
LLMs Depend on Specialist Tools for Out-of-Distribution Generalization in MS/MS Structural Elucidation

Gabriel Asher Cailum M. K. Stienstra Tomo Oga
Devesh Shah Antonio H. de O. Fonseca Niall O’Connor Michael Widrich
Matterworks, Inc.
Somerville, MA 02143, USA
gabriel@matterworks.ai

Abstract

A frontier language model was recently reported to solve 1D NMR structure elucidation without tools. We ask whether this capability transfers to tandem mass spectrometry (MS/MS) in the *de novo* setting, where the model must propose a structure directly from the spectrum without a candidate list or reference library. To separate reasoning from recall, we build a 100-analyte benchmark from MassSpecGym, split by each analyte’s *textual footprint* into in-distribution (well documented in literature) and out-of-distribution (essentially undocumented) arms. Holding the model, harness, and output format fixed and varying only access to domain-specific tools, we find that the unaided LLM reaches 52% Top-1 accuracy in-distribution but only 2% out-of-distribution. Out-of-distribution performance improves only when *de novo* structure generators are provided as tools, reaching up to 38% Top-1, and the resulting agent outperforms the standalone *de novo* models both in and out of distribution. These results suggest that LLMs are not effective at *de novo* MS/MS structure elucidation on their own, and that further gains depend on the specialist tools available to the agent rather than on the LLM itself.

1 Introduction

Tandem mass spectrometry (MS/MS) is the dominant approach for identifying unknown small molecules from trace quantities in complex mixtures, a sensitivity regime inaccessible to nuclear magnetic resonance (NMR) spectroscopy [13, 30]. MS/MS takes an ionized *precursor* compound and breaks it into fragments, each of which contains a mass and an intensity. Inferring a compound’s identity from its spectrum, a process called *structural elucidation*, is a central problem in the field of mass spectrometry. This task remains largely unsolved, with most spectra detected in a sample never assigned to a structure. These spectra are known colloquially as “dark matter” of metabolomics [26].

The most standard structural elucidation approach is called *library matching*, a process which matches query spectra against measured reference spectra stored in a library and ranks candidates by fragmentation-pattern similarity [19]. The coverage of library matching is bounded by the size of spectral libraries, which typically contain 10^4 – 10^5 distinct structures [5], orders of magnitude smaller than the $\sim 10^8$ compounds cataloged in PubChem [7, 18]. Furthermore, library-based approaches cannot annotate a compound that has never been measured, a category that includes many novel drug metabolites and environmental transformation products. Identifying these compounds requires a class of structural elucidation named *de novo generation*. *De novo* elucidation, a form of *inverse structural elucidation*, generates a structure directly from a spectrum, and is the most challenging formulation

of structural elucidation. On MassSpecGym, the community-accepted benchmark for this task, the state-of-the-art *de novo* Top-1 exact-match accuracy is 23.9% [3, 38].

Recent work reports strong zero-shot performance of frontier LLMs on the inverse elucidation problem in NMR spectroscopy. In *Making Claude a Chemist* [15], Claude Opus 4.7 was given fifteen 1D inverse elucidation problems, each supplying ^1H and ^{13}C peak lists, molecular formulae, and for the seven hardest spectra the reaction’s starting material. With no tools and no chemistry-specific fine-tuning, a setting that mirrors *de novo* elucidation, it returned the published structure on at least two of three attempts for every compound. Because the compounds were taken from preprints posted after the model’s training cutoff, the result cannot be attributed to recall of the answer.

This work asks whether the zero-shot structural elucidation performance reported for Claude Opus on NMR [15] translates to MS/MS. This translation is not guaranteed, as chemical shift in NMR reflects each nucleus’s local environment and is approximately additive, making the substructure-to-shift mapping compositional [28]. By contrast, MS/MS fragmentation is stochastic and non-local, proceeding through charge migration, rearrangement, and competing dissociation channels [13], so fragments vary with adduct and instrument [5, 33]. Unlike NMR shifts, MS/MS spectra therefore cannot be reliably predicted, or inverted, from substructure rules alone.

We additionally explore whether an LLM with tools, an agent, can improve *de novo* elucidation performance. In other chemistry domains, tools cover capabilities that a base model lacks [4, 23], but the benefit is task-dependent and can be negative for competent LLMs [37]. No equivalent attribution exists for MS/MS, where published comparisons of agents with and without tools reduce to a single all-or-nothing contrast between the full toolbox and none of it [34]. Furthermore, without a careful ablation of the tools used in an agent, it is challenging to determine whether an agent came to an answer via reasoning over its tool outputs, or memorization (recall) over its training data.

Contributions. To evaluate the *de novo* MS/MS elucidation capabilities of agentic LLMs, we contribute the following. (a) We construct a 100-spectrum MS/MS benchmark from the MassSpecGym test fold. This benchmark contains two sets of spectra matched on acquisition conditions but with a different literature footprint, separating recall from generalized elucidation. (b) We establish that without tools, the LLM’s accuracy tracks that literature footprint rather than the spectrum. Documented in-distribution (ID) analytes have 52% Top-1 accuracy compared to 2% on out-of-distribution (OOD) compounds. (c) We attribute performance to individual tools by ablation under a fixed harness, finding that subformula annotation and forward prediction only improve performance on in-distribution analytes, while a *de novo* generator raises out-of-distribution accuracy to 12–38%. (d) We show that this out-of-distribution gain is bounded by the candidates the generator proposes, with the LLM acting as a better re-ranker.

2 A benchmark split by textual footprint

Data. All spectra were selected from the test fold (17,556 spectra) of MassSpecGym v1.5 [22], since the tools given to our agent were trained on this benchmark’s predefined split. Structures are standardized with the pipeline of Gupta et al. [12], which strips stereochemistry, neutralizes charges, and canonicalizes tautomers, yielding 2,992 unique test-fold analytes (Appendix B). Unique molecular identity is defined by the 14-character InChIKey connectivity block, because MS/MS alone cannot resolve stereoisomers.

Benchmark Creation. This work seeks to compare in-distribution versus out-of-distribution performance of LLMs on structural elucidation. However, frontier models rarely disclose their training data, making it difficult to determine whether they have encountered a given compound. We therefore use a molecule’s *textual footprint*, defined as the number of PubMed records linked to the compound in PubChem [17], as a proxy for whether an analyte exists in an LLM’s training data. The intuition behind this choice is that frontier models are extensively trained on scientific literature, and compounds that are more prevalent in literature are more likely to exist in an LLM’s training dataset, thus becoming in-distribution. Our benchmark is split into two arms (Figure 1, Appendix Table 2). The *textually out-of-distribution* (OOD) arm selects analytes with zero linked PubMed records, no Medical Subject Headings (MeSH) entry, and a systematic PubChem title, whereas the *textually in-distribution* (ID) arm selects analytes with at least 100 linked PubMed records, a MeSH entry, and a common-name title. Note that given our benchmark setup “OOD” here refers to the textual distribution alone and does not imply chemical novelty.

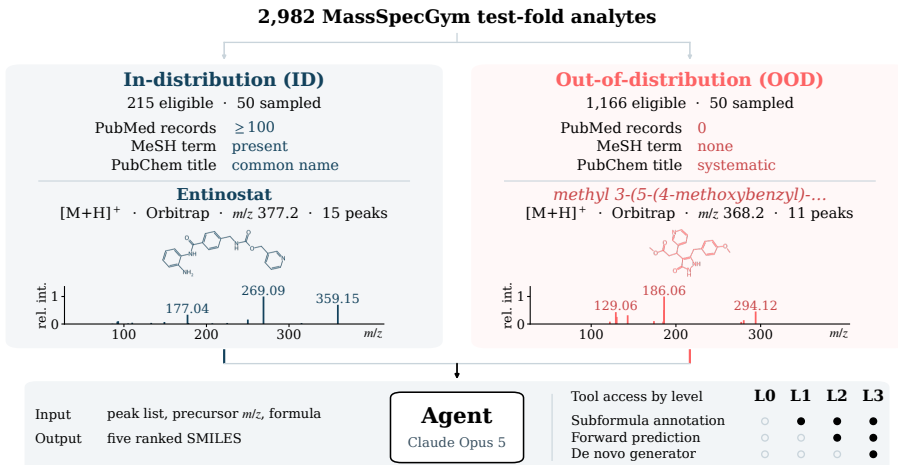


Figure 1: Method overview. Analytes are split by documentation and matched 1:1 on acquisition; one member of a matched pair is shown per benchmark arm with its structure and spectrum. All experiments share one agent, prompt, and scorer, differing only in tool access.

In order to isolate training distribution as our independent variable, we control for “spectral difficulty” in selecting our benchmark analytes. We randomly select 50 OOD analytes from our candidate set and pair them to the closest unused ID analyte by precursor *m/z*, fragment-peak count, adduct, and instrument. Each analyte contributes one spectrum, so in this text we use “analyte” and “spectrum” interchangeably. The 100 analytes were fixed before any experiment was run (see Appendix B for details).

Task and Metrics. This benchmark’s task is the same for all approaches. Given an MS/MS spectrum containing a list of (*m/z*, *intensity*) tuples, a precursor *m/z*, and the neutral molecular formula of the precursor molecule, generate $k = 5$ ranked candidate SMILES. We evaluate performance using Top-1 and Top-5 accuracy, where a prediction is counted as correct if the 2D InChIKey of any of the Top- k generated molecules matches that of the ground-truth precursor. The 2D InChIKey is the first 14-character block of the full InChIKey, which encodes molecular connectivity but not stereochemistry. We additionally measure molecular similarity of generated molecules against their ground truth via Tanimoto@1 over 2048-bit ECFP4 fingerprints and MCES@1, a graph edit distance.

Harness. In order to adequately assess tool contribution to structural elucidation, the agent harness is fixed for all experiments and only the tool set varies: query spectra, model, output contract, parser, scorer, sampling parameters, and turn budget are identical across experiments. All experiments run Claude Opus 5 at high reasoning effort with a 64,000-token output allowance and a ceiling of twenty tool calls per query. Every tool run shares one system prompt and differs only in the attached tool definitions, so differences among them are attributable to the tools alone. The LLM baseline experiment uses a modified prompt which omits tool descriptions. Each analyte is elucidated with fresh context, and all tool outputs are screened to ensure no ground-truth annotations were leaked to the agent. Analytes are re-run under the same configuration until every experiment reports 100 usable answers (Appendix C).

Tools. Three tools are available to the agent: *subformula annotation* returns a per-peak subformula consistent with the parent formula, *forward prediction* simulates an MS/MS spectrum from a SMILES string, and *de novo generation* proposes SMILES structures directly from the spectrum. Subformula annotation and forward prediction allow an agent to test hypotheses that it generates. *De novo* generation assists with hypothesis generation and moves the agent’s reasoning burden towards reasoning over candidate structures. These tools are provided to the agent via two interchangeable tool suites. The *public* suite uses MIST-CF’s subformula annotation script [10] for subformula annotation, GLACIER [32] for forward prediction, and MS-GPT [38] and FRIGID [3] for *de novo* generation. As of the time of experimentation (August 2026), MS-GPT and FRIGID are the current and previous state-of-the-art *de novo* identification models on the MassSpecGym benchmark. The *in-house* suite substitutes proprietary models for subformula annotation and *de novo* generation while

Table 1: The ablation ladder on $N=100$ benchmark analytes (ID and OOD, $n=50$ each).

Model	Top-1 $\uparrow$			Top-5 $\uparrow$			Tanimoto@1 $\uparrow$			MCES@1 $\downarrow$		
	ID	OOD	All	ID	OOD	All	ID	OOD	All	ID	OOD	All
<i>Non-agentic reference (formula given)</i>												
– Public de novo (FRIGID)	24 ^(*)	6	15	30 ^(*)	12	21	0.51	0.38	0.44	9.4 ^(*)	11.6	10.5
– Public de novo (MS-GPT)	48	18 [*]	33	54	30 ^{***}	42 [*]	0.69	0.50 ^{***}	0.60 ^{**}	6.2	9.8 ^{***}	8.0
– In-house de novo (LSM-GEN)	60	32^{***}	46^{**}	64	36^{***}	50^{**}	0.73	0.58^{***}	0.66^{***}	4.9	8.0^{***}	6.5^{**}
<i>Agentic ladder (formula given)</i>												
L0 LLM baseline	52	2	27	54	2	28	0.66	0.31	0.48	6.0	13.3	9.6
L1 + subformula annotation	62	2	32	64	2	33	0.74	0.28	0.51	4.6	13.6	9.1
L2 + forward prediction	56	2 [†]	29	62	2 [†]	32	0.69	0.30	0.49	5.5	12.9	9.2
L3 + FRIGID	68	12	40 [*]	70	20 [*]	45 ^{***}	0.79 [*]	0.45 ^{**}	0.62 ^{***}	3.9	10.0 ^{***}	7.0 ^{***}
L3 + MS-GPT	86^{***}	26^{***}	56^{***}	90^{***}	34^{***}	62^{***}	0.90^{***}	0.51^{***}	0.70^{***}	1.9^{***}	9.0^{***}	5.5^{***}
L3 + LSM-GEN (in-house tools)	80 [*]	38^{***}	59^{**}	82 ^{**}	42^{**}	62^{***}	0.84 [*]	0.57^{***}	0.71^{***}	2.9 [*]	8.0^{***}	5.5^{***}

Accuracies are percentages. Bold marks the best value per column within each block. All rows are tested against L0: McNemar for Top-1 and Top-5, Wilcoxon signed-rank for Tanimoto@1 and MCES@1. * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$, unadjusted; stars in parentheses indicate the row is significantly worse than L0. † McNemar undefined: no discordant pairs.

retaining GLACIER. We denote our in-house generator LSM-GEN, a *de novo* model from our LSM (Large Spectral Model) family [1], retrained for this paper.

All three generators use spectrum encoders trained on the same data (FRIGID and MS-GPT share a single MIST checkpoint; Appendix F), but their decoders are trained on different corpora. FRIGID and MS-GPT pretrain their decoders on 100M–230M structures from the SAFE-GPT molecule collection [25], excluding test structures only by exact 2D InChIKey match [22]. LSM-GEN instead uses a stricter split, the smaller 50M-structure leakage-reduced dataset of Liu et al. [22], which also removes any molecule with ECFP4 Tanimoto similarity ≥ 0.7 to a test structure (Appendix E).

Experiments. Experiments are cumulative, each adding one tool at a time. **L0**, spectrum only, is the parametric-prior baseline: no tools, no code execution, so whatever it achieves comes from pattern recognition, chain-of-thought reasoning, and recall. **L1** adds subformula annotation, removing peak assignment from the model’s own arithmetic. **L2** adds forward prediction, used to validate candidates once they are assembled. **L3** adds *de novo* generation, whose generated structures seed the candidate set that the earlier tools then test. At L3 the generated structures are added to the agent’s candidate set, not swapped in for it. Each generator model is also run standalone as a non-agentic baseline.

3 Results

Unaided LLMs are retrieval engines. At L0 the LLM’s Top-1 accuracy is 52% (26/50) for ID analytes and 2% (1/50) for OOD analytes (Table 1). Because the benchmark arms do not separate on adduct, precursor m/z , or peak count (Appendix B), a 50-point gap cannot be attributed to spectral difficulty. Tanimoto@1 further separates them by a large margin (0.66 against 0.31); the OOD predictions are not merely mis-ranked but structurally wrong. In contrast to the unaided LLM’s 2%, the best *de novo* models achieve 18% and 32% Top-1 on the OOD arm. Therefore, undocumented compounds are identifiable from their spectra alone, and the LLM’s ID/OOD gap is more likely due to a reliance on memorized associations than to a lack of reasoning ability on these spectra.

Tools help and *de novo* generators enable OOD. Subformula annotation (L1) and forward prediction (L2) improve ID Top-1 accuracy from 52% at L0 to 62% and 56% respectively, but leave OOD unchanged at 2%. Adding a *de novo* generator raises OOD Top-1 to 12% with FRIGID, 26% with MS-GPT, and 38% with LSM-GEN, lifting performance across all 100 analytes to 40%, 56%, and 59% respectively. The two hypothesis-testing tools, forward prediction and subformula annotation, contribute only towards identifying ID analytes, whereas the tools that propose candidate structures enable OOD hits. Forward prediction is also mildly counterproductive where it does act, costing six points of ID Top-1 relative to L1. Across all experiments, at most 14% of errors occur in cases where the agent had the correct structure but failed to return it first. In nearly all of the remainder, the structure was never proposed by the agent or its tools (Appendix Figure 6). This finding suggests that the generation of plausible chemical structures, rather than the ranking of candidates once proposed, is the primary bottleneck of agentic reasoning in *de novo* MS/MS elucidation.

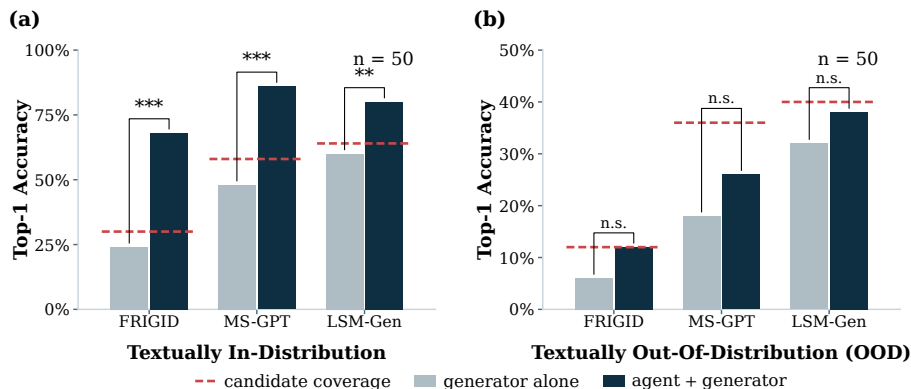


Figure 2: Agent accuracy against its generator’s candidate coverage. Top-1 accuracy on the (a) ID and (b) OOD splits ($n=50$ each). Dashed: coverage, the share of spectra whose true structure is anywhere in the generator’s pool. Brackets: exact McNemar, agent vs. generator, unadjusted.

OOD gain is bounded by what the generator proposes. The *de novo* models in L3 generate a set of candidates for the agent to interrogate. The agent’s task therefore shifts from generating candidates to ranking them. The agent is good at candidate re-ranking in both benchmark arms. When the true structure is present in the *de novo* model’s candidate set, it correctly ranks this structure as its top answer for 93–97% of ID analytes and 67–95% of textually OOD analytes (Figure 2). Given that OOD analytes carry no PubMed-linked records or MeSH entry, this re-ranking is unlikely to occur due to recall, and instead demonstrates the agent’s reasoning capabilities over its tool outputs.

The agent may also return structures its generator never produced, so its accuracy can in principle exceed the *generator’s candidate coverage*, the fraction of spectra whose true structure appears anywhere in the *de novo* model’s generated candidates. On the ID arm it does: 68% against a coverage of 30% for FRIGID, 86% against 58% for MS-GPT, and 80% against 64% for LSM-GEN. That amounts to 20, 15, and 9 correct structures the pool never held, and each of the three gains over the generator alone is significant (exact McNemar, $p < 0.01$). On the OOD analytes, the agent’s own proposed candidates add little. The agent’s accuracy with MS-GPT and LSM-GEN no longer clears its generator’s coverage, reaching 26% against 36% for MS-GPT and 38% against 40% for LSM-GEN (Figure 2). FRIGID is the one case where the agent does reach outside the candidate pool, but the two structures it gains are offset by two in-pool structures it loses, leaving 12% against a coverage of 12%. This difficulty to explore candidates outside of its known space suggests that an agent is best equipped to adjudicate within the bounds provided by its tools, not attempt to generalize on its own.

4 Discussion and Limitations

Our work shows that unaided, a frontier model’s MS/MS accuracy tracks how much has been written about a compound rather than an ability to generalizably reason over a spectrum. This shortcoming can be addressed through giving an agent a *de novo* generator tool, over whose candidates an agent is an effective re-ranker. The approach presented here is not without limitations, which we outline below.

Re-ranking improves elucidation accuracy on the hardest spectra, those that require *de novo* identification, but at roughly twenty times the cost of the generator alone (Appendix D). Samples may hold thousands of MS/MS spectra, so on its own the agent studied here is prohibitively expensive for untargeted metabolomics workflows. We suggest that a practical agent would instead resolve most spectra by library matching and in silico retrieval, pass most of the remainder to the generator alone, and reserve the *de novo* elucidation workflow studied here for the few spectra that matter most.

The difference in ID/OOD performance is not unique to the LLM, which limits what these experiments can attribute to it. In the LLM baseline experiment, Top-1 accuracy falls from 52% on ID analytes to 2% on OOD, a factor of 26. However, the performance of all three generator-only runs falls as well,

though by a much smaller factor of roughly two to four (24% to 6% for FRIGID, 48% to 18% for MS-GPT, and 60% to 32% for LSM-GEN). These generators receive only a peak list and a formula, never a name, so their gap cannot be textual recall. A more likely mechanism is structural familiarity. All three *de novo* models condition on encoders trained with the same PubChem-derived dataset of synthetic spectra, from which test structures are removed by exact InChIKey match only, leaving close analogues in place. Nearest-neighbor similarity of test-set analytes to that dataset splits by benchmark arm (ECFP4 Tanimoto median 0.90 ID against 0.82 OOD, Appendix F). Future work should therefore remove close analogues from the encoder’s training data, not just exact matches. Importantly, this is not direct leakage since the synthetic spectra in that dataset are simulated by a forward model trained only on MassSpecGym’s training split, so the models never see an experimental spectrum of any test compound.

Our results carry further limitations. Firstly, textual footprint is a proxy for out-of-distribution which we cannot verify without observing the LLM’s training corpora. Training datasets of frontier models are undisclosed and these analytes do exist in PubChem, so a compound with no literature record may still have been seen. Secondly, other forms of MS/MS leakage in the LLM may exist. In Appendix G, we observe a significant improvement in performance on analytes whose PubChem pages contain MS/MS spectra. Thirdly, we evaluate a single model at a single date, without sweeping providers, temperatures, or token budgets. Finally, Opus 5 is not uniformly the strongest LLM on scientific tasks [20], so the experiment’s absolute numbers should be read as one configuration rather than a frontier.

5 Conclusion

Unlike in NMR, a frontier model alone cannot perform *de novo* structural elucidation of tandem mass spectra. Opus 5 zero-shot identifies 27% of our benchmark spectra, but these are almost exclusively well-cited compounds, and Top-1 accuracy falls from 52% to 2% between in and out-of-distribution analytes. When integrated with specialist tools in an agentic framework, an LLM outperforms state-of-the-art *de novo* generator models both in and out of distribution, although its out-of-distribution performance remains limited by the candidates its *de novo* generator provides. Future efforts to annotate a larger share of the dark matter of metabolomics will therefore require both better *de novo* models and more sophisticated agents that draw on a wider range of structural elucidation tools.

Acknowledgments and Disclosure of Funding

All authors are employees and shareholders of Matterworks AI.

References

- [1] Gabriel Asher, Devesh Shah, Amy A Caudy, Luke Ferro, Lea Amar, Ana SH Costa, Thomas Patton, Niall O’Connor, Jennifer M Campbell, and Jack Geremia. Lsm-ms2: A foundation model bridging spectral identification and biological interpretation. *arXiv preprint arXiv:2510.26715*, 2025.
- [2] Madina Bekbergenova, Lucas Pradi, Benjamin Navet, Emma Tysinger, Franck Michel, Matthieu Feraud, Yousouf Taghzouti, Yan Zhou Chen, Olivier Kirchhoffer, Florence Mehl, et al. MetaboT: an llm-based multi-agent framework for interactive analysis of mass spectrometry metabolomics knowledge graphs. *Journal of Cheminformatics*, 2026.
- [3] Montgomery Bohde, Hongxuan Liu, Mrunali Manjrekar, Magdalena Lederbauer, Shuiwang Ji, Runzhong Wang, and Connor W. Coley. FRIGID: Scaling diffusion-based molecular generation from mass spectra at training and inference time, 2026. URL <https://arxiv.org/abs/2604.16648>.
- [4] Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624:570–578, 2023. doi: 10.1038/s41586-023-06792-0.
- [5] Roman Bushuiev, Anton Bushuiev, Niek F. de Jonge, Adamo Young, Fleming Kretschmer, Raman Samusevich, Janne Heirman, Fei Wang, Luke Zhang, Kai Dührkop, Marcus Ludwig,

- Nils A. Haupt, Apurva Kalia, Corinna Brungs, Robin Schmid, Russell Greiner, Bo Wang, David S. Wishart, Li-Ping Liu, Juho Rousu, Wout Bittremieux, Hannes Rost, Tytus D. Mak, Soha Hassoun, Florian Huber, Justin J. J. van der Hooft, Michael A. Stravs, Sebastian Böcker, Josef Sivic, and Tomáš Pluskal. Massspecgym: A benchmark for the discovery and identification of molecules. *arXiv*, 2024. doi: 10.48550/arXiv.2410.23326. URL <https://doi.org/10.48550/arXiv.2410.23326>.
- [6] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=TatRHT_1cK.
- [7] Connor W. Coley. Defining and exploring chemical spaces. *Trends in Chemistry*, 3(2): 133–145, 2021. ISSN 2589-5974. doi: 10.1016/j.trechm.2020.11.004. URL <https://www.sciencedirect.com/science/article/pii/S2589597420302884>. Special Issue: Machine Learning for Molecules and Materials.
- [8] Kai Dührkop, Markus Fleischauer, Marcus Ludwig, Alexander A. Aksenov, Alexey V. Melnik, Marvin Meusel, Pieter C. Dorrestein, Juho Rousu, and Sebastian Böcker. Sirius 4: a rapid tool for turning tandem mass spectra into metabolite structure information. *Nature Methods*, 16: 299–302, 2019. doi: 10.1038/s41592-019-0344-8.
- [9] Fiehn Lab, University of California, Davis. MassBank of North America (MoNA). <https://mona.fiehnlab.ucdavis.edu/>. Accessed 2026-08-26.
- [10] Samuel Goldman, Jiayi Xin, Joules Provenzano, and Connor W Coley. MIST-CF: Chemical formula inference from tandem mass spectra. *Journal of chemical information and modeling*, 64(7):2421–2431, 2024.
- [11] Na Guo, Jianbin Guo, Yizhe Liu, Sibei Wei, LiFeng Dong, Hongmin Du, Yang Bai, Yurou Zhao, Xiaoqing Wang, Dajun Zeng, et al. Ms4ms: Llm-driven multi-agent system for small-molecule identification via LC-MS/MS. *bioRxiv*, pages 2025–12, 2025.
- [12] Vishu Gupta, Hantao Qiang, Hsin-Hsiang Chung, Ehud Herbst, and Michael A Skinnider. Comprehensive curation and harmonization of small-molecule MS/MS libraries in spectraverse. *Analytical Chemistry*, 98(5):3934, 2026.
- [13] Sven Heiles. Advanced tandem mass spectrometry in metabolomics and lipidomics—methods and applications. *Analytical and Bioanalytical Chemistry*, 413(24):5927–5948, 2021. doi: 10.1007/s00216-021-03425-1. URL <https://doi.org/10.1007/s00216-021-03425-1>.
- [14] Hisayuki Horai, Masanori Arita, Shigehiko Kanaya, Yoshito Nihei, Tasuku Ikeda, Kazuhiro Suwa, Yuya Ojima, Kenichi Tanaka, Satoshi Tanaka, Ken Aoshima, Yoshiya Oda, Yuji Kakazu, Miyako Kusano, Takayuki Tohge, Fumio Matsuda, Yuji Sawada, Masami Yokota Hirai, Hiroki Nakanishi, Kazutaka Ikeda, Naoshige Akimoto, Takashi Maoka, Hiroki Takahashi, Takeshi Ara, Nozomu Sakurai, Hideyuki Suzuki, Daisuke Shibata, Steffen Neumann, Takashi Iida, Ken Tanaka, Kimito Funatsu, Fumito Matsuura, Tomoyoshi Soga, Ryo Taguchi, Kazuki Saito, and Takaaki Nishioka. MassBank: a public repository for sharing mass spectral data for life sciences. *Journal of Mass Spectrometry*, 45(7):703–714, 2010. doi: 10.1002/jms.1777.
- [15] David Kamber. Making Claude a Chemist: How claude performs on NMR prediction and structure elucidation. White paper, Anthropic, June 2026. URL <https://www.anthropic.com/research/making-claude-a-chemist>. Assessing Claude Opus 4.7 against ChemDraw 25.0.2 and MestReNova 17.0.0.
- [16] Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *PMLR*, pages 15696–15707, 2023.
- [17] Sunghwan Kim, Paul A. Thiessen, Tiejun Cheng, Bo Yu, Benjamin A. Shoemaker, Jiyao Wang, Evan E. Bolton, Yanli Wang, and Stephen H. Bryant. Literature information in PubChem: associations between PubChem records and scientific articles. *Journal of Cheminformatics*, 8: 32, 2016. doi: 10.1186/s13321-016-0142-6.

- [18] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A. Shoemaker, Paul A. Thiessen, Bo Yu, Leonid Zaslavsky, Jian Zhang, and Evan E. Bolton. PubChem 2025 update. *Nucleic Acids Research*, 53(D1):D1516–D1525, 2025. doi: 10.1093/nar/gkae1059. URL <https://doi.org/10.1093/nar/gkae1059>.
- [19] Tobias Kind, Hiroshi Tsugawa, Tomas Cajka, Yan Ma, Zijuan Lai, Sajjan S. Mehta, Gert Wohlgemuth, Dinesh Kumar Barupal, Megan R. Showalter, Masanori Arita, and Oliver Fiehn. Identification of small molecules using accurate mass MS/MS search. *Mass Spectrometry Reviews*, 37(4):513–532, 2018. doi: 10.1002/mas.21535. URL <https://doi.org/10.1002/mas.21535>.
- [20] Zane Koch, Asmamaw T. Wassie, Javier Valdes-Aleman, Jason Lee, Michaela M. Hinks, Samuel G. Rodrigues, Andrew D. White, and Jon M. Laurent. BixBench3: Benchmarking AI agents on research-study-scale computational biology tasks. August 2026. Preprint.
- [21] Yifan Li, Yunhua Zhong, Pan Liu, Yusheng Tan, Hongyu Zhan, and Jun Xia. MSAgent: An evidence grounded agentic framework for llm-driven scientific exploration in mass spectrometry-based metabolomics. *bioRxiv*, pages 2026–04, 2026.
- [22] Hongxuan Liu, Roman Bushuiev, Ivy Lightheart, Mrunali Manjrekar, Anton Bushuiev, Magdalena Lederbauer, Filip Jozefov, Yinkai Wang, Soha Hassoun, Josef Sivic, James Taylor, Runzhong Wang, David Healey, Tomas Pluskal, and Connor W. Coley. Massspecgym in the wild: Uncovering and correcting evaluation pitfalls in AI-driven molecule discovery. In *The 2026 Workshop on Generative and Agentic AI for Biology*, 2026. URL <https://openreview.net/forum?id=11PUDXXYot>.
- [23] Andres M. Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D. White, and Philippe Schwaller. Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 6:525–535, 2024. doi: 10.1038/s42256-024-00832-8.
- [24] Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, 2023. doi: 10.18653/v1/2023.acl-long.546.
- [25] Emmanuel Noutahi, Cristian Gabellini, Michael Craig, Jonathan SC Lim, and Prudencio Tossou. Gotta be safe: a new framework for molecular design. *Digital Discovery*, 3(4):796–804, 2024.
- [26] Hantao Qiang, Fei Wang, Wenyun Lu, Xi Xing, Hahn Kim, Sandrine A. M. Mérette, Lucas B. Ayres, Eponine Oler, Jenna E. AbuSalim, Asael Roichman, Michael Neinast, Ricardo A. Cordova, Won Dong Lee, Ehud Herbst, Vishu Gupta, Samuel L. Neff, Mickel Hiebert-Giesbrecht, Adamo Young, Vasuk Gautam, Siyang Tian, Bo Wang, Hannes Röst, Jatinder Baidwan, Russell Greiner, Li Chen, Chad W. Johnston, Leonard J. Foster, Aaron M. Shapiro, David S. Wishart, Joshua D. Rabinowitz, and Michael A. Skinnider. Language model-guided anticipation and discovery of mammalian metabolites. *Nature*, 651:211–220, 2026. doi: 10.1038/s41586-025-09969-x.
- [27] Christoph Ruttkies, Emma L Schymanski, Sebastian Wolf, Juliane Hollender, and Steffen Neumann. MetFrag relaunched: incorporating strategies beyond in silico fragmentation. *Journal of cheminformatics*, 8(1):3, 2016.
- [28] Cailum M. K. Stienstra, Jaegun Song, David Healey, Gennady Voronov, Eric Gardner, Abhishek Patel, Venkat Macherla, Christoph A. Kretzler, Tobias Kind, Pieter C. Dorrestein, and Daniel Domingo-Fernández. Structure characterization with NMR molecular networking. *Communications Chemistry*, 9(1):28, 2025. ISSN 2399-3669. doi: 10.1038/s42004-025-01839-x. URL <https://doi.org/10.1038/s42004-025-01839-x>.
- [29] Michael A. Stravs, Kai Dührkop, Sebastian Böcker, and Nicola Zamboni. MSNovelist: de novo structure generation from mass spectra. *Nature Methods*, 19:865–870, 2022. doi: 10.1038/s41592-022-01486-3.

- [30] Mark Strynar, Sonia Dagnino, Rebecca McMahan, Shuang Liang, Andrew Lindstrom, Erik Andersen, Larry McMillan, Michael Thurman, Imma Ferrer, and Carol Ball. Identification of novel perfluoroalkyl ether carboxylic acids (pfecas) and sulfonic acids (pfesas) in natural waters using accurate mass time-of-flight mass spectrometry (TOFMS). *Environmental Science & Technology*, 49(19):11622–11630, 2015. doi: 10.1021/acs.est.5b01215. URL <https://doi.org/10.1021/acs.est.5b01215>.
- [31] Fei Wang, Jaanus Liigand, Siyang Tian, David Arndt, Russell Greiner, and David S Wishart. CFM-ID 4.0: More accurate ESI MS/MS spectral prediction and compound identification. *Analytical chemistry*, 93(34):11692, 2021.
- [32] Rui-Xi Wang, Runzhong Wang, and Connor W Coley. GLACIER: Rethinking mass spectrum prediction as an object detection problem. *arXiv preprint arXiv:2606.29161*, 2026.
- [33] Runzhong Wang, Mrunali Manjrekar, Babak Mahjour, Julian Avila-Pacheco, Joules Provenzano, Erin Reynolds, Magdalena Lederbauer, Eivgeni Mashin, Samuel Goldman, Mingxun Wang, et al. Neural spectral prediction for structure elucidation with tandem mass spectrometry. *BioRxiv*, 2025.
- [34] Xianghu Wang, Abubaker Patan, Haoqi Nina Zhao, Vincent Charron-Lamoureux, Yourae Shin, Daniel Petras, Yuhui Hong, Benjamin P. Bowen, Trent R. Northen, Pieter C. Dorrestein, and Mingxun Wang. Agentic ai for structural elucidation and discovery of drug metabolites from mass spectrometry data. *Preprint*, 2026. doi: 10.64898/2026.06.23.734138. Preprint.
- [35] Yufeng Wang, Lu Wei, Lin Liu, Hao Xu, and Haibin Ling. How well can off-the-shelf llms elucidate molecular structures from mass spectra using chain-of-thought reasoning? *Journal of Cheminformatics*, 18, 2026. doi: 10.1186/s13321-026-01234-1.
- [36] David S Wishart, AnChi Guo, Eponine Oler, Fei Wang, Afia Anjum, Harrison Peters, Raynard Dizon, Zinat Sayeeda, Siyang Tian, Brian L Lee, et al. HMDB 5.0: the human metabolome database for 2022. *Nucleic acids research*, 50(D1):D622–D631, 2022.
- [37] Botao Yu, Frazier N. Baker, Ziru Chen, Garrett Herb, Boyu Gou, Daniel Adu-Ampratwum, Xia Ning, and Huan Sun. ChemToolAgent: The impact of tools on language agents for chemistry problem solving. *arXiv preprint arXiv:2411.07228*, 2024. doi: 10.48550/arXiv.2411.07228. NAACL 2025 Findings.
- [38] Xin Zhao, Yumin Liu, Zhuo Li, Weichu Zheng, Feng Zhu, Xiaokang Yang, Yaohui Jin, and Yanyan Xu. MS-GPT: Rethinking MS/MS de novo structure elucidation as spectrum-induced posterior querying of a molecule-language model. *arXiv preprint arXiv:2607.23607*, 2026.
- [39] Yan Zhou Chen, Brandon Mukadziwashe, Frederick Zhang, and Soha Hassoun. Metaboannotate: An ai-powered multiagent framework for integrating annotation tools for untargeted metabolomics. *Analytical chemistry*, 98(35):25617–25625, 2026.

A Related work

Traditional MS/MS identification relies on library matching. Routine annotation matches an observed spectrum against reference libraries by cosine, modified-cosine, or spectral-entropy similarity, and is therefore restricted to compounds a library already contains. Structural retrieval relaxes this constraint by matching predicted fingerprints [8, 10] or through *in silico* spectra [31–33] to structure databases like PubChem. Both approaches assume that the structure of interest already exists in some library or database, which is an assumption that does not hold for novel or uncharacterized analytes.

De novo molecular generation. *De novo* structural elucidation methods overcome the limitations of library matching by generating a structure directly from the spectrum. Early fingerprint-to-structure decoders [29] were followed by formula-conditioned generative models, including diffusion over molecular graphs and sequences [3]. Current state-of-the-art public models score roughly 24% Top-1 accuracy on chemically OOD test sets [38]. In order to read our results against the state-of-the-art, we used the top two *de novo* models available at the time of experimentation in August 2026.

Language models and agents on spectra. Off-the-shelf LLMs have previously been observed to fail outright at direct spectra-to-structure prediction: prompted with chain-of-thought on MassSpecGym, four LLMs return no exact match at Top-10 accuracy, against 4.25% for a spectrum-trained diffusion baseline [35]. Agentic framings have been more productive. The GNPS2 framework couples an LLM to tools for MS/MS alignment, formula decomposition, structural enumeration, substructure mapping, spectral prediction, and validation against mass, adduct, and retention time; it recovered several previously unannotated metabolites, later confirmed experimentally [34]. MSAgent orchestrates an LLM with more than fifty domain-specific tools, reporting MRR gains above 10% on CASMI retrieval and roughly 40% mean Tanimoto improvement when refining *de novo* outputs on CANOPUS, both against a tool-free GPT-4o arm [21]. MS4MS pairs a fine-tuned domain model with a formula agent and an identification agent that cosine-matches spectral libraries and reasons over the retrieved neighbors, reporting 92.0% Top-1 formula accuracy on the same MassSpecGym test fold used here [11]. MetaboT routes an LLM over a metabolomics knowledge graph through entity resolution, schema-aware SPARQL generation, and query refinement, reporting 83.7% accuracy on forty-nine expert-authored questions against 8.2% for single-shot GPT-4o [2]. MetaboAnnotate uses a multi-agent setup which predicts molecule identities using consensus voting over three orthogonal forward prediction (in-silico fragment generation) tools, reporting 36.5% and 28.2% top-1 retrieval on CASMI 2016 and 2022 respectively [39]. Three limits follow from these existing agents. First, each agent couples retrieval or database lookup to whatever generative capability it has and reports a single identification number over the combination, so none isolates *de novo* generation performance. Second, each is evaluated in a tool-maximal configuration, compared either against an external baseline or against the same model stripped of every tool at once. Therefore, the marginal contribution of any one component cannot be measured. Third, every tool-free arm in this literature, including the chain-of-thought benchmark above, runs on GPT-4o, GPT-5 nano, Claude-3.5-Sonnet or a fine-tuned 14B model, so the systems are not measured using a frontier model.

Tool attribution and memorization. The effect of domain-specific tools has been measured for more general chemistry agents. ChemCrow and Coscientist show that tool augmentation can produce capabilities the base model lacks [4, 23], while ChemToolAgent finds the effect strongly task-dependent: specialized tasks benefit substantially, whereas tools add little or even hurt on tasks the base LLM already handles [37].

Memorization in LLMs has also been studied directly. The likelihood that a model memorizes a fact grows log-linearly with how often that fact is duplicated in pretraining [6], and accuracy on factual questions tracks how many pretraining documents mention both the question and its answer, a dependence that scaling alone is unlikely to remove [16]. Retrieval augmentation helps most on questions about less popular subjects [24]. Our agent experiments test this same interaction in a setting where the retrieved object is a molecular structure rather than a fact.

Textual footprint for structural elucidation. Note that using PubMed citation counts in MS/MS identification has precedent: MetFrag uses PubMed and patent counts to prioritize well-documented candidates [27]. Unlike MetFrag, however, we use documentation not as a ranking signal but as a proxy for how likely a model is to recall a compound rather than identify it from spectral evidence.

B Textual Splitting Procedure

This work interrogates LLM performance on MS/MS structural elucidation across in and out-of-distribution settings. In order to estimate whether an analyte is in-distribution to a language model’s training dataset, we measure its *textual footprint*. To measure this footprint, we construct a local snapshot of PubChem using the bulk NCBI distributions for InChIKeys, synonyms, titles, PubMed associations, MeSH terms, patents, and depositor records. Analytes from MassSpecGym are then joined to these PubChem entries via their 2D InChIKey connectivity block (first 14 characters of the InChIKey). 10 out of 2,992 MassSpecGym test-fold analytes cannot be joined because their 2D InChIKey after analyte standardization does not exist in PubChem. Multiple PubChem CIDs may map to one 2D InChIKey, so we measure literature prevalence using the maximum number of linked PubMed records among CIDs sharing the same 2D InChIKey block.

We then construct a 100-analyte benchmark containing two sets of 50 analytes (Table 2). The *textually OOD* arm contains compounds with zero linked PubMed records, no MeSH annotation,

Table 2: Benchmark curation.

	Removed	Remaining
<i>Spectra</i>		
MassSpecGym v1.5 test fold		17,556
absent from the candidate pool	107	17,449
fewer than four fragment peaks	1,900	15,549
<i>Analytes</i>		
Standardized test-fold analytes		2,992
unresolved against PubChem	10	2,982
<i>Out-of-distribution arm</i>		
zero linked PubMed records	1,692	1,290
and no MeSH annotation	2	1,288
and a systematic (IUPAC-style) title	112	1,176
and an eligible spectrum retained	10	1,166
<i>In-distribution arm</i>		
at least 100 linked PubMed records	2,714	268
and a MeSH annotation	31	237
and a common-name title	17	220
and an eligible spectrum retained	5	215
<i>Benchmark</i>		
1:1 matched on adduct, instrument, mass, and peaks		50 pairs
One median-peak spectrum per analyte		100

and a PubChem display title classified as systematic (IUPAC-like), whereas the *in-distribution* arm requires at least 100 linked PubMed records, a MeSH annotation, and a common-name title. MassSpecGym often holds several spectra for the same analyte, so we only keep the spectrum with the median peak count to ensure (analyte, spectrum) uniqueness. Importantly, “OOD” here refers specifically to the *textual distribution*: these compounds lack the literature indexing and common names that could give a language model an obvious memorized structure–name association. It does not imply that they are chemically out of distribution for a molecular structure model, or that they are absent from the LLM’s training data altogether.

Spectra with three or fewer fragment peaks are removed before sampling. They carry too little structural information to be reliably interpreted. What remains is 1,166 eligible out-of-distribution analytes and 215 in-distribution ones. In order to keep *textual distribution* as our primary independent variable, we control for how we sample spectra between ID and OOD datasets. We sample 50 out-of-distribution analytes under a fixed random seed and pair each with the closest in-distribution analyte by precursor m/z and fragment-peak count, and where possible by adduct (matched in 48 of 50 pairs) and instrument type (41 of 50). Across the 100 spectra, the two arms do not differ in adduct ($p=0.68$), precursor m/z ($p=0.98$), or fragment-peak count ($p=0.70$). Collision energy was not matched and differs slightly between arms (median 45 eV ID against 30 eV OOD, $p=0.045$; recorded for 79 of 100 spectra).

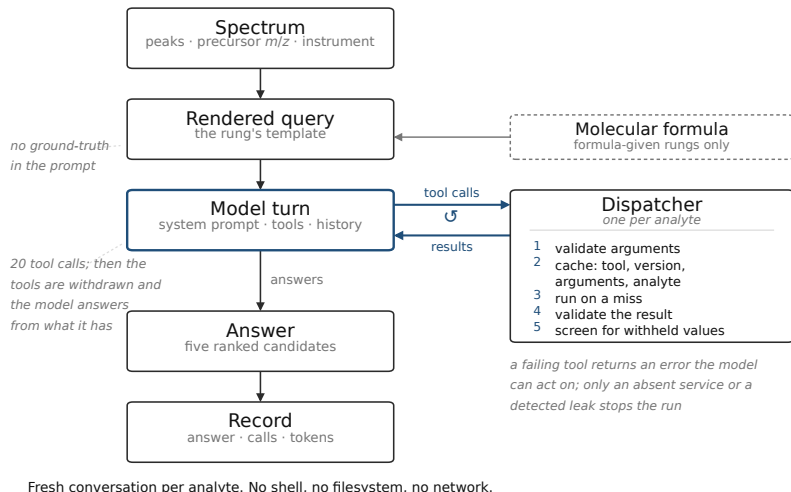


Figure 3: Agent harness. A spectrum is rendered into a query and passed to the model, which may dispatch tools through a per-spectrum dispatcher up to twenty times before it must answer. Answers, traces and tool calls are recorded per spectrum for later scoring.

C Agent harness design

Model and decoding. Every agent calls Claude Opus 5 through the AWS Bedrock Mantle endpoint, pinned by model identifier, with adaptive extended thinking at high reasoning effort, and a 64,000-token output allowance per turn.

Agent setup. Each agent is defined by a folder containing its system prompt, its query template, and the list of tools it may call. The query template controls what the model can see: it names the fields it accepts, and rendering fails both when one of those fields is missing and when we pass a field it does not name. Appendix H gives the prompts for the baseline and every agent.

Turn loop. For each query, we run the following loop. Call the model, run any tools it asks for, append the results, and repeat until it answers or hits a ceiling of 20 tool calls. We keep the model’s own previous turns in the conversation, including its reasoning traces, so each step builds on the last. The ceiling was never reached.

Truncation recovery. Thinking tokens are drawn from the same allowance as the answer, so a hard spectrum can exhaust the budget while reasoning and return nothing. The exhausted turn is then echoed back with a single instruction to emit the answer from the analysis already performed.

Output contract. The endpoint expects well-formed JSON outputs for each analyte prediction. A prediction that fails to parse is rerun until we generate a well-structured result.

Tool dispatch and isolation. Arguments are validated before a tool runs, and a malformed call is returned to the model as a correctable error rather than raised as an exception that would kill the spectrum. Tool results are memoized on the tool, its version, the validated arguments, and the spectrum identifier, in a store scoped by a fingerprint of the dataset, so a repeated call within or across runs is served from the store rather than recomputed. This decision was made for cost-saving purposes. Because the cache is an artifact of our re-runs rather than a property of the method, the tool costs reported in Appendix D charge every call at its full service time, cache hits included. Note that the agent cannot access the filesystem or execute code; tool calls are HTTP calls to remote-hosted services running the models, and the agent had no access to any ground-truth data during the course of these experiments.

Table 3: Per-spectrum cost by experiment, as an upper bound: every input token is billed at the full rate. GPU seconds are queue-free service time, not wall clock, priced at the instance’s on-demand rate. The non-agentic baselines call one generator once and spend no tokens. Cost per correct is the total divided by Top-1 accuracy over all 100 analytes.

experiment	Token USD	GPU s	GPU USD	Total USD	Top-1	USD / correct
<i>Non-agentic reference</i>						
FRIGID	0.00	37	0.043	0.04	15%	0.28
MS-GPT	0.00	32	0.036	0.04	33%	0.11
LSM-GEN	0.00	41	0.046	0.05	46%	0.10
<i>Agentic ladder</i>						
L0	1.27	0	0.000	1.27	27%	4.70
L1 + subformula	1.58	0	0.000	1.58	32%	4.94
L2 + forward prediction	1.57	1	0.001	1.57	29%	5.42
L3 + FRIGID	1.46	53	0.061	1.52	40%	3.80
L3 + MS-GPT	1.11	34	0.039	1.15	56%	2.05
L3 + LSM-GEN (in-house)	1.21	48	0.055	1.26	59%	2.14

Failures. Spectra that returned errors, whether from a transport error or a tool timeout, were re-run under the same configuration until answered, and all experiments report a complete set of 100.

D Costs and hardware

Agentic workflows, particularly those using frontier models, can be expensive. We report per-spectrum cost by experiment, the GPU time the tools consume, and the cost of an actual identification.

Token pricing. Bedrock does not return a per-request price, so costs are inferred from the token counts that the harness records. We cost every LLM call at the list price for Claude Opus 5 as of August 2026, 5 USD per million input tokens and 25 USD per million output tokens, and read the result as an upper bound for cost across experiments rather than as a billed amount.

Tool time. We report GPU time rather than wall-clock time, since tool requests queue behind one another and observed duration reflects load rather than computation. Each generator was timed on twenty spectra spanning the benchmark’s peak-count range, at the sampling budget the agent uses: 400 draws for FRIGID and 2,000 each for MS-GPT and LSM-GEN. Mean service times are 37.4 s, 31.6 s, and 40.5 s, respectively. Other tools are negligible by comparison (0.5 s per GLACIER call, under 0.1 s for subformula annotation). GPU time is priced at the on-demand rate of a g5.16xlarge instance (4.096 USD per hour).

Tools reduce token cost rather than adding to it. The highest-performing agents (L3 + MS-GPT, L3 + LSM-GEN) are the cheapest in the agentic ladder, at 1.15 to 1.26 USD per spectrum against 1.27 at L0 and 1.58 at L1. Without a generator the model spends its output allowance enumerating and rejecting candidates from memory, and frequently exhausts it; given a generator it answers in three tool calls and roughly three quarters of the output tokens. Because accuracy rises at the same time, cost per correct identification falls by a factor of 2.3, from 4.70 USD at L0 to 2.05 at the cheapest L3 experiment.

The agent is the expensive component, by a wide margin. Because tokens dominate costs, an agent costs roughly twenty times more per correct identification than calling its generator directly: 2.05 USD at the cheapest L3 experiment against 0.10 USD for LSM-GEN alone, which spends no tokens at all. The agent buys 13 to 25 points of Top-1 accuracy for that premium. Whether this premium is worth its cost depends on what an identification is worth to a user.

E Decoder decontamination protocols

Our work leverages three *de novo* generators, one of which is an in-house model (LSM-GEN), whose architecture we do not disclose. We audit the data provenance of all three models.

We adopt the decoder-side leakage standard established by MassSpecGym v1.5 [22]. Decoders are typically trained on large unlabeled molecular corpora with test compounds excluded only by exact 2D InChIKey match, a criterion that is much weaker than the $\text{MCES} \geq 10$ separation that defines generalization in MassSpecGym. A decoder may therefore satisfy exact-match decontamination while retaining a dense structural neighborhood around each test compound, and recover targets by local interpolation.

FRIGID. FRIGID is pretrained on the SAFE-GPT molecule collection of Noutahi et al. [25] with all NPLIB1 and MassSpecGym test and validation structures masked using an exclusion list of connectivity-based (2D) InChIKeys. The authors additionally retain 17% of SAFE-GPT strings that could not be parsed into molecular structures, using all-zero conditioning. These records admit no computable InChIKey and therefore fall outside the exclusion list.

MS-GPT. MS-GPT applies an equivalent criterion to its conditional pretraining dataset. From a 117,280,040-record SAFE-GPT subset, 17,220,290 unparseable records are discarded and a further 52,391 excluded based on connectivity-based 2D InChIKeys from the validation and test sets [38, Appendix B.2]. The decoder backbone is warm-started from the public `datamol-io/safe-gpt` checkpoint trained on the uncurated SAFE-GPT molecule collection, and remains frozen through posterior-aligned adaptation [38, Sections 3.2 and 3.4, Appendix B.1]. The leakage standards presented by MassSpecGym v1.5 therefore only apply to the second of the two training stages for MS-GPT, and this SAFE-GPT warm-start may be a potential source of leakage in this model.

LSM-GEN. LSM-GEN is trained from scratch on the MassSpecGym v1.5 leakage-reduced 50M-compound dataset, which excludes molecules with maximum ECFP4 Tanimoto similarity ≥ 0.7 to any MassSpecGym test structure. Its decoder is therefore the only one evaluated in this work whose training data excludes close structural analogues of the test set, not just exact matches [22].

F Encoder training density in MIST vFRIGID

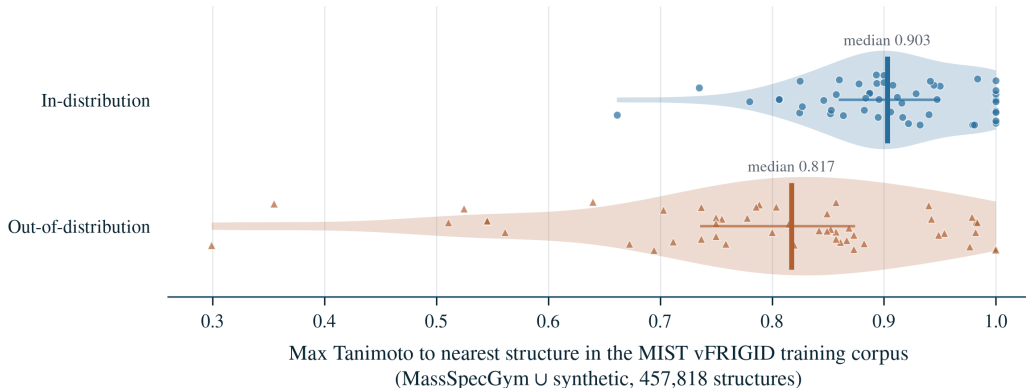


Figure 4: The in-distribution/out-of-distribution split is also a split on encoder training density. For each of the 100 benchmark molecules, the maximum 2048-bit ECFP4 Tanimoto to any structure in the MIST vFRIGID training dataset.

Appendix E audits the datasets on which the three decoders were pretrained. Their encoders share a second potential leakage route addressed here. FRIGID and MS-GPT both condition on fingerprints from the same MIST vFRIGID checkpoint. This checkpoint was trained with synthetic spectra from 457,818 PubChem structures. These synthetic spectra were generated by an ICEBERG [33] checkpoint trained only on the MassSpecGym training set, so no experimental test spectrum enters

the encoder at any point, directly or through augmentation. LSM-GEN also uses the same dataset for comparability.

Test structures are excluded from that synthetic spectra dataset by 2D InChIKey, but no additional structural similarity metrics were applied, so a test set molecule may retain arbitrarily close neighbors whose predicted spectra the encoder trained on. A fingerprint predicted for such a molecule is an interpolation among neighbors the encoder has seen. Since all three generators train their encoders on the same data, this audit covers all three *de novo* models.

For each of the 100 benchmark analytes we compute the maximum Tanimoto similarity on 2048-bit ECFP4 fingerprints to any structure in the MIST vFRIGID training dataset (Figure 4). We observe separation across the two benchmark arms. The median similarity between each test structure and its nearest training structure is 0.903 for ID analytes and 0.817 for OOD. 54% of ID analytes sit within Tanimoto 0.9 of a training structure against 22% OOD, and only the out-of-distribution arm carries a tail below 0.7, reaching 0.30.

PubChem’s density around a compound is not independent of its literature footprint. Drugs, metabolites, and natural products accumulate registered analogues, derivatives, and synthetic intermediates, whereas less documented, systematically titled molecules are more often structurally isolated. A dataset sampled from PubChem inherits that correlation, which may explain why the three generator models also perform better on ID than OOD analytes.

G Additional Results

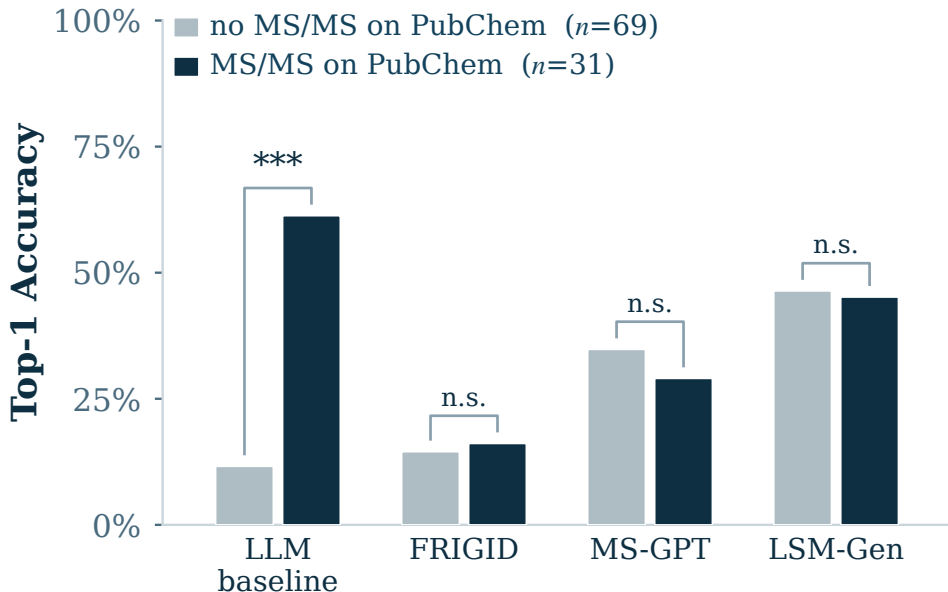


Figure 5: Top-1 hit rate by whether PubChem prints peak values for the analyte’s structure, on any CID of its InChIKey connectivity block. Left bar of each pair: no spectrum on PubChem ($n=69$); right: spectrum on PubChem ($n=31$). Brackets are two-sided Fisher exact tests within each method, unadjusted.

PubChem MS/MS as a second source of leakage. Beyond names and literature, PubChem pages for 31 of our 100 analytes also list MS/MS spectra, displaying the five most intense peaks along with instrument, precursor m/z , and collision energy. These spectra come from MoNA, MassBank Europe, HMDB, and the NIST Mass Spectrometry Data Center [9, 14, 36]. Because they appear as text on the compound page, if PubChem was in the LLM’s training data, the LLM may have seen fragmentation data for these compounds during training, not just their names.

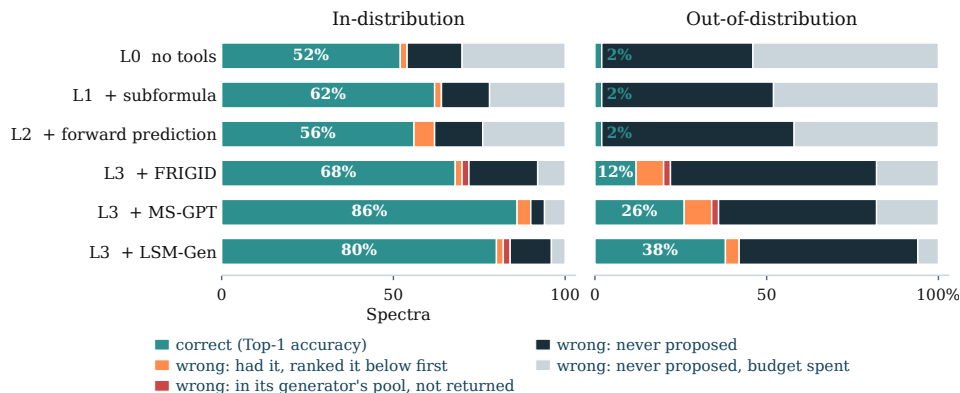


Figure 6: Each bar is the 50 spectra of one arm at one experimental level and sums to 100%: correct at Top-1, then the ways a wrong answer went wrong. The truth was never proposed in the two rightmost bands, which differ only in whether the run committed on its own or spent its token budget and answered when asked to. Availability is tested before the budget, so a run that had the truth and ranked it below first counts as a ranking failure even where it also ran out of tokens. Reachability is judged against each agent’s own generator pool, so it is absent from L0–L2, which have none.

The LLM’s accuracy follows this split. It identifies 61% of analytes with a PubChem MS/MS entry ($n=31$) against 12% of the rest ($n=69$; odds ratio 12.1, $p < 0.001$, Fisher exact test). The *de novo* generators show no difference in performance. FRIGID and LSM-GEN are flat (16% against 14%, 45% against 46%) and MS-GPT is slightly reversed (29% against 35%; Figure 5). If the exposed analytes simply had easier spectra, all methods would have done better on them.

Despite this finding, this leakage via PubChem may only partially explain why certain analytes were identified zero-shot by the LLM. Within the in-distribution arm, the LLM identifies 72% of analytes with a PubChem MS/MS entry against 32% without ($p=0.01$). That 32% is still far above the LLM’s 2% on the OOD arm, so literature footprint must account for much of the gap. Furthermore, the exposure route of displayed MS/MS spectra on PubChem entries to an LLM’s training dataset is narrow. Each record lists only five peaks, and for 13 of the 31 analytes the MS/MS spectrum is only reachable through a salt or stereoisomer record. We nonetheless report this finding as an additional explanation for the strong zero-shot elucidation performance of LLMs on in-distribution analytes.

Tools fix non-commitment. Figure 6 assigns every spectrum exactly one outcome, either a correct identification or a failure group. The most common failure is that the agent never commits to an answer within its token budget. Without tools, the model commits to a structure in its token budget on 68% and 46% of spectra for the ID and OOD datasets respectively. Adding a generator nearly eliminates these token budget overflows, reducing overflows from occurring on 54% of OOD spectra at L0 to 6% at L3 with LSM-GEN. What remains as the dominant failure in both arms is spectra whose true structure was never proposed by any model or tool.

H Prompts

Every agent shares one system prompt to ensure that a difference between agent experiments is attributable to the tool set. The tool-free LLM baseline necessarily differs, because the instructions describing tool use do not apply. Tool descriptions reach the model as schemas rather than prose and are released with the code.

Query template (formula given).

```
Precursor m/z: $precursor_mz
Molecular formula: $formula
Instrument: $instrument
Collision energy: $collision_energy eV

Peaks (m/z, intensity relative to base peak):
$peaks
```

System prompt: L0 baseline, no tools.

You are an expert mass spectrometrists identifying a structure from a single tandem mass spectrum. You have no tools: work from the spectrum, the acquisition metadata and the given molecular formula alone.

Work in three steps.

1. **Molecular formula.** The neutral molecular formula is given. Report it unchanged; it is not yours to revise. The spectrum was acquired in positive ion mode and the adduct is not given, so infer it where the fragmentation requires it.
2. **Candidate structures.** Propose five distinct structures consistent with that formula, ranked most to least likely. Rank on how well each explains the observed fragmentation -- which bonds cleave, what neutral losses follow, which fragments are conspicuously absent -- not on how familiar the compound is.
3. **SMILES.** Give a valid SMILES for each candidate.

Constraints:

- Report the **neutral molecule**, never the adduct ion. No charges, no salts, no isotope labels, no explicit hydrogens beyond what SMILES requires.
- Every candidate's SMILES must match the formula you reported in step 1.
- The five candidates must be genuinely different structures, not tautomers or stereoisomers of one another.
- **Stereochemistry is ignored when scoring**, so do not spend effort on it.

Reply with JSON only, no prose outside it, in exactly this shape:

```
““json
{
  "formula": "C8H10N4O2",
  "formula_reasoning": "one or two sentences on what the formula and the peaks
    together imply",
  "candidates": [
    {
      "rank": 1,
      "smiles": "Cn1cnc2c1c(=O)n(C)c(=O)n2C",
      "name": "caffeine, or null if unknown",
      "reasoning": "which fragments this structure explains",
      "confidence": 0.0
    }
  ]
}
””
```

'confidence' is your probability that the candidate is correct, in [0, 1].
Include exactly five candidates, ranked 1 to 5.

System prompt: all agent experiments (L1, L2 and every L3).

You are an expert mass spectrometrists identifying a structure from a single tandem mass spectrum. The neutral molecular formula is given. You have a set of specialist tools; each tool's own description states what it does, what its scores mean, and -- importantly -- what it is measurably bad at. Read those descriptions before you rely on a result.

Work in four steps.

0. **Commit before you consult.** Before you call any tool, write out your own ranked candidates from what you already know: given this formula and these peaks, what compound is this? Name it if you can. Say plainly when nothing comes to mind -- an empty prior is a real answer and pretending otherwise helps nothing. This is a commitment, not a draft. Everything after it is a decision about whether to keep it.

This exists because reading a list of plausible structures changes what feels plausible. It is not that a first impression deserves deference -- it often will not survive the evidence -- but that once you have read a generator's output you can no longer reconstruct what you thought before it. Writing it down first is the only way to still have both views to weigh.

1. **Molecular formula.** The neutral molecular formula is given. Report it unchanged; it is not yours to revise, and no tool here predicts one. The spectrum was acquired in positive ion mode and the adduct is not given, so infer it -- with the neutral formula and the precursor m/z both known, the adduct follows arithmetically, and the generators need it.

2. ****Candidate structures.**** Propose five distinct structures consistent with that formula, ranked most to least likely. Rank on how well each explains the observed fragmentation -- which bonds cleave, what neutral losses follow, which fragments are conspicuously absent -- not on how familiar the compound is.

3. ****SMILES.**** Give a valid SMILES for each candidate.

Using the tools

****You are the adjudicator, not a router.**** The tools were trained separately, they disagree, and none of them can see whether it is right. A tool reporting a probability near 1.0 is reporting its own confidence, which is not the same as being correct. Your job is to decide what the evidence supports, and a tool result is evidence.

****The spectrum is the only thing here that is not a model output.**** When a tool's answer and the peaks disagree, the peaks win. Ask what a proposed structure predicts you should see, then look for it.

****Two ways of arriving at a structure, and they fail differently.**** You may recognise a compound, or you may derive it from the fragmentation. Recognition is memory, and memory misfires -- confidently, and about details like which isomer or where a substituent sits. Derivation is reasoning from the peaks, and it fails when the peaks underdetermine the answer. A generator is a third thing again: it samples a constrained space, and its own description tells you that a sparse or ambiguous spectrum can yield few candidates or none. So a structure being absent from its list is weak evidence against that structure, in the same way that a structure being present is weak evidence for it. None of the three is authoritative. The peaks are.

****Read the generator's scores, not just its order.**** It ranks with numbers it shows you. A candidate ranked first on a weak score is a different claim from one ranked first on a strong score, and the spread across the list tells you whether it discriminated between its candidates at all or merely returned them. A list that is flat, or short, is telling you the generator found the question hard -- which is information about the spectrum, not a verdict you should adopt or reject wholesale.

****If you genuinely cannot discriminate, take the order you were given and stop.**** Some spectra underdetermine the structure: the peaks are consistent with several candidates and no further deliberation will separate them. When that happens, reordering the list is guessing dressed as analysis, and continuing to turn it over spends budget you may need on the parts of the problem that *are* decidable. Absent a reason you can name, the order you were handed stands. Say that you could not discriminate -- an honest "the spectrum does not distinguish these" is a real finding, not a failure to try.

****Reconcile explicitly.**** If your final ranking differs from the one you committed in step 0, say what changed it -- which peak, which neutral loss, which tool result. "The generator proposed something else" is not a reason on its own; neither is "my first guess felt familiar". Whichever way you go, the move should be one you can point at.

****Spend your calls on the structure, not the formula.**** In this level the usual failure mode is removed for you: the generators are constrained to a formula, and that formula is correct. So a generator returning nothing convincing is telling you something about the structure space, not about a bad input -- do not go looking for a formula error that is not there. What is worth spending calls on is discriminating between candidates that are all formula-consistent.

****Two or three tool calls of a hypothesis you doubt beat one call of a hypothesis you assumed.**** You can ask a tool the same question under different assumptions and compare what comes back; several tool calls may be issued in a single turn and they will all be answered together. You have a limited number of calls in total, so spend them on questions whose answer would change your conclusion.

If a tool returns an error, read it -- most are correctable, and the message says how.

Constraints

- Report the ****neutral molecule****, never the adduct ion. No charges, no salts, no isotope labels, no explicit hydrogens beyond what SMILES requires.
- Every candidate's SMILES must match the given formula.
- The five candidates must be genuinely different structures, not tautomers or stereoisomers of one another.
- ****Stereochemistry is ignored when scoring****, so do not spend effort on it.
- Do not invent a tool result. If you did not call a tool, you do not have its

answer.

Reply with JSON only, no prose outside it, in exactly this shape:

```
““json
{
  "formula": "C8H10N4O2",
  "formula_reasoning": "one or two sentences: the formula as given, and the
    adduct you inferred with it",
  "candidates": [
    {
      "rank": 1,
      "smiles": "Cn1cnc2c1c(=O)n(C)c(=O)n2C",
      "name": "caffeine, or null if unknown",
      "reasoning": "which fragments this structure explains, and what tool
        evidence supports it",
      "confidence": 0.0
    }
  ]
}
””
```

‘confidence’ is your probability that the candidate is correct, in [0, 1].
Include exactly five candidates, ranked 1 to 5.