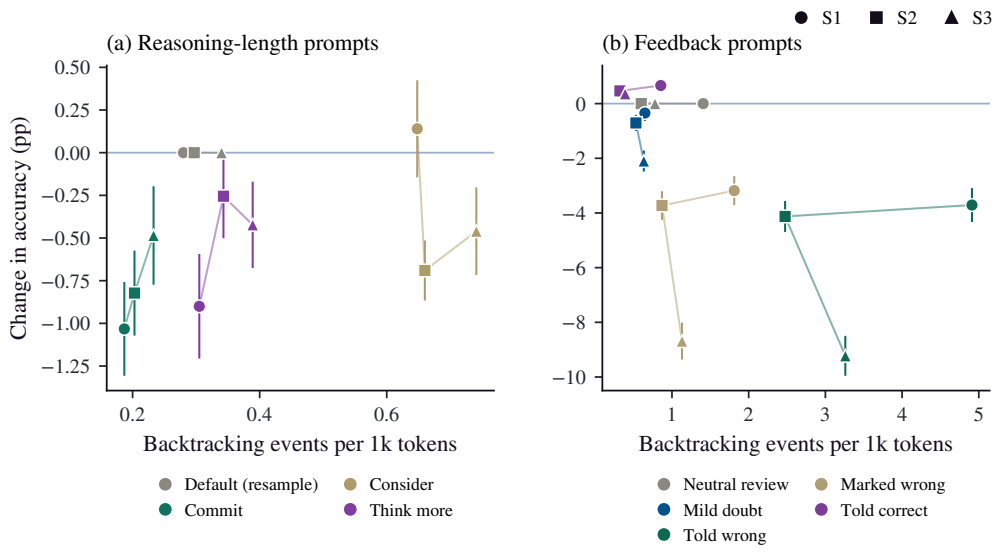


Figure 9: **Prompts raise backtracking without raising accuracy.** Change in accuracy against backtracking events per 1k output tokens for (a) reasoning-length prompts and (b) feedback prompts, by stage. "Consider alternatives" raises backtracking about 2.4× (0.27→0.66 per 1k at S1) with no accuracy gain; "told wrong" raises it 8–18× over the default resample (3.5–4.1× over the neutral review) and produces the largest losses.

Table 17: **Structure changes within the trace** (RDKit-canonicalized SMILES; first draw). Columns 3–4: % of traces; columns 5–7: % of change events; column 8: % of traces with ≥1 change.

Stage	Condition	Any candidate	≥1 change	Wrong →right	Right →wrong	Wrong →wrong	Final answer = last cand.
S1	Default	64.0	8.0	16.8	4.0	79.2	68.4
	Consider alternatives	58.2	7.8	17.0	1.4	81.6	68.9
	Think more	64.3	9.6	18.2	3.3	78.6	60.7
	Told wrong	51.4	11.1	13.3	8.5	78.2	38.7
S3	Default	58.5	7.9	25.9	2.2	71.9	62.9
	Consider alternatives	54.6	7.0	26.3	2.1	71.7	64.8
	Think more	56.7	8.4	22.5	3.0	74.5	59.0
	Told wrong	49.8	7.8	16.4	9.4	74.2	36.0

D.6 Sampling several answers per reaction

Question. Does self-training teach new reactions, or sharpen answers the model could already reach?

Supports. §4.3: self-training mostly sharpens.

Takeaway. Mostly sharpens. From S1 to S3, single-sample accuracy rises 12.9 points but pass@9 rises only 4.1 (63.8→67.9%). Plurality voting over nine S1 samples matches or exceeds one S3 sample (46.1 vs. 41.6% overall, and in 8 of the 10 classes with $n \geq 40$); heterocycle formation stays near zero at every k for the 8B model (pass@9 ≤1.9%).

The exceptions to "nine S1 samples beat one S3 sample" are oxidations (49.1 vs. 50.9%) and heterocycle formation (0 vs. 0%).

Table 18: **Plurality@k and pass@k (%), Qwen3-8B, all 5,000 reactions.** Nine draws per reaction (the first answer plus 8 fresh resamples); pass@k is the empirical share of reactions with a correct answer among the first k draws.

Stage	Metric	$k=1$	$k=3$	$k=5$	$k=7$	$k=9$
S1	plurality@k	28.7	35.4	41.3	44.2	46.1
	pass@k	28.7	47.5	55.8	60.3	63.8
S2	plurality@k	38.9	44.6	48.5	50.6	51.7
	pass@k	38.9	55.0	61.2	65.0	67.4
S3	plurality@k	41.6	46.9	50.0	52.1	54.0
	pass@k	41.6	56.9	62.5	65.7	67.9

E STATISTICAL METHODS

Comparisons between models or stages. All models are evaluated on the same 5,000 reactions, so comparisons use paired tests. For one sample per reaction (App. B and C) we use McNemar's test with continuity correction on the discordant pairs and report paired Wald 95% confidence intervals on the accuracy difference.

Repeated draws. For the §4.3 probes with 8 draws per reaction, draws of the same reaction are not independent. We average within each reaction and test the per-reaction mean difference across reactions with a two-sided normal (z) test, with standard error $\mathrm{SD}/\sqrt{n}$ over the $n = 5{,}000$ reactions. Standard errors are therefore clustered by reaction (0.22–0.25 points for the reasoning-length prompts).

Single-model intervals. Error bars on single-model accuracy are 95% Wilson score intervals. They describe uncertainty in one model's accuracy and are not used to compare models; overlapping Wilson intervals do not imply equivalence.

Multiple comparisons. Within each family of related tests (the 9 reasoning-length cells; the 12 feedback cells) we report Holm–Bonferroni-adjusted significance alongside raw p-values.

Training variance. Each configuration was trained once. Test-set p-values describe uncertainty over reactions and sampling, not variation between training runs.