LLM-AS-JUDGE MEETS LLM-AS-OPTIMIZER: ENHANCING ORGANIC DATA EXTRACTION EVALUATIONS THROUGH DUAL LLM APPROACHES

Martiño Ríos-García[1] **Kevin Maik Jablonka**[1,2,3,4]
martino.rios@uni-jena.de mail@kjablonka.com

[1]Laboratory of Organic and Macromolecular Chemistry (IOMC)
[2]Center for Energy and Environmental Chemistry Jena (CEEC Jena)
[3]Helmholtz Institute for Polymers in Energy Applications Jena (HIPOLE Jena)
[4]Jena Center for Soft Matter (JCSM)

ABSTRACT

Large language models (LLMs) show promise for extracting structured data from scientific literature, but their use in chemistry faces unique challenges due to the complex, variable nature of experimental procedures. Here, we present a dual-LLM framework that combines an LLM-as-Judge to evaluate data extraction quality with an LLM-as-Optimizer to refine evaluation prompts systematically. To evaluate the performance, we leverage a manually annotated dataset of over 800 reaction steps in an action-centric schema that captures the sequential nature of chemical procedures rather than relying on rigid key-value pairs that are conventionally used. Through systematic analysis of various parameters, including temperature settings and prompt structures, we identify optimal configurations that maximize agreement with expert chemists while minimizing computational costs. The framework shows good agreement with expert annotations while reducing manual prompt engineering effort. This systematic approach not only demonstrates how modern machine learning techniques can address fundamental challenges in scientific data extraction but also provides a reusable pipeline for evaluating extraction results across domains where experimental variability has historically limited the development of standardized evaluation metrics.

1 INTRODUCTION

Recent advances in large language models (LLMs) (DeepSeek-AI et al., 2024; Radford et al., 2019; Brown et al., 2020; Grattafiori et al., 2024) have opened transformative opportunities for leveraging datasets that underpin scientific knowledge. A significant portion of this knowledge remains embedded in unstructured formats across vast, heterogeneous scientific literature, posing challenges for systematic analysis and accessibility (Schilling-Wilhelmi et al., 2025). Traditional deep learning approaches, though impactful, are constrained by their reliance on large-scale labeled datasets (Kononova et al., 2021; Swain & Cole, 2016). This requirement becomes impractical given the scale and diversity of modern scientific publications (Olivetti et al., 2020).

LLMs address this bottleneck by enabling efficient extraction of structured insights from unstructured corpora. Their ability to parse domain-specific text with minimal supervision (Khalighinejad et al., 2024) reduces dependence on labor-intensive manual annotation and custom extraction pipelines. This capability accelerates the initial stages of data processing and facilitates cross-disciplinary knowledge synthesis at unprecedented scales (Guo et al., 2021; Schilling-Wilhelmi & Jablonka, 2024; Ai et al., 2024). These advancements situate LLMs as a promising solution to the data-extraction challenge, with recent research highlighting their adaptability across diverse scientific contexts (Suvarna et al., 2023; Polak & Morgan, 2024; Gupta et al., 2022).

A domain in which the lack of high-quality datasets is particularly limiting progress is organic chemistry (Jablonka et al., 2022; Davies, 2019). Current organic chemistry datasets, predominantly de-

rived from patents (Lowe, 2012; Kearnes et al., 2021; Mayfield et al., 2018; Schwaller et al., 2022), suffer from fragmentation, bias toward a small number of reaction types (Jia et al., 2019; Schneider et al., 2016; Brown & Boström, 2015), and incomplete procedural details (e.g., temperature, solvents), hindering reproducibility, mechanistic insights, and the generalization of deep learning models for synthesis planning (Bradshaw et al., 2025).

However, existing extraction frameworks (Bran et al., 2024) exacerbate this by enforcing rigid schemas that oversimplify experimental variability (e.g., reagent notation, reaction setups), undermining predictive capabilities and model robustness.

In addition, evaluating organic reaction data extraction poses unique challenges due to chemical systems' complexity and high variability (Patiny & Godin, 2023). Reactions differ widely in their conditions (e.g., temperature gradients, solvent interactions) or reagent roles (catalytic vs. stoichiometric)—each introducing layers of experimental and semantic ambiguity. This variability complicates the development of bespoke standardized evaluation metrics, as too many possible cases of notation and meaning must be considered.

As an alternative to hand-crafted evaluation pipelines, LLMs are commonly used to assess the performance of various systems (Zheng et al., 2023a). Yet, despite their potential, those LLM-based evaluation frameworks remain fragile for addressing these challenges: they rely on manually optimized prompts that are sensitive to subtle phrasing changes, requiring labor-intensive refinement that often fails to align with human expert judgment.

In this work, we show how data extraction into a very flexible action-centric data model for organic reactions can be systematically evaluated using LLM-as-Judge approaches. We identify key parameters (e.g., temperature settings, few-shot example selection) that stabilize evaluations across diverse reaction types. Furthermore, we augment this framework with an LLM-as-Optimizer agent, establishing a systematic self-improving evaluation loop. By directly addressing the interplay between organic chemistry's variability and the structured demands of machine learning pipelines, this dual approach advances robust, scalable frameworks for evaluating scientific data extraction.

Concretely, our main contributions are:

1. **Extraction of organic reaction data into a sequential schema:** To account for the almost limitless flexibility in experimental procedures, we, for the first time, automatically extract data in an action-centered sequential data model rather than rigid key-value pairs. This approach also preserves temporal relationships between reaction steps.
2. **Systematic sensitivity analysis of LLM-as-Judge for data extraction:** We quantify how evaluation parameters affect the reliability of LLM-based assessment of chemical data extraction. Our analysis reveals optimal temperature settings, prompt structures, and few-shot example selections that maximize agreement with expert chemists while minimizing computational costs.
3. **Optimization of LLM-as-Judge pipeline using an LLM-as-Optimizer approach:** We demonstrate how an LLM-as-Optimizer can systematically refine evaluation prompts through a feedback loop with the LLM-as-Judge. This dual-LLM self-optimization loop slightly improves the agreement with expert annotations while reducing prompt engineering effort compared to manual optimization. Our approach represents a reusable pipeline that can easily be adapted in other data extraction cases.

2 Related Work

LLM-as-Judge The paradigm of LLM-as-Judge has gained traction for automating evaluations across diverse tasks. For instance, LLMs have been employed to assess question-answer correctness (Thakur et al., 2025; Li et al., 2023) and to evaluate summarization tasks (Gao et al., 2023; Luo et al., 2023; Zheng et al., 2023b). Beyond task-specific applications, a prominent focus of this paradigm lies in safety and harm mitigation (Inan et al., 2023; Wang et al., 2024). A foundational contribution in this domain is Constitutional AI (CAI) by Bai et al. (2022), which introduced an iterative framework where an LLM evaluator critiques and revises harmful responses. These refined responses are then integrated into instruction-tuning datasets, while preference-tuning utilizes LLM judgments to select safer outputs from paired candidates. In a similar approach, Guan et al. (2025)

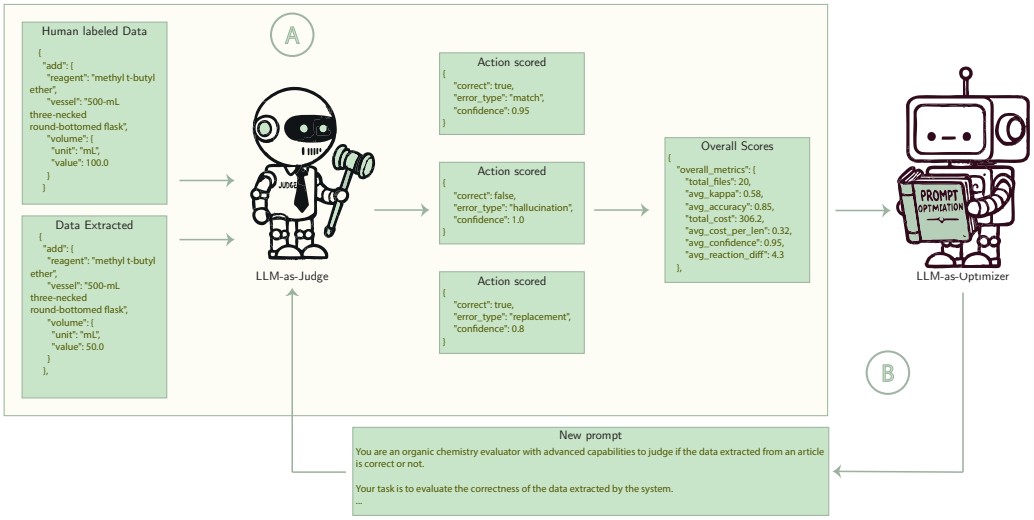


Figure 1: **Workflow overview showing two distinct components: (A) LLM-as-Judge scoring module and (B) LLM-as-Optimizer feedback loop. A.** The LLM-as-Judge module evaluates the extracted data by comparing it against human-labeled ground truth, generating quantitative scores. **B.** The LLM-as-Optimizer operates iteratively over a fixed 20-cycle process: it refines prompts using previous prompts and the scores of the LLM-as-Judge evaluation, proposing optimized prompts for subsequent evaluation. This closed-loop framework ensures systematic alignment with ground-truth benchmarks through repeated feedback-driven adjustments.

demonstrated using a safety-aware LLM Judge to generate reward signals for reinforcement learning (RL), enabling harm reduction without reliance on human-labeled data.

However, the reliability of LLM-based evaluation systems remains a critical concern in AI safety research (Bavaresco et al., 2024). As highlighted by Shankar et al. (2024), deploying LLMs as evaluators demands rigorous validation to address inherent biases and methodological inconsistencies. Their work advocates for systematic benchmarking to strengthen automated safety evaluations, particularly emphasizing the dynamic nature of human-aligned assessment criteria.

While the LLM-as-Judge framework has been widely employed across diverse applications (Li et al., 2024a; Völker et al., 2024), to the best of our knowledge, no prior study has rigorously evaluated or methodically optimized its utility for the task of scientific data extraction.

LLM-as-Optimizer Despite advancements in LLMs, their performance remains highly sensitive to prompt variations (Salinas & Morstatter, 2024; Li et al., 2024b). While manual trial-and-error persists, automated methods like DSPy (Khattab et al., 2022) and LLM-driven optimization strategies show promise (Wang et al., 2023; Guo et al., 2024; Das et al., 2024). Notably, Yang et al. (2023) introduced Optimization by PROmpting (OPRO), using natural language meta-prompts to refine solutions based on previous prompts-scores data iteratively. Parallel work by Billa et al. (2024) proposed Supervisory Prompt Training (SPT), employing dual LLMs (generator and corrector) in an iterative feedback loop. Both methodologies exhibit enhanced performance with prolonged optimization cycles (McAleese et al., 2024; Cohen et al., 2023).

Despite advancements in LLM-driven methodologies, systematic analysis of prompt performance for data extraction and evaluation tasks remains unexplored. Building on top of Yang et al. (2023)'s parameter optimization framework and Billa et al. (2024)'s dual-agent collaborative architecture, our work tries to address critical gaps in the task-specific adaptability of the LLM-as-Judge framework and the evaluation of data extraction tasks.

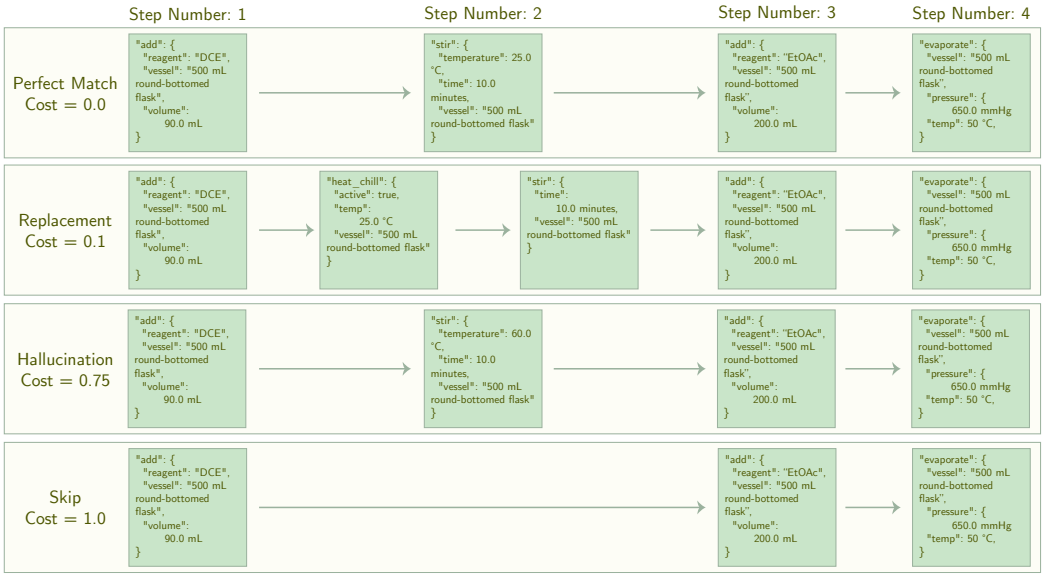


Figure 2: **Types of errors evaluated in the cost function calculation. A.** Perfect Match: The reaction step matches the ground truth data. **B.** Replacement error: A single step is subdivided into two valid actions without disrupting the reaction pathway. **C.** Hallucination error: An invalid parameter (e.g., volume value) is introduced, which may disrupt the reaction depending on thermodynamic or kinetic requirements. **D.** Skip error: A critical step is omitted (Step 2), halting the reaction progression.

3 METHODS

3.1 DATA MODEL, ANNOTATION, AND EXTRACTION

Data model Our action-centric schema represents each reaction as a sequence of discrete experimental steps, where each action captures the type of operation (e.g., addition, heating, stirring), temporal information (duration, sequence), and associated parameters (quantities, conditions; two examples can be found in Appendix A.2). This approach differs fundamentally from traditional rigid schemas by preserving procedural logic through explicit ordering and allowing flexible representation of complex multi-step procedures. Our data model is built on the Chemical Description Language XDL (Mehr et al., 2020) and hence, in principle, could directly be compiled to code that can be used for automated chemical synthesis on a system like the Chemputer (Steiner et al., 2019).

Data extraction We employ a multi-stage computational architecture for structured information extraction, combining rule-based parsing with advanced LLM-based extraction. The core extraction process utilizes the closed-source model from Anthropic Claude Sonnet 3.5 (`claude-3-5-sonnet-20241022`), a state-of-the-art multimodal language model.

The extraction workflow is implemented using Instructor (Liu), a specialized library enabling constrained generation through schema-guided output control. We employ a `Pydantic` schema to configure the system, explicitly defining the structure and relationships of the reaction actions.

Data Annotation An organic chemist labeled the ground truth dataset through a Human-in-the-Loop workflow (Dagdelen et al., 2024a), rigorously correcting the extracted reaction steps (see Appendix A.1). This process involved classifying each step as valid or erroneous. The dataset itself comprises reaction steps systematically extracted from 20 peer-reviewed articles. To ensure unbiased evaluation during iterative model optimization, the data were partitioned into training and testing subsets using a randomized 75:25 split.

3.2 WORKFLOW

Our approach combines iterative evaluation and prompt refinement through two interconnected frameworks—LLM-as-Judge and LLM-as-Optimizer to enhance model performance (Figure 1).

The workflow progresses through two interdependent phases:

1. *LLM-as-Judge*: Quantitative and qualitative evaluation of model outputs against expert-validated ground-truth data. Beyond binary correctness classification, it provides structured rationales for errors, enabling subsequent optimization.
2. *LLM-as-Optimizer*: Targeted refinement of the prompts of the LLM-as-Judge using Judge-derived scores.

This dual architecture enables continuous performance improvement, where evaluation outcomes directly guide subsequent prompt adjustments while maintaining human oversight through scoring.

3.3 LLM-AS-JUDGE

The LLM-as-Judge conducts fine-grained comparisons between model-extracted reaction data and expert-labeled ground truth. We implement a categorical evaluation system rather than numerical scoring to ensure reproducible scoring and minimize ambiguity (see Section 3.5). For each reaction step, the Judge provides three output variables:

- `correct`: Boolean indicating if the error at hand would cause the reaction to fail or not proceed as desired, guided by instructions provided in the prompt (see Appendix A.3.
- `error_type`: Categorical classification of discrepancies (see Figure 2).
- `confidence`: Self-reported certainty score (0–1 scale). This can be considered as a verbalized confidence estimate (Xiong et al., 2023; Mirza et al., 2024).

To constrain the `error_type`, we define four different categories with associated cost weights for quantitative analysis (see Section 3.5):

$$\text{cost}(a) = \begin{cases} 0.0 & \text{if } a \text{ perfectly matches ground truth} \\ 0.1 & \text{if } a \text{ is semantically equivalent but lexically distinct} \\ 0.75 & \text{if } a \text{ contains unsupported information} \\ 1.0 & \text{if critical ground truth information is missing} \end{cases} \tag{1}$$

Ablation Studies We systematically studied the best conditions for the Judge component through controlled experiments comparing five key parameters (detailed prompts and `Pydantic` schema in Appendix A.3):

- **Evaluation granularity**: Per-step evaluation (multiple LLM calls) vs. batch evaluation (single LLM call). In the per-step, the model is prompted to focus on one specific step each time, while in the batch evaluation, it is asked to loop over all the reaction steps.
- **Reasoning requirements**: Consists of introducing a critique output variable, asking the model to think step-by-step before providing the other variables.
- **Stochastic sampling**: In stochastic sampling experiments, deterministic (temperature=0.0) and stochastic (0.3, 0.7, 1.0) conditions were evaluated, with non-zero temperatures averaged over eight runs (@8) to mitigate variability and ensure robust performance assessment.
- **Exemplar usage**: We compare zero-shot evaluation vs. few-shot with critiquing examples.
- **Judge model**: We test `Claude-3-5-Sonnet-20241022`, `GPT-4o-2024-08-06` (OpenAI et al., 2024a), and `o1-2024-12-17` (OpenAI et al., 2024b) for all the ablations above.

This parametric analysis informed our final configuration by quantifying each variable's impact on evaluation accuracy and computational efficiency.

3.4 LLM-AS-OPTIMIZER

The LLM-as-Optimizer performs an iterative refinement of the Judge's prompt by being provided with previous prompts and the corresponding scores of the evaluation of the training set. The Optimizer is configured with the most optimal settings from the work of (Yang et al., 2023) with the same models as for the Judge.

- **Temperature**: Following Yang et al. (2023) we use a temperature of 1.
- **Number of steps**: In contrast to Yang et al. (2023), we limit the number of steps to 20, given that their work showed lower improvement rates above 20 steps.
- **Prompt-scores sorting**: Following Yang et al. (2023), we sorted the prompt-score pairs ascending based on the score value (detailed prompt in Appendix A.4).
- **Prompts per Iteration**: Yang et al. (2023) conclude that the best results are obtained by sampling eight prompts each iteration when a high number of steps is employed. However, they show that one prompt per iteration shows a higher rate of improvement for a low number of steps, as in our case.

By mirroring reinforcement learning from AI feedback (RLAIF) principles—but replacing model training with prompt engineering—we iteratively enhance evaluation reliability through Chain-of-Thought (CoT) prompting strategies (Wei et al., 2023), which improve the Judge's reasoning transparency and task alignment (Billa et al., 2024).

3.5 METRICS

We use the manually labeled ground truth to evaluate our extraction pipeline and the annotations to assess the LLM-as-Judge performance through three metrics:

Accuracy The accuracy acc for each reaction is the fraction of correctly extracted steps.

Cohen's Kappa We use Cohen's κ to measure the agreement between the LLM output and expert annotations. Unlike Spearman's ρ or Kendall's τ, κ provides more conservative estimates for categorical data. We average κ across all reactions.

Cost per Step Binary metrics like Accuracy and Cohen's κ only capture whether a reaction step succeeds or fails. To account for the nuanced variations in organic reactions, we introduce a cost-based metric that penalizes different extraction errors (Figure 2). We normalize this cost by the number of reaction steps to enable comparisons across procedures of different lengths.

We combine these metrics into a single score, s:

$$s = \frac{(\mathrm{acc} + \kappa)}{\mathrm{cost}}, \tag{2}$$

where the cost is based on Equation (1).

The optimization objective then becomes:

$$\theta^*(\text{prompt}) = \arg\max\ (s) \tag{3}$$

This Score balances prediction accuracy and statistical agreement against extraction costs, guiding the Optimizer toward prompts that maximize extraction quality while minimizing errors.

4 RESULTS

We used a series of experiments and ablations using our framework of LLM-as-Judge plus LLM-as-Optimizer to systematically evaluate the best conditions for the Judge when evaluating the task of organic data extraction.

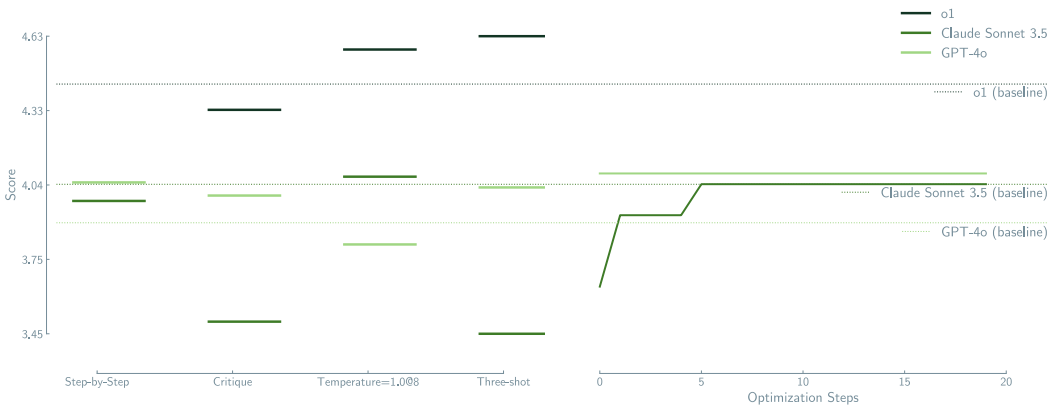


Figure 3: **Overview of LLM-as-Judge ablation studies (left) and prompt optimization process (right).** Baseline experiments use no critique, 0-shot evaluation, full text-based evaluation, and temperature 0 (temperature 1 for o1 due to model constraints by its provider, with results averaged over a single run). Due to time constraints, the ablation study involving step-by-step evaluation and the o1-based prompt optimization were omitted. Optimization results (right) reflect evaluations using the best prompt identified at each iterative step. For a detailed breakdown of optimization progress using the prompt generated across each step, see Figure 6. Discrepancies emerge between initial optimization points and the baseline despite identical prompts and a temperature setting of zero. We note variability even under these deterministic conditions, which we hypothesize may arise from default parameter configurations (e.g., `log_p`, `top_k`). Further investigation is required to elucidate the underlying causes.

4.1 JUDGE ABLATIONS

The ablation studies in the test set assessing model capabilities and optimal conditions for the LLM-as-Judge framework are presented on the left side of Figure 3 (to see the results for the training set, please refer to Appendix A.5). Baseline comparisons include configurations without critique generation, zero-shot evaluation, full-text-based evaluation, and temperature-zero sampling (except for o1, which inherently requires temperature=1 due to the provider constraints). Across these baselines, a consistent performance hierarchy emerges: o1 significantly outperforms both Claude and GPT-4o, with the latter two models exhibiting mixed results across ablations.

Notably, Claude exhibits counterintuitive behavioral patterns in response to specific ablations. For instance, including three-shot examples or explicit requests for critique paradoxically degrades its performance. This observation is particularly intriguing given that the critique component—embedded within the three-shot examples is explicitly designed to scaffold the model's reasoning process during inference.

A parallel discrepancy emerges in o1's behavior: while three-shot prompting improves its performance relative to the baseline, critique generation reduces accuracy in a similar measure under identical conditions. This contrast is unexpected, as the provided three-shot examples (see Appendix A.3) contain critique-guided refinement to enhance task alignment. The divergent responses of Claude and o1 to similar interventions highlight nuanced model sensitivities, implying that critique integration and reasoning mechanisms may require model-specific optimization, ideally using an automated pipeline, rather than serving as universal performance enhancers.

Temperature ablation analyses are more complex and difficult to predict: while higher temperature values (e.g., 1.0) may enable creative responses that improve task performance, they simultaneously increase hallucination risks. To mitigate stochastic variability, temperature-ablated results represent averages across eight independent runs (see Appendix A.7 for a more detailed analysis). The per-step evaluation ablation for o1 was omitted due to time constraints. However, for the other models, we observe some contrast again; while GPT-4o evaluation improves with a step-by-step-based evaluation, the performance degrades for Claude.

Our findings indicate that the optimal configuration for the LLM-as-Judge framework combines the o1 model (temperature=1.0, averaged over eight runs) with three-shot prompting. Additionally, full-text evaluation is more convenient than per-step evaluation due to time and economic constraints. Finally, we observe agreement rates higher than 50% with human data in both test and training sets, while accuracy surpasses 80% in all scenarios for Claude 3.5 Sonnet and o1 (Appendix A.6).

4.2 Optimizer Runs

The optimization experiments employed identical parameters to the baseline configuration described in the previous section (including the same prompt template, temperature = 0, critique disabled, and full-text zero-shot evaluation). Results are displayed on the right side of Figure 3.

For Claude Sonnet 3.5, performance improved significantly during early optimization steps, stabilizing after step 5 with a final prompt marginally outperforming the baseline. In contrast, GPT-4o exhibited divergent behavior: while it failed to improve the scores from the first optimization iteration, it nevertheless surpassed baseline performance——a trend inconsistent with prior findings in LLM-driven optimization literature (Yang et al., 2023). Final optimized prompts for both models are provided in Appendix A.4.1.

4.3 Confidence Analysis

We systematically analyzed how confidence metrics correlate with evaluation scores across ablation conditions to evaluate the relationship between model confidence and the quality of LLM-as-Judge evaluations. This investigation aimed to determine whether variations in evaluation parameters (e.g., prompt structure, instruction sets) influence the models' self-reported confidence and whether such confidence reflected judgment accuracy.

As illustrated in the confidence distributions (Figure 4), ablation parameters exhibited minimal influence on confidence values, with patterns remaining consistent across the different parameter changes. A notable parallel relation emerged between confidence and evaluation scores: configurations producing higher scores consistently correlated with elevated confidence values.

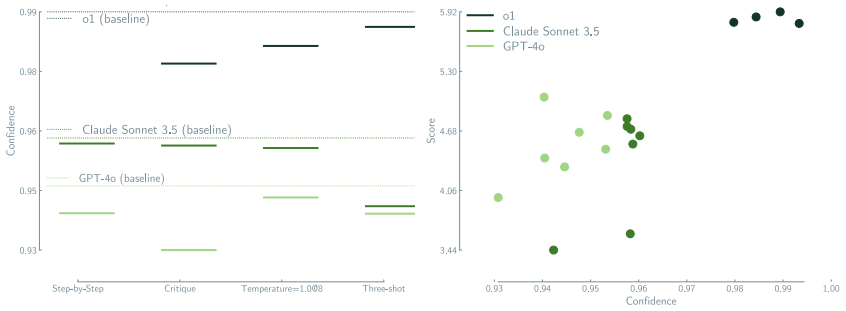
Model-specific behaviors further contextualize these findings. The Claude model displayed uniformly high confidence values across all ablation studies. Yet, this confidence showed little alignment with some score variations, implying a potential disconnect between its self-assurance and judgment quality. In contrast, GPT-4o exhibited no clear confidence-score relationship, with scattered values indicating inconsistent calibration.

While confidence distributions appeared similar between training and test sets, systematic score differences emerged. Training set evaluations showed slightly inflated scores compared to test conditions, hinting at possible overconfidence during training data assessments.

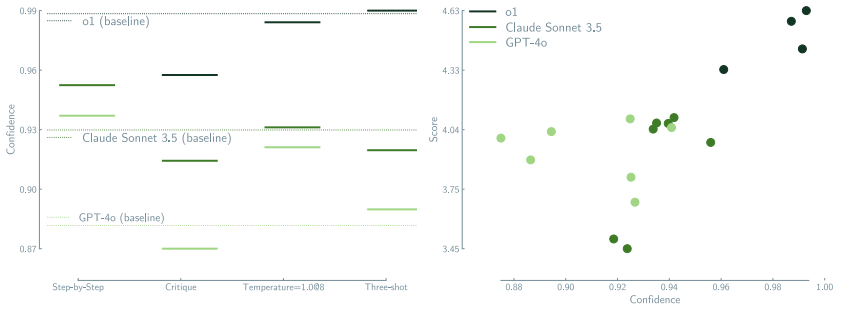
Despite these trends, substantial confidence variance persisted even for identical evaluation scores across all models. This observation underscores the limitations of interpreting raw (verbalized) confidence metrics as direct proxies for judgment reliability.

5 Limitations

While our approach shows promise, several limitations should be acknowledged. The prompt design process relied on a non-optimized distribution of examples, with only three manually crafted instances included in the prompts, potentially constraining the robustness of the approach (Zhou et al., 2024). Additionally, the lack of standardized units in input data introduces a risk of model hallucination by the Judge—a challenge that could be addressed using packages such as `pint` (Grecco, 2014) or `unyt` (Goldbaum et al., 2018). The dataset's scope is constrained by both size and source diversity, drawing from a single scientific journal, which may limit generalizability across different organic reaction procedures. Furthermore, reliance on closed-source LLMs imposes practical constraints, including high operational costs that restrict the Optimizer to only 20 iterations. Finally, the labeling process involved a single annotator, which risks subjective bias; future work should incorporate multiple annotators to assess inter-rater agreement and resolve discrepancies through methods such as third-party arbitration (Rein et al., 2023).



(a) A. Confidence distribution in the training set



(b) B. Confidence distribution in the test set

Figure 4: **Confidence distributions across ablation conditions.** (A) Training set analyses reveal slight score inflation despite confidence patterns being similar to the test set trends. (B) Test set distributions show marginally stronger confidence-score correlations, though with persistent variance. Both plots highlight model-specific calibration behaviors, with o1 configurations producing the highest confidence and score values.

6 CONCLUSIONS

Extracting structured data from scientific literature remains a fundamental challenge in chemistry and materials science. While the scientific community continuously generates vast amounts of valuable experimental data in publications, traditional approaches to extracting this knowledge have relied on rigid schemas and hand-crafted evaluation pipelines. These approaches fail to capture the inherent complexity of experimental sciences and create bottlenecks in developing scalable data extraction systems.

Large language models have recently emerged as a promising solution for extracting complex information from unstructured text. However, their application to specialized domains like organic chemistry faces two critical challenges: the need for flexible data models that can capture procedural complexity and the requirement for systematic, reliable evaluation frameworks that do not introduce new bottlenecks through manual optimization.

Here, we have shown how these challenges can be addressed through a dual-LLM approach that combines LLM-as-Judge with LLM-as-Optimizer. Our framework systematically optimizes evaluation prompts while maintaining high agreement with expert annotations. By coupling this evaluation framework with an action-oriented schema for organic reactions, we demonstrate how complex procedural information can be reliably extracted and evaluated at scale. The success of this approach—achieving high agreement with expert annotations—highlights how modern machine learning techniques can solve fundamental challenges in scientific data extraction for which no reliable alternatives exist.

Our work provides both immediate practical tools and broader methodological insights. The action-oriented extraction pipeline we developed will directly impact reaction prediction models by provid-

ing higher-quality training data. More fundamentally, our systematic approach to coupling LLM-as-Judge with LLM-as-Optimizer provides a template for developing reliable, scalable evaluation frameworks across other scientific domains where complex, structured data needs to be extracted from unstructured text.

ACKNOWLEDGMENTS

This work was supported by the Carl Zeiss Foundation. Part of the work of M.R.G. was supported by Intel and Merck via the AWASES Center.

K.M.J. is part of the NFDI consortium FAIRmat funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – project 460197019.

REFERENCES

Qianxiang Ai, Fanwang Meng, Jiale Shi, Brenden Pelkie, and Connor W. Coley. Extracting structured data from organic synthesis procedures using a fine-tuned large language model. *Digital Discovery*, 3(9):1822–1831, 2024. ISSN 2635-098X. doi: 10.1039/d4dd00091a. URL http://dx.doi.org/10.1039/D4DD00091A.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback, 2022. URL https://arxiv.org/abs/2212.08073.

Anna Bavaresco, Raffaella Bernardi, Leonardo Bertolazzi, Desmond Elliott, Raquel Fernández, Albert Gatt, Esam Ghaleb, Mario Giulianelli, Michael Hanna, Alexander Koller, André F. T. Martins, Philipp Mondorf, Vera Neplenbroek, Sandro Pezzelle, Barbara Plank, David Schlangen, Alessandro Suglia, Aditya K Surikuchi, Ece Takmaz, and Alberto Testoni. Llms instead of human judges? a large scale empirical study across 20 nlp evaluation tasks, 2024. URL https://arxiv.org/abs/2406.18403.

Jean Ghislain Billa, Min Oh, and Liang Du. Supervisory prompt training, 2024. URL https://arxiv.org/abs/2403.18051.

John Bradshaw, Anji Zhang, Babak Mahjour, David E. Graff, Marwin H. S. Segler, and Connor W. Coley. Challenging reaction prediction models to generalize to novel chemistry, 2025. URL https://arxiv.org/abs/2501.06669.

Andres M Bran, Zlatko Jončev, and Philippe Schwaller. Knowledge graph extraction from total synthesis documents. In *Proceedings of the 1st Workshop on Language+ Molecules (L+ M 2024)*, pp. 74–84, 2024.

Dean G. Brown and Jonas Boström. Analysis of past and present synthetic methodologies on medicinal chemistry: Where have all the new reactions gone?: Miniperspective. *Journal of Medicinal Chemistry*, 59(10):4443–4458, December 2015. ISSN 1520-4804. doi: 10.1021/acs.jmedchem.5b01409. URL http://dx.doi.org/10.1021/acs.jmedchem.5b01409.

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

Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. Lm vs lm: Detecting factual errors via cross examination, 2023. URL https://arxiv.org/abs/2305.13281.

John Dagdelen, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S. Rosen, Gerbrand Ceder, Kristin A. Persson, and Anubhav Jain. Structured information extraction from scientific text with large language models. *Nature Communications*, 15(1), February 2024a. ISSN 2041-1723. doi: 10.1038/s41467-024-45563-x. URL http://dx.doi.org/10.1038/s41467-024-45563-x.

John Dagdelen, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S. Rosen, Gerbrand Ceder, Kristin A. Persson, and Anubhav Jain. Structured information extraction from scientific text with large language models. *Nat. Commun.*, 15(1):1418, February 2024b. ISSN 2041-1723. doi: 10.1038/s41467-024-45563-x. URL https://www.nature.com/articles/s41467-024-45563-x.

Sarkar Snigdha Sarathi Das, Ryo Kamoi, Bo Pang, Yusen Zhang, Caiming Xiong, and Rui Zhang. Greater: Gradients over reasoning makes smaller language models strong prompt optimizers, 2024. URL https://arxiv.org/abs/2412.09722.

Ian W. Davies. The digitization of organic synthesis. *Nature*, 570(7760):175–181, June 2019. ISSN 1476-4687. doi: 10.1038/s41586-019-1288-y. URL http://dx.doi.org/10.1038/s41586-019-1288-y.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. Deepseek-v3 technical report, 2024. URL https://arxiv.org/abs/2412.19437.

Mingqi Gao, Jie Ruan, Renliang Sun, Xunjian Yin, Shiping Yang, and Xiaojun Wan. Human-like summarization evaluation with chatgpt, 2023. URL https://arxiv.org/abs/2304.02554.

Nathan J. Goldbaum, John A. ZuHone, Matthew J. Turk, Kacper Kowalik, and Anna L. Rosen. unyt: Handle, manipulate, and convert data with units in python. *J. Open Source Softw.*, 3(28):809, August 2018. doi: 10.21105/joss.00809.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj,

Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Hernan E. Grecco. Pint: a Python Units Library. https://github.com/hgrecco/pint, 2014.

Melody Y. Guan, Manas Joglekar, Eric Wallace, Saachi Jain, Boaz Barak, Alec Helyar, Rachel Dias, Andrea Vallone, Hongyu Ren, Jason Wei, Hyung Won Chung, Sam Toyer, Johannes Heidecke, Alex Beutel, and Amelia Glaese. Deliberative alignment: Reasoning enables safer language models, 2025. URL https://arxiv.org/abs/2412.16339.

Jiang Guo, A. Santiago Ibanez-Lopez, Hanyu Gao, Victor Quach, Connor W. Coley, Klavs F. Jensen, and Regina Barzilay. Automated chemical reaction extraction from scientific literature. *Journal of Chemical Information and Modeling*, 62(9):2035–2045, June 2021. ISSN 1549-960X. doi: 10.1021/acs.jcim.1c00284. URL http://dx.doi.org/10.1021/acs.jcim.1c00284.

Qingyan Guo, Rui Wang, Junliang Guo, Bei Li, Kaitao Song, Xu Tan, Guoqing Liu, Jiang Bian, and Yujiu Yang. Connecting large language models with evolutionary algorithms yields powerful prompt optimizers, 2024. URL https://arxiv.org/abs/2309.08532.

Tanishq Gupta, Mohd Zaki, N. M. Anoop Krishnan, and Mausam. Matscibert: A materials domain language model for text mining and information extraction. *npj Computational Materials*, 8(1), May 2022. ISSN 2057-3960. doi: 10.1038/s41524-022-00784-w. URL http://dx.doi.org/10.1038/s41524-022-00784-w.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. Ruler: What's the real context size of your long-context language models?, 2024. URL https://arxiv.org/abs/2404.06654.

Qian Huang, Jian Vora, Percy Liang, and Jure Leskovec. Mlagentbench: Evaluating language agents on machine learning experimentation, 2024. URL https://arxiv.org/abs/2310.03302.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. Llama guard: Llm-based input-output safeguard for human-ai conversations, 2023. URL https://arxiv.org/abs/2312.06674.

Kevin Maik Jablonka, Luc Patiny, and Berend Smit. Making the collective knowledge of chemistry open and machine actionable. *Nature Chemistry*, 14(4):365–376, April 2022. ISSN 1755-4349. doi: 10.1038/s41557-022-00910-7. URL http://dx.doi.org/10.1038/s41557-022-00910-7.

Xiwen Jia, Allyson Lynch, Yuheng Huang, Matthew Danielson, Immaculate Lang'at, Alexander Milder, Aaron E. Ruby, Hao Wang, Sorelle A. Friedler, Alexander J. Norquist, and Joshua Schrier. Anthropogenic biases in chemical reaction data hinder exploratory inorganic synthesis. *Nature*, 573(7773):251–255, September 2019. ISSN 1476-4687. doi: 10.1038/s41586-019-1540-5. URL http://dx.doi.org/10.1038/s41586-019-1540-5.

Steven M. Kearnes, Michael R. Maser, Michael Wleklinski, Anton Kast, Abigail G. Doyle, Spencer D. Dreher, Joel M. Hawkins, Klavs F. Jensen, and Connor W. Coley. The open reaction database. *Journal of the American Chemical Society*, 143(45):18820–18826, November 2021. ISSN 1520-5126. doi: 10.1021/jacs.1c09820. URL http://dx.doi.org/10.1021/jacs.1c09820.

Ghazal Khalighinejad, Defne Circi, L. C. Brinson, and Bhuwan Dhingra. Extracting polymer nanocomposite samples from full-length documents, 2024. URL https://arxiv.org/abs/2403.00260.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive NLP. *arXiv preprint arXiv:2212.14024*, 2022.

Olga Kononova, Tanjin He, Haoyan Huo, Amalie Trewartha, Elsa A. Olivetti, and Gerbrand Ceder. Opportunities and challenges of text mining in materials research. *iScience*, 24(3):102155, March 2021. ISSN 2589-0042. doi: 10.1016/j.isci.2021.102155. URL http://dx.doi.org/10.1016/j.isci.2021.102155.

Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. Llms-as-judges: A comprehensive survey on llm-based evaluation methods, 2024a. URL https://arxiv.org/abs/2412.05579.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. Halueval: A large-scale hallucination evaluation benchmark for large language models, 2023. URL https://arxiv.org/abs/2305.11747.

Mingqi Li, Karan Aggarwal, Yong Xie, Aitzaz Ahmad, and Stephen Lau. Learning from contrastive prompts: Automated optimization and adaptation, 2024b. URL https://arxiv.org/abs/2409.15199.

Jason Liu. jxnl/instructor: structured outputs for llms. https://github.com/jxnl/instructor/.

Daniel Mark Lowe. *Extraction of chemical structures and reactions from the literature*. PhD thesis, 2012. URL https://www.repository.cam.ac.uk/handle/1810/244727.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. Chatgpt as a factual inconsistency evaluator for text summarization, 2023. URL https://arxiv.org/abs/2303.15621.

John Mayfield, Daniel Lowe, and Roger Sayle. Pistachio. *Patent.[Online]. Available: https://www.nextmovesoftware. com/pistachio. html*, 2018.

Nat McAleese, Rai Michael Pokorny, Juan Felipe Ceron Uribe, Evgenia Nitishinskaya, Maja Trebacz, and Jan Leike. Llm critics help catch llm bugs, 2024. URL https://arxiv.org/abs/2407.00215.

S. Hessam M. Mehr, Matthew Craven, Artem I. Leonov, Graham Keenan, and Leroy Cronin. A universal system for digitization and automatic execution of the chemical synthesis literature. *Science*, 370(6512):101–108, October 2020. ISSN 1095-9203. doi: 10.1126/science.abc2986. URL http://dx.doi.org/10.1126/science.abc2986.

Adrian Mirza, Nawaf Alampara, Sreekanth Kunchapu, Martiño Ríos-García, Benedict Emoekabu, Aswanth Krishnan, Tanya Gupta, Mara Schilling-Wilhelmi, Macjonathan Okereke, Anagha Aneesh, Amir Mohammad Elahi, Mehrdad Asgari, Juliane Eberhardt, Hani M. Elbeheiry, María Victoria Gil, Maximilian Greiner, Caroline T. Holick, Christina Glaubitz, Tim Hoffmann, Abdelrahman Ibrahim, Lea C. Klepsch, Yannik Köster, Fabian Alexander Kreth, Jakob Meyer, Santiago Miret, Jan Matthias Peschel, Michael Ringleb, Nicole Roesner, Johanna Schreiber, Ulrich S. Schubert, Leanne M. Stafast, Dinga Wonanke, Michael Pieler, Philippe Schwaller, and Kevin Maik Jablonka. Are large language models superhuman chemists? *arXiv preprint arXiv: 2404.01475*, 2024.

Elsa A. Olivetti, Jacqueline M. Cole, Edward Kim, Olga Kononova, Gerbrand Ceder, Thomas Yong-Jin Han, and Anna M. Hiszpanski. Data-driven materials research enabled by natural language processing and information extraction. *Applied Physics Reviews*, 7(4), December 2020. ISSN 1931-9401. doi: 10.1063/5.0021106. URL http://dx.doi.org/10.1063/5.0021106.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mkadry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button,

Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. Gpt-4o system card. *arXiv preprint arXiv: 2410.21276*, 2024a.

OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, Ilge Akkaya, Ilya Kostrikov, Ilya Sutskever, Irina Kofman, Jakub Pachocki, James Lennon, Jason Wei, Jean Harb, Jerry Twore, Jiacheng Feng, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joaquin Qui nonero Candela, Joe Palermo, Joel Parish, Johannes Heidecke, John Hallman, John Rizzo, Jonathan Gordon, Jonathan Uesato, Jonathan Ward, Joost Huizinga, Julie Wang, Kai Chen, Kai Xiao, Karan Singhal, Karina Nguyen, Karl Cobbe, Katy Shi, Kayla Wood, Kendra Rimbach, Keren Gu-Lemberg, Kevin Liu, Kevin Lu, Kevin Stone, Kevin Yu, Lama Ahmad, Lauren Yang, Leo Liu, Leon Maksin, Leyton Ho, Liam Fedus, Lilian Weng, Linden Li, Lindsay McCallum, Lindsey Held, Lorenz Kuhn, Lukas Kondraciuk, Lukasz Kaiser, Luke Metz, Madelaine Boyd, Maja Trebacz, Manas Joglekar, Mark

Chen, Marko Tintor, Mason Meyer, Matt Jones, Matt Kaufer, Max Schwarzer, Meghan Shah, Mehmet Yatbaz, Melody Y. Guan, Mengyuan Xu, Mengyuan Yan, Mia Glaese, Mianna Chen, Michael Lampe, Michael Malek, Michele Wang, Michelle Fradin, Mike McClay, Mikhail Pavlov, Miles Wang, Mingxuan Wang, Mira Murati, Mo Bavarian, Mostafa Rohaninejad, Nat McAleese, Neil Chowdhury, Neil Chowdhury, Nick Ryder, Nikolas Tezak, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Patrick Chao, Paul Ashbourne, Pavel Izmailov, Peter Zhokhov, Rachel Dias, Rahul Arora, Randall Lin, Rapha Gontijo Lopes, Raz Gaon, Reah Miyara, Reimar Leike, Renny Hwang, Rhythm Garg, Robin Brown, Roshan James, Rui Shu, Ryan Cheu, Ryan Greene, Saachi Jain, Sam Altman, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Santiago Hernandez, Sasha Baker, Scott McKinney, Scottie Yan, Shengjia Zhao, Shengli Hu, Shibani Santurkar, Shraman Ray Chaudhuri, Shuyuan Zhang, Siyuan Fu, Spencer Papay, Steph Lin, Suchir Balaji, Suvansh Sanjeev, Szymon Sidor, Tal Broda, Aidan Clark, Tao Wang, Taylor Gordon, Ted Sanders, Tejal Patwardhan, Thibault Sottiaux, Thomas Degry, Thomas Dimson, Tianhao Zheng, Timur Garipov, Tom Stasi, Trapit Bansal, Trevor Creech, Troy Peterson, Tyna Eloundou, Valerie Qi, Vineet Kosaraju, Vinnie Monaco, Vitchyr Pong, Vlad Fomenko, Weiyi Zheng, Wenda Zhou, Wes McCabe, Wojciech Zaremba, Yann Dubois, Yinghai Lu, Yining Chen, Young Cha, Yu Bai, Yuchen He, Yuchen Zhang, Yunyun Wang, Zheng Shao, and Zhuohan Li. Openai o1 system card. *arXiv preprint arXiv: 2412.16720*, 2024b.

Luc Patiny and Guillaume Godin. Automatic extraction of fair data from publications using llm. *ChemRxiv preprint*, 2023. doi: 10.26434/chemrxiv-2023-05v1b-v2. URL https://chemrxiv.org/engage/chemrxiv/article-details/65570cb1dbd7c8b54b6ff36b.

Maciej P. Polak and Dane Morgan. Extracting accurate materials data from research papers with conversational language models and prompt engineering. *Nature Communications*, 15(1), February 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-45914-8. URL http://dx.doi.org/10.1038/s41467-024-45914-8.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI*, 2019. Accessed: 2024-11-15.

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

Abel Salinas and Fred Morstatter. The butterfly effect of altering prompts: How small changes and jailbreaks affect large language model performance, 2024. URL https://arxiv.org/abs/2401.03729.

Mara Schilling-Wilhelmi and Kevin Maik Jablonka. Using machine-learning and large-language-model extracted data to predict copolymerizations. In *AI for Accelerated Materials Design-Vienna 2024*, 2024.

Mara Schilling-Wilhelmi, Martiño Ríos-García, Sherjeel Shabih, María Victoria Gil, Santiago Miret, Christoph T. Koch, José A. Márquez, and Kevin Maik Jablonka. From text to insight: large language models for chemical data extraction. *Chemical Society Reviews*, 2025. ISSN 1460-4744. doi: 10.1039/d4cs00913d. URL http://dx.doi.org/10.1039/D4CS00913D.

Nadine Schneider, Daniel M. Lowe, Roger A. Sayle, Michael A. Tarselli, and Gregory A. Landrum. Big data from pharmaceutical patents: A computational analysis of medicinal chemists' bread and butter. *Journal of Medicinal Chemistry*, 59(9):4385–4402, April 2016. ISSN 1520-4804. doi: 10.1021/acs.jmedchem.6b00153. URL http://dx.doi.org/10.1021/acs.jmedchem.6b00153.

Philippe Schwaller, Alain C Vaucher, Ruben Laplaza, Charlotte Bunne, Andreas Krause, Clemence Corminboeuf, and Teodoro Laino. Machine intelligence for chemical reaction space. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 12(5):e1604, 2022.

Shreya Shankar, J. D. Zamfirescu-Pereira, Björn Hartmann, Aditya G. Parameswaran, and Ian Arawjo. Who validates the validators? aligning llm-assisted evaluation of llm outputs with human preferences, 2024. URL https://arxiv.org/abs/2404.12272.

Sebastian Steiner, Jakob Wolf, Stefan Glatzel, Anna Andreou, Jarosław M Granda, Graham Keenan, Trevor Hinkley, Gerardo Aragon-Camarasa, Philip J Kitson, Davide Angelone, et al. Organic synthesis in a modular robotic system driven by a chemical programming language. *Science*, 363 (6423):eaav2211, 2019.

Manu Suvarna, Alain Claude Vaucher, Sharon Mitchell, Teodoro Laino, and Javier Pérez-Ramírez. Language models and protocol standardization guidelines for accelerating synthesis planning in heterogeneous catalysis. *Nature Communications*, 14(1), December 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-43836-5. URL http://dx.doi.org/10.1038/s41467-023-43836-5.

Matthew C. Swain and Jacqueline M. Cole. Chemdataextractor: A toolkit for automated extraction of chemical information from the scientific literature. *Journal of Chemical Information and Modeling*, 56(10):1894–1904, October 2016. ISSN 1549-960X. doi: 10.1021/acs.jcim.6b00207. URL http://dx.doi.org/10.1021/acs.jcim.6b00207.

Aman Singh Thakur, Kartik Choudhary, Venkat Srinik Ramayapally, Sankaran Vaidyanathan, and Dieuwke Hupkes. Judging the judges: Evaluating alignment and vulnerabilities in llms-as-judges, 2025. URL https://arxiv.org/abs/2406.12624.

Christoph Völker, Tehseen Rug, Kevin Jablonka, and Sabine Kruschwitz. Llms can design sustainable concrete -a systematic benchmark (re-submitted version). 01 2024. doi: 10.13140/RG.2.2.33795.27686.

Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Haotian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P. Xing, and Zhiting Hu. Promptagent: Strategic planning with language models enables expert-level prompt optimization, 2023. URL https://arxiv.org/abs/2310.16427.

Zhenting Wang, Shuming Hu, Shiyu Zhao, Xiaowen Lin, Felix Juefei-Xu, Zhuowei Li, Ligong Han, Harihar Subramanyam, Li Chen, Jianfa Chen, Nan Jiang, Lingjuan Lyu, Shiqing Ma, Dimitris N. Metaxas, and Ankit Jain. Mllm-as-a-judge for image safety without human labeling, 2024. URL https://arxiv.org/abs/2501.00192.

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, 2023. URL https://arxiv.org/abs/2201.11903.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*, 2023.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. *International Conference on Learning Representations*, 2023. doi: 10.48550/arXiv.2309.03409. URL https://arxiv.org/abs/2309.03409v3.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023a. URL https://arxiv.org/abs/2306.05685.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023b. URL https://arxiv.org/abs/2306.05685.

Hanzhang Zhou, Zijian Feng, Zixiao Zhu, Junlang Qian, and Kezhi Mao. Unibias: Unveiling and mitigating llm bias through internal attention and ffn manipulation, 2024. URL https://arxiv.org/abs/2405.20612.

A APPENDIX

A.1 DATA CORPUS

The article curation process employed systematic random sampling and manual verification to ensure that all selected publications contained end-to-end organic synthesis procedures.

The labeled dataset comprises 20 peer-reviewed articles describing organic reaction syntheses. A Human-in-the-Loop approach was implemented to streamline the annotation of reaction steps into the action-based schema adopted in this work. In this hybrid methodology, the organic expert corrects the original extracted data by the model, notably reducing the time needed to label the entire corpus of the dataset (Dagdelen et al., 2024b). This hybrid methodology leveraged human expertise for context-sensitive labeling and LLM assistance to reduce manual effort. The result is a corpus of more than 800 labeled reaction steps.

The action ground data follows the action-based schema (two examples can be found in Appendix A.2). In contrast, the evaluation ground data schema is simplified, containing only step numbers from the ground truth data and a binary `correct` variable that indicates whether the reaction terminates at each reaction step.

Notably, the reaction procedures exhibited significant variation in length, ranging from 14 steps (shortest) to 119 steps (longest). To investigate potential correlations between procedural complexity and model performance, we conducted a dedicated analysis of score-length relationships across this spectrum (see Appendix A.8).

A.2 ACTIONS EXAMPLE

The data is extracted following an action-based schema, which intends to capture the complex nuances that organic reactions involve. To illustrate how these actions look, we include two simple examples from some of the data used in this work.


```
    {
    "step_number": 10,
    "stir": {
      "temp": {
        "value": "room temperature"
      },
      "time": {
        "unit": "h",
        "value": 18.0
      }
    }
  },
  {
    "step_number": 11,
    "transfer": {
      "from_vessel": "250 mL three-necked round-bottomed flask",
      "rinsing": false,
      "to_vessel": "100 mL round-bottomed flask"
    }
  }
```

```
{
  "add": {
    "reagent": "dichloromethane",
    "vessel": "300 mL two-necked round bottom flask",
    "volume": {
      "unit": "mL",
      "value": 20.0
    }
  },
  "step_number": 10
},
{
  "evaporate": {
    "method": "rotary evaporator",
    "pressure": {
      "unit": "Torr",
      "value": 18.0
    },
    "stir": false,
    "temp": {
      "unit": "\u00b0C",
      "value": 25.0
    },
    "vessel": "300 mL two-necked round bottom flask"
  },
  "step_number": 11
},
{
  "prepare_column": {
    "adsorbent": "silica gel",
    "adsorbent_mass": {
      "unit": "g",
      "value": 300.0
    },
    "column": "chromatography column",
    "topped_with": "sand",
    "wetted_with": "1:2 dichloromethane/hexane"
  },
  "step_number": 12
},
{
  "run_column": {
    "column": "chromatography column",
    "eluent": "1:2 dichloromethane/hexane",
    "eluent_volume": {
      "unit": "mL",
      "value": 4000.0
    },
    "from_vessel": "300 mL two-necked round bottom flask",
    "portions_collected": "fractions 15-23",
    "to_vessel": "200 mL Erlenmeyer flasks"
  },
  "step_number": 13
}
```

A.3 LLM-AS-JUDGE PROMPTS AND PYDANTIC SCHEMAS

The base prompt used for the different ablations varies among them to introduce the parameter variations that are introduced for each ablation. The baseline prompt includes guidelines for instructing the models on evaluating the correctness of the reaction steps.

```
You are an organic chemistry evaluator with advanced capabilities to
    judge if the data extracted from an article is correct or not.
Your task is to evaluate the correctness of the data extracted by the
    system.
You understand the nuances of organic reactions and can evaluate the
    details that are not important and which ones will impact the
    reaction.
To accomplish your task, you are provided with the ground truth data and
    the data extracted by the system.
The data will consist in both cases of a list of actions to follow in the
     organic reaction.
You are asked to loop over all the actions in the ground truth data,
    evaluating if each action is represented in the ground truth data is
    present in the extracted data.
To report the results of the evaluation return a list of the same length
    as the number of actions in the ground truth data, being each element
     of the list each of the actions of the ground truth data. To return
    the data follow the provided schema exactly.

The schema to follow for each of the steps are:
- `step_number`: the exact same `step_number` as in the ground truth data
    .
- `correct`: if the action from the ground truth data is present somehow
    in the extracted data, meaning that the organic reaction will not be
    compromised because of that step.
- `error_type`: the type of error of the extracted data for the specific
    action of the ground truth data.
- `confidence`: a value between 0 and 1 indicating the confidence level
    of your evaluation for the corresponding action.
To correctly evaluate the data extracted for the corresponding step,
    follow the next guidelines:
{guidelines}
```

Note that this system prompt is intended to include some guidelines that detail the specific instructions to evaluate the correctness of the reaction correctly:

```
For each action in the ground truth data:

- Evaluate if the ground truth data action is present somewhere in the
    data extracted.
- If the action is present, evaluate if the data is correct even if the
    name of the action changed, or if the data is expressed in a
    different way.
- If the step_number is different in the ground truth data and the
    extracted data, evaluate if the reaction will crash because of that
    variation.
- If the action is missing, and not similar action is detailed in place,
    assume that the reaction will crash.
- If the action is missing, but a similar action is detailed in place,
    evaluate if the reaction will crash because of that variation.
- Note that we do not consider the 'prepare_equipment' indispensable for
    the reaction to happen.
```

The user prompt only presents the ground truth and the extracted data to the model.

```
The ground truth data is the following:

{ground_truth}

The data extracted by the system is the following:

{data_extracted}

Please evaluate if the data extracted is correct or not for each of the
    steps.
```

The `Pydantic` schema presents the different variables needed for the evaluation:

```
class ErrorTypeEnum(str, Enum):
   MATCH = "match"
   HALLUCINATION = "hallucination"
   REPLACEMENT = "replacement"
   SKIP = "skip"


class JudgeSchema(BaseModel):
   step_number: float = Field(..., description="Exact same `step_number` as in the ground truth data.")
   correct: bool = Field(
      ...,
      description="Indicates if the ground truth data action is present somehow in the extracted data, meaning that the organic reaction will not be compromised because of that step.",
   )
   error_type: ErrorTypeEnum = Field(
      ...,
      description=(
         "Type of error in the extracted data for the specific action of the ground truth data:\n\n"
         "- **Perfect match**: The action in the extracted data exactly matches the ground truth data\n"
         "- **Hallucination**: The action in the extracted data contains information or details that are not present in the ground truth\n"
         "- **Replacement**: The action from the extracted data conveys the same meaning but uses different wording than the ground truth\n"
         "- **Skip**: The extracted content is missing information or details that were present in the ground truth"
      ),
   )
   confidence: float = Field(
      ...,
      description="Confidence level of the correctness of your evaluation for the corresponding action.",
      ge=0.0,
      le=1.0,
   )

class JudgeOutput(BaseModel):
   steps: List[JudgeSchema] = Field(
      ...,
      description="Evaluation of the extracted data. Each `step_number` in the ground truth data must be evaluated as a different item in this list.",
   )
```

Per-step evaluation For the ablations in which the per-step evaluation is performed, we add a simple instruction to the user prompt asking the model to focus on one specific reaction step, and the `JudgeSchema` is the one invoked for this case:

```
The step or action to focus on is the one with the following `step_number` in the ground truth data:

{step}
```

Critique The description of this variable is introduced in the system prompt and the `Pydantic` schema:

```
- 'critique': a detailed explanation for each of the actions in the
    ground truth data action about if it is somehow represented in the
    extracted data
```

```
critique: str = Field(
   ..., description="A␣detailed␣explanation␣for␣each␣action␣in␣the␣ground
       ␣truth␣data␣about␣if␣it␣is␣somehow␣represented␣in␣the␣extracted␣
       data."
)
```

Three-shot The three shots are added to the System prompt following the format:

```
To help you with the evaluation, three examples are provided below.

Example 1:

{Example_1}

Example 2:

{Example_2}

Example 3:

{Example_3}
```

From the text above, each example is replaced by the corresponding data, for example:

````
````
   {{
     "evacuate_and_refill": {{
       "gas": "nitrogen",
       "repeats": 3,
       "vessel": "100 mL round-bottomed flask"
     }},
     "step_number": 4
   }},
````

The data extracted by the system is the following:

````
````
   {{
     "prepare_equipment": {{
       "description": "sealed with a 24/40 rubber septum and connected to
           a Schlenk line via an 18-gauge x 1.5 in needle",
       "vessel": "100 mL round-bottomed flask"
     }},
     "step_number": 4
   }},
   {{
     "evacuate_and_refill": {{
       "gas": "nitrogen",
       "repeats": 3,
       "vessel": "100 mL round-bottomed flask"
     }},
     "step_number": 5
   }},
````

Evaluation:

```
- `step_number`: 4
- `critique`: "the action `evacuate_and_refill` is present in the
    extracted data, but the `step_number` is different. The reaction will
     crash because of that variation, but the action being evaluated is
    correct so the evaluation for the step 4 is positive."
- `correct`: true
- `error_type`: "match"
- `confidence`: 1.0
```

A.4 LLM-AS-OPTIMIZER PROMPT AND PYDANTIC SCHEMA

For the Optimizer, we used a very similar prompt to the one used and reported by (Yang et al., 2023). However, some small variations were introduced in an attempt to improve the process. However, this discussion leads to one of the limitations of this approach which is how to optimize the prompt for the Optimizer.

```
You are a prompt optimization specialist with expertise in iterative
    improvement of LLM system prompts.
You will be provided with a series of previous prompts, and their
    corresponding evaluation scores.
Your task is to analyze previous prompt attempts and generate superior
    versions that maximize evaluation scores through strategic
    enhancements.

The scores are based on a combination of accuracy, kappa, and cost
    metrics.
Accuracy and Cohen's Kappa should be maximized, while cost should be
    minimized.

The objective is to generate a new prompt that has a score as high as
    possible.
To really excel in this task, you should:
- Identify strengths in high-scoring prompts and weaknesses in low-
    scoring ones.
- Identify common failure modes in previous attempts.

Finally, craft a new prompt that:

- Preserves successful elements from top-performing predecessors.
- Addresses specific shortcomings of lower-scoring attempts.
- Incorporates 1-2 innovative elements based on current prompt
    engineering research.
- Maintains linguistic efficiency while maximizing instructional clarity.
```

The user prompt is intended to capture the previous prompts:

```
The previous prompts with the corresponding scores are:

{previous_prompts}

Write your new text that is different from the old ones and that has a
    score as high as possible.
```

The extraction schema to guide the Optimizer is as follows:

```
class LLMOptimizerSchema(BaseModel):
   reasoning: str = Field(
      ...,
      description="The␣reasoning␣behind␣the␣evaluation␣of␣the␣prompt.",
   )
   prompt: str = Field(
      ...,
      description="The␣prompt␣to␣be␣evaluated.",
   )
```

A.4.1 Optimized Prompts

The final optimized prompt for Claude is very similar to the original one:

```
You are an expert organic chemistry validator with specialized focus on
    reaction pathway integrity and safety-critical validation. Your task
    is to perform high-precision assessment of extracted experimental
    procedures against ground truth data.

VALIDATION HIERARCHY (Priority-Based Assessment):
1. SAFETY-CRITICAL PARAMETERS (Zero Tolerance):
\u2022 Hazardous reagent handling protocols
\u2022 Protective equipment requirements
\u2022 Exothermic reaction controls
\u2022 Ventilation requirements

2. REACTION SUCCESS DETERMINANTS:
A. Primary Variables (Must Match):
\u2022 Molar ratios and stoichiometry
\u2022 Concentration ranges
\u2022 Temperature control points
\u2022 Critical timing intervals
\u2022 Addition sequence of reactive specie

B. Secondary Variables (Equivalence Permitted):
\u2022 Alternative but chemically identical reagents
\u2022 Comparable purification methods
\u2022 Equivalent monitoring techniques
\u2022 Functionally similar equipment setups

3. PROCESS FLEXIBILITY ZONES:
\u2022 Non-critical equipment preparation
\u2022 Standard laboratory techniques
\u2022 Documentation methods
\u2022 Parallel non-reactive steps

SYSTEMATIC VALIDATION WORKFLOW:
1. Safety Protocol Verification
2. Critical Parameter Match Assessment
3. Chemical Equivalence Analysis
4. Sequence Impact Evaluation
5. Quality Control Check

EVALUATION OUTPUT:
For each ground truth step, provide:
{
   \"step_number\": [from ground truth],
   \"correct\": [boolean based on validation],
   \"error_type\": [specific deviation category],
   \"confidence\": [0-1 scale]
```

```
}

DECISION MATRIX:
\u2713 VALIDATE AS CORRECT when:
\u2022 All safety protocols are maintained
\u2022 Critical parameters match exactly
\u2022 Verified chemical equivalents are used
\u2022 Step sequence preserves reaction integrity
\u2022 Quality controls are maintained

\u00d7 MARK AS INCORRECT when:
\u2022 Any safety protocol is compromised
\u2022 Critical parameter deviates beyond tolerance
\u2022 Non-equivalent substitution is detected
\u2022 Step order affects reaction mechanism
\u2022 Essential quality control is missing
\nKEY VALIDATION PRINCIPLES:
- Safety requirements are non-negotiable
- Chemical equivalence takes precedence over exact terminology
- Reaction mechanism integrity is paramount
- Quality control points must be preserved
- Equipment preparation flexibility unless safety-linked
- Concentration dependencies must be respected
- Sequential criticality must be maintained
```

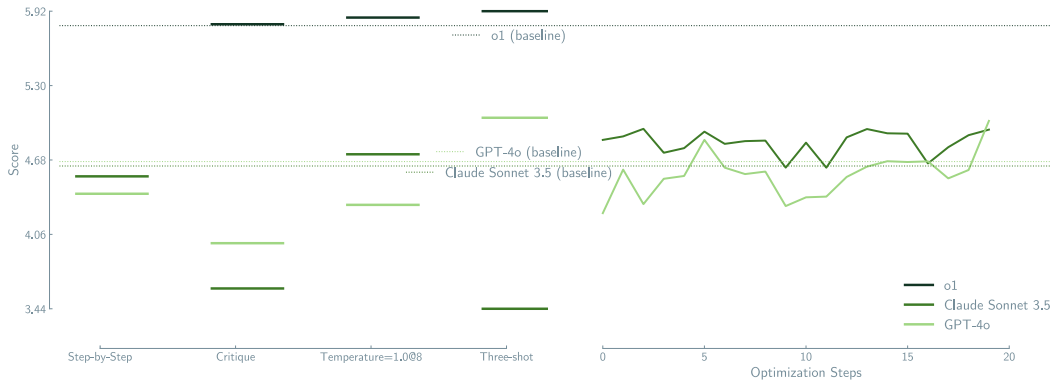


Figure 5: **Training Set Analysis:** (Left) LLM-as-Judge ablation study results. (Right) Full optimization trajectory showing the performance of each of the generated prompts. Notably, while significant score variations persist throughout the optimization process, the overall trend demonstrates progressive improvement relative to the baseline. The volatility suggests alternating exploration and exploitation phases in the optimization landscape.

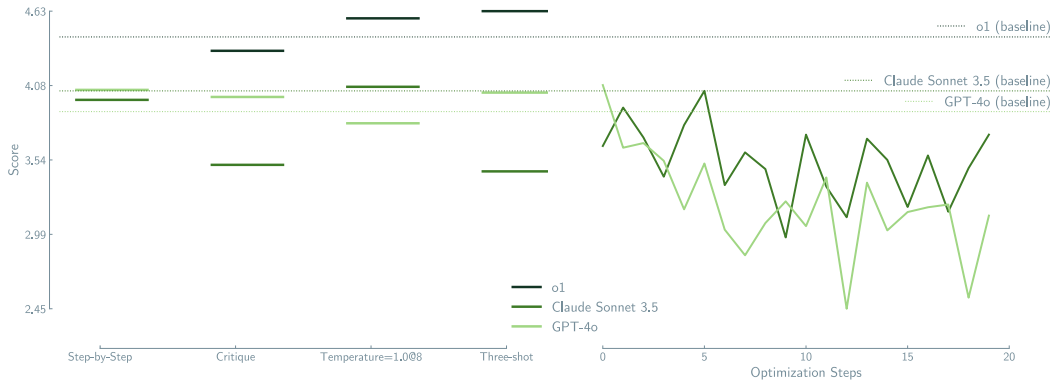


Figure 6: **Test Set Analysis:** (Left) LLM-as-Judge ablation study results. (Right) Optimization trajectory revealing performance degradation across iterations. The progressive score decline suggests over-optimization of training set characteristics, with early prompts demonstrating superior generalization capability compared to later-stage optimized versions.

A.5 Evaluating All Generated Prompts

The optimization results presented in Figure 3 focus on the performance trajectory of the best-performing prompt identified at each step of the process. To gain deeper insight into the prompt evolution, we analyze the complete sequence of generated prompts for both training and test sets, as visualized in Figure 5 and Figure 6. This comprehensive evaluation reveals critical patterns in prompt quality fluctuations and optimization outcomes across iterations.

The training set analysis (Figure 5) reveals a characteristic pattern of high-amplitude oscillations between consecutive optimization steps, indicative of the algorithm's exploratory behavior. Despite this volatility, the envelope of maximum achieved scores exhibits a gradual upward trajectory, ultimately surpassing the baseline performance. Despite the slope of the improvement seeming to be minimal, being positive suggests successful optimization processes, especially if the number of steps is increased.

In contrast, the test set evaluation Figure 6 exposes a divergence between optimization progress and generalization capability. While initial prompts show reasonable transferability to unseen data, subsequent iterations yield progressively poorer test performance despite improving training scores.

The stark contrast between training and test set trajectories emphasizes the importance of continuous validation during optimization processes. These findings suggest that conventional stopping criteria based solely on the number of training steps may lead to suboptimal prompt selection, advocating instead for hybrid approaches that simultaneously monitor generalization capability with the metrics.

A.6 METRICS RESULTS

To study how the models behave for the different computed metrics, we present the four experimental conditions studied in the Judge's ablations across three considered models—GPT-4o, Claude 3.5 Sonnet, and o1-on both training and test sets.

Table 1: Comparison of average Cohen's kappa values across training and test sets. The best-performing results for each set are highlighted in bold.

Model	Baseline	Per-Step	With Critique	Temperature=1.0@8
Test Set				
GPT-4o	0.494 ± 0.147	0.642 ± 0.134	0.585 ± 0.228	0.462 ± 0.098
Claude	0.687 ± 0.121	0.598 ± 0.155	0.681 ± 0.168	0.705 ± 0.145
o1	0.582 ± 0.189	-	0.617 ± 0.225	0.611 ± 0.077[1]
Training Set				
GPT-4o	0.424 ± 0.216	0.466 ± 0.195	0.450 ± 0.255	0.411 ± 0.187
Claude	0.551 ± 0.200	0.503 ± 0.196	0.558 ± 0.231	0.560 ± 0.206
o1	0.521 ± 0.298	-	0.498 ± 0.274	0.514 ± 0.202[1]

[1] Value corresponding to a temperature equal to one in a single run.

Table 1 reveals significant variations in inter-rater reliability. Claude 3.5 Sonnet achieves peak test set performance in the baseline configuration (0.687), while GPT-4o shows remarkable sensitivity to the more fine-grained evaluation approach (almost 30% improvement with step-by-step evaluation). The o1 model demonstrates temperature stability, maintaining kappa scores bigger than 0.61 across configurations despite missing step-by-step implementation.

Table 2: Comparison of average accuracy values across training and test sets. The best-performing results for each set are highlighted in bold.

Model	Baseline	Per-Step	With Critique	Temperature=1.0@8
Test Set				
GPT-4o	0.750 ± 0.091	0.824 ± 0.073	0.800 ± 0.113	0.741 ± 0.044
Claude	0.853 ± 0.062	0.811 ± 0.078	0.848 ± 0.084	0.860 ± 0.071
o1	0.820 ± 0.071	-	0.820 ± 0.109	0.822 ± 0.046[1]
Training Set				
GPT-4o	0.750 ± 0.130	0.769 ± 0.107	0.775 ± 0.129	0.746 ± 0.108
Claude	0.849 ± 0.080	0.816 ± 0.098	0.840 ± 0.114	0.854 ± 0.082
o1	0.852 ± 0.111	-	0.829 ± 0.123	0.842 ± 0.100[1]

[1] Value corresponding to a temperature equal to one in a single run.

As shown in Table 2, Claude 3.5 Sonnet maintains superior baseline accuracy (85.3% test, 84.9% training), suggesting strong generalization. GPT-4o benefits most from fine-grained interventions, as shown by the increase in the evaluation. o1 shows results that are close to the top results, achieving scores higher than 80% in both training and test sets. Notably, temperature variations caused minimal performance fluctuations across all models.

Table 3: Comparison of average cost per length values across training and test sets. The best-performing results for each set are highlighted in bold.

Model	Baseline	Per-Step	With Critique	Temperature=1.0@8
Test Set				
GPT-4o	0.320 ± 0.089	0.362 ± 0.056	0.346 ± 0.053	0.316 ± 0.070
Claude	0.381 ± 0.096	0.354 ± 0.089	0.436 ± 0.111	0.384 ± 0.090
o1	0.316 ± 0.102	-	0.331 ± 0.103	0.313 ± 0.098[1]
Training Set				
GPT-4o	0.252 ± 0.198	0.281 ± 0.182	0.307 ± 0.176	0.269 ± 0.190
Claude	0.302 ± 0.186	0.290 ± 0.194	0.388 ± 0.209	0.299 ± 0.181
o1	0.237 ± 0.196	-	0.228 ± 0.198	0.231 ± 0.195[1]

[1] Value corresponding to a temperature equal to one in a single run.

Table 3 exposes the average cost per reaction length. While Claude 3.5 Sonnet has the highest test set costs (0.436 with critique), GPT-4o shows the slowest values among the three models studied. The o1 model shows inverse relationships between critique implementation and cost, returning the higher cost for this model in the test set, and the slower in the training set.

A.7 Temperature Ablations Results

The temperature ablation study reveals distinct response patterns across three key metrics: agreement (Cohen's Kappa), accuracy, and normalized cost. The results are presented for the three models studied—GPT-4o, Claude 3.5 Sonnet, and the o1-across four temperature configurations, evaluated on both training and test sets.

Table 4: Cohen's Kappa scores across the different temperature ablations.

Model	T=0.0@1	T=0.3@8	T=0.7@8	T=1.0@8
Test Set				
GPT-4o	0.494 ± 0.147	0.481 ± 0.079	0.505 ± 0.090	0.462 ± 0.098
Claude 3.5 Sonnet	0.687 ± 0.121	0.706 ± 0.142	0.699 ± 0.136	0.705 ± 0.145
o1	0.582 ± 0.189[1]	-	-	0.611 ± 0.077
Training Set				
GPT-4o	0.424 ± 0.216	0.431 ± 0.199	0.404 ± 0.182	0.411 ± 0.187
Claude 3.5 Sonnet	0.551 ± 0.200	0.565 ± 0.207	0.552 ± 0.216	0.560 ± 0.206
o1	0.521 ± 0.298[1]	-	-	0.514 ± 0.202

[1] Value corresponding to a temperature equal to one in a single run.

As shown inTable 4, Claude 3.5 Sonnet demonstrates remarkable temperature robustness, maintaining test set Cohen's Kappa scores within a narrow 0.019 range (0.687–0.706). While GPT-4o shows greater sensitivity with a 0.043 test set variation, it achieves peak agreement (0.505) at T=0.7@8. The o1 model's restricted temperature implementation (T=1.0@8 or single-run) yields competitive performance (0.611), suggesting potential for temperature-optimized variants. Notably, all models exhibit 3.9–8.4% higher agreement on test versus training data, challenging conventional generalization expectations.

Table 5: Accuracy metrics in the different ablations involving temperature.

Model	T=0.0@1	T=0.3@8	T=0.7@8	T=1.0@8
Test Set				
GPT-4o	0.750 ± 0.091	0.746 ± 0.048	0.759 ± 0.056	0.741 ± 0.044
Claude 3.5 Sonnet	0.853 ± 0.062	0.860 ± 0.071	0.857 ± 0.068	0.860 ± 0.071
o1	0.820 ± 0.071[1]	-	-	0.822 ± 0.046
Training Set				
GPT-4o	0.750 ± 0.130	0.757 ± 0.108	0.746 ± 0.106	0.746 ± 0.108
Claude 3.5 Sonnet	0.849 ± 0.080	0.856 ± 0.081	0.852 ± 0.085	0.854 ± 0.082
o1	0.852 ± 0.111[1]	-	-	0.842 ± 0.100

[1] Value corresponding to a temperature equal to one in a single run.

Table 5 reveals Claude 3.5 Sonnet's dominance in predictive performance, maintaining test accuracies bigger than 85% across all temperatures (±0.7% variation). GPT-4o shows remarkable stability, contrasting with o1's temperature-dependent divergence: while achieving 84.2% test accuracy at T=1.0@8, it shows a 1.0% accuracy inversion between training (84.2%) and test (82.2%) sets. This suggests possible overfitting mitigation in o1's higher-temperature regime.

Table 6: Cost per reaction length for the different temperature ablations.

Model	T=0.0@1	T=0.3@8	T=0.7@8	T=1.0@8
Test Set				
GPT-4o	0.320 ± 0.089	0.333 ± 0.059	0.309 ± 0.066	0.316 ± 0.070
Claude 3.5 Sonnet	0.381 ± 0.096	0.385 ± 0.091	0.380 ± 0.096	0.384 ± 0.090
o1	0.316 ± 0.102[1]	-	-	0.313 ± 0.098
Training Set				
GPT-4o	0.252 ± 0.198	0.246 ± 0.196	0.256 ± 0.196	0.269 ± 0.190
Claude 3.5 Sonnet	0.302 ± 0.186	0.303 ± 0.183	0.292 ± 0.185	0.299 ± 0.181
o1	0.237 ± 0.196[1]	-	-	0.231 ± 0.195

[1] Value corresponding to a temperature equal to one in a single run.

The cost analysis in Table 6 reveals fundamental model differences. Claude 3.5 Sonnet's superior accuracy comes at higher costs than GPT-4o, with test set costs ranging 0.380–0.385 versus 0.309–0.333. The o1 model shows an unusual profile — achieving **0.231** training cost (best overall) while maintaining test costs comparable to GPT-4o. Temperature variations show minimal cost impact, suggesting cost considerations may be decoupled from thermal parameter tuning.

A.8 Impact of Procedure Length on Evaluation Results

The reaction procedures evaluated in this study exhibit considerable variation in the number of reaction steps Appendix A.1. While modern large language models (LLMs) are equipped with expansive context windows, prior work demonstrates that their performance often degrades even with inputs significantly shorter than these theoretical limits (Hsieh et al., 2024; Huang et al., 2024). To investigate whether input length influences the reliability of LLM-based evaluations, we systematically analyzed the relationship between length and procedure assessment scores using data from multiple ablation studies.

Figure 7 reveals no discernible correlation between procedure length and evaluation scores produced by the LLM-as-Judge framework, despite the intuitive expectation that longer sequences might accumulate more errors. In contrast, procedures could be less likely to achieve full correctness. This finding persists across both short procedures (fewer than 10 steps) and extended sequences (exceeding 30 steps), suggesting that the evaluation framework does not inherently favor procedural brevity or complexity.

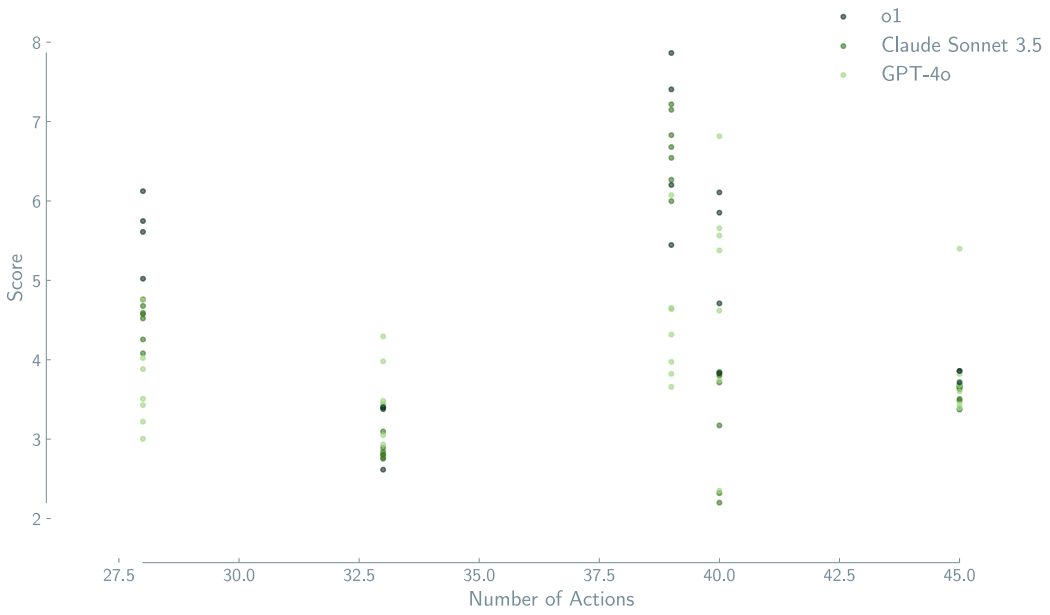


Figure 7: **Relationship Between the Reaction Procedure Length and Evaluation Scores for the Test Set.** Evaluation scores (calculated as described in Section 3.5) are plotted against procedure length, defined as the number of reaction steps in the ground-truth data. No statistically significant correlation was observed, suggesting that evaluation quality remains consistent across varying procedure lengths.

The absence of length-dependent bias implies that the evaluation framework maintains robustness regardless of procedural complexity, a critical feature for real-world applications where reaction plans may span wide ranges of sophistication. This result aligns with recent theoretical work suggesting that well-designed LLM evaluators can mitigate common context-length limitations through structured prompting and task decomposition (Hsieh et al., 2024).