SOUR: SCALING SYNTHETIC DATA FOR SMALL MOLECULE STRUCTURE ELUCIDATION VIA TRAIN- AND TEST-TIME GUIDANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Synthetic data can reduce reliance on scarce experimental annotations but naively increasing its volume does not guarantee better inverse prediction. This is pronounced for structural elucidation from tandem mass spectra, where high-quality experimental annotation is limited while the forward simulators remain imperfect. We introduce SOUR, a framework for scaling synthetic supervision at training and/or test time by tailoring it to experimental evidence and accommodating imperfect simulations. We use forward simulation to select synthetic examples relevant to experimental spectra of interest and construct virtual refinement trajectories. In training, this supervision supports spectral encoder adaptation and teaches an autoregressive generator to interpret imperfect fingerprints and revise candidate structures. At test time, observed spectra guide query-specific adaptation, while comparisons with newly simulated spectra provide feedback for iterative correction. Our empirical study supports a general strategy for synthetic-data scaling whose benefits transfer across model architectures. On public benchmarks for mass spectrum-based structure elucidation, SOUR-full achieves state-of-the-art identification accuracy, substantially outperforming prior models. These results highlight a novel strategy for how imperfect forward simulation can find useful synthetic supervision around an unlabeled observation of interest to guide model adaptation and correction both at train and test time.

1 INTRODUCTION

Scaling generative models increasingly relies on synthetic data, from language-model pretraining to instruction tuning (Gunasekar et al., 2023; Wang et al., 2023; Kang et al., 2025). It holds enormous potential in low-data regimes (Sennrich et al., 2015), which are prevalent in biology and chemistry: high-quality experimental annotations are scarce, while unpaired sequences or small-molecule structures are abundant (Dou et al., 2023; van Breugel et al., 2024). In these domains, first-principles rules can reliably map inputs to properties, but inferring or designing inputs from observed or desired outputs is much less well defined. Learned forward simulators facilitate synthetic data generation for the inverse task. As shown in Figure 1, denoting

$$y \xrightarrow{\text{Forward Simulation}} x, \; x \xrightarrow{\text{Inverse Generation}} y,$$

Hsu et al. (2022) defined y as the protein sequence and x as its 3D structure, and trained protein inverse-folding models using AlphaFold-predicted structures. Similarly, in small-molecule structural elucidation, y is the molecular structure and x is a mass spectrum or an NMR spectrum; a mass spectrum reflects the molecule's inherent fragmentation properties.

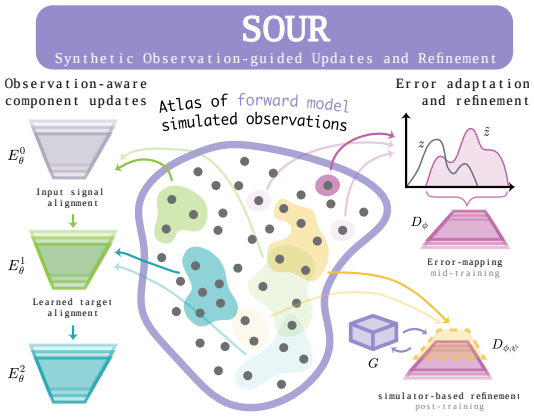

Figure 1: Overview of SOUR. We introduce a multi-stage recipe for strategically leveraging synthetic data generated with a forward model $G(y) \to \hat{x}$ to improve model components previously bottlenecked by scarce data (left), while calibrating other non-bottlenecked components to understand the error and prepare to maximally utilize forward model feedback dynamically at test time (right).

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

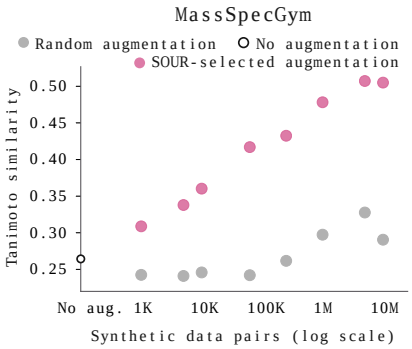

Figure 2: Query-conditioned synthetic data scales better with MIST network.

Although there is evidence that forward simulation models can learn physics-grounded principles (Wang et al., 2025a), these models inevitably introduce approximation errors and provide uneven coverage of the underlying data distribution, both of which are inherited by the synthetic examples they generate. Thus, simply increasing the amount of synthetic data does not necessarily improve performance; from Figure 2, naively scaling training with up to 10 million synthetic examples yields only marginal improvements in reconstruction performance. This observation motivates a more strategic approach to synthetic data selection, prioritizing examples that are both informative and relevant to the target data. However, no universal criterion exists for defining such relevance. In this work, we draw on the idea of test-time adaptation (Hardt & Sun, 2023), using the target data distribution itself to guide the selection of relevant synthetic examples.

We investigate this opportunity in small-molecule structure elucidation from tandem mass spectrometry (MS/MS). In MS/MS, collisions fragment a molecular ion, and the resulting fragments are recorded by their mass-to-charge ratios (m/z) and relative intensities. A mass spectrum is therefore represented as a set of peaks, each defined by an m/z value and an intensity. Recovering molecular structure from these peaks is central to identifying compounds from biological and environmental samples (Bushuiev et al., 2024; 2026). Following the $y \xrightarrow{\text{Forward Simulation}} x$ framework, forward mass spectrum simulators (Young et al., 2024; Murphy et al., 2023; Goldman et al., 2024; Wang et al., 2025b) predict mass spectra (x) from molecular structures (y). Recent approaches combining explicit fragmentation modeling with geometric deep learning have substantially improved prediction accuracy (Young et al., 2025; Wang et al., 2025a; 2026). Although these simulators remain imperfect, their increasing accuracy provides a foundation for scaling synthetic training data for inverse *de novo* generation models, which recover molecular structures (y) directly from mass spectra (x).

The dominant line of work in *de novo* generation addresses this challenge by first predicting a molecular fingerprint from a mass spectrum and then generating molecular structures conditioned on that fingerprint (Stravs et al., 2022; Bohde et al., 2025; 2026). This decomposition enables large-scale pretraining on molecule-only libraries since molecular fingerprints can be computed deterministically from structures, and recent architectures have been pretrained on up to one billion molecules (Bohde et al., 2026). As a result, the spectrum-to-fingerprint encoder has emerged as a key bottleneck in overall performance (Liu et al., 2026). Existing *de novo* models have used forward simulators to generate synthetic training data for this encoder (Bohde et al., 2025; 2026; Zhao et al., 2026). However, our analysis shows that such synthetic data cannot be scaled effectively through naive expansion alone (Figure 2), motivating more efficient and scalable strategies for synthetic-data selection and utilization.

We incorporate these strategies into **SOUR**, **S**ynthetic **O**bservation-guided **U**pdates and **R**efinement, a framework that scales simulated synthetic supervision by using experimental spectra to guide its construction. The base spectral encoder learns first from an unprocessed spectrum, retrieving the most similar spectra from a simulated PubChem atlas by cosine similarity. A second phase of encoder training then uses the improved encoder to additionally query the atlas by the new fingerprint prediction, to leverage training on structurally similar molecules. This approach refines synthetic scaling strategies through the utility to the query spectrum, and can be readily adopted by other architectures bottlenecked by experimental data. We also adapt the decoder to accommodate imperfect encoding, by first mid-training it to generate from perturbed fingerprints that mimic the omissions and spurious bits made by the synthetically-scaled encoder. Afterward, we utilize forward simulators to fragment guesses and construct virtual refinement trajectories that teach the generator to revise plausible candidates by measured spectral consistency. At test time, newly simulated feedback guides successive corrections. This extends use of synthetic data from simply helping learn relevant examples to helping learn how to adapt predictions iteratively. The state-of-the-art simulator, GLACIER (Wang et al., 2026), is adopted in this framework.

Our contributions are summarized as follows: **(1) A framework for query-conditioned synthetic-data scaling in chemical inverse problems.** We introduce SOUR, which uses test-time obser-

vations to guide synthetic supervision for model components constrained by limited experimental data. The framework works in both bulk and single-query test settings. **(2) A progressive synthetic augmentation scheme following test-time query for spectral encoders.** We introduce a two-phase strategy that progressively expands query-relevant supervision through input-side and learned-target-side synthetic data augmentation. In mass spectra, we start from the observed spectrum to first retrieve its relevant simulated examples, then secondly use its improved predictive fingerprint to incorporate structural neighbors and construct pseudo-labels for further adaptation. This spectral-side augmentation strategy transfers across encoder architectures. **(3) An autoregressive decoder trained to accommodate prediction errors and refine molecular structures.** We extend large-scale molecular pretraining with synthetic supervision that teaches the decoder to accommodate imperfect encoder predictions, and revise candidate guesses with forward-model-guided virtual refinement trajectories. At test time, forward-simulator feedback guides successive refinements of candidates. **(4) State-of-the-art MS/MS *de novo* structure elucidation accuracy.** We highlight the success of SOUR scaling strategy in MS/MS, combining query-conditioned encoder adaptation with scalable autoregressive generation and forward-model-guided refinement. SOUR-full achieves state-of-the-art accuracy on widely adopted benchmarks (MassSpecGym and NPLIB1). Its overall inference cost, including test-time synthetic-data generation and encoder training, is comparable to that of previous state-of-the-art models such as FRIGID (Bohde et al., 2026) and MS-GPT (Zhao et al., 2026).

2 Related Work

Synthetic data for scientific inverse problems. Synthetic data have become an important source of supervision for scientific inverse problems, including protein inverse folding (Hsu et al., 2022), medical imaging (Unberath et al., 2018), and molecular structure elucidation. Their utility depends not only on simulator fidelity but also on how synthetic examples are selected and incorporated into learning; poorly curated or naively mixed synthetic data can degrade performance (Gerstgrasser et al., 2024; Shi et al., 2025). Recent work has therefore explored more selective synthetic supervision, such as confidence-aware training in protein generation (Daras et al., 2025). In spectroscopy-based small-molecule elucidation, physics-based simulators and learned forward models have been used to generate synthetic MS/MS, NMR, and other spectroscopic measurements for training inverse models (Alberts et al., 2024). These approaches, however, predominantly generate synthetic examples *independently of the specific test observation*. In contrast, we use the target spectrum to guide synthetic-data selection, which we find critical for scaling synthetic supervision in mass-spectrometry-based *de novo* structure elucidation.

Test-time adaptation and feedback-guided correction. Test-time adaptation updates model parameters using information available for a specific query (Sun et al., 2020), with supervision from high-quality, query-relevant, or uncertainty-reducing examples (Hardt & Sun, 2023; Hübotter et al., 2025). Separately, test-time scaling allocates additional inference compute to feedback and revision without necessarily updating model weights (Snell et al., 2025), through iterative feedback (Madaan et al., 2023) or learned multi-turn correction (Kumar et al., 2025). In small-molecule elucidation, Mismetti et al. (2025) adapt a spectra-to-SMILES model using neighboring spectra from the *real* training set. Forward-model feedback has also been used to retain candidates with high simulated spectral consistency (Manjrekar et al., 2026) or selectively correct inconsistent substructures (Bohde et al., 2026). FRIGID is closest to our test-time reasoning approach, but uses synthetic spectra only as inference-time feedback without model updates. In contrast, we use query-specific synthetic data as supervision for test-time training, extending synthetic-data scaling from training to test-time.

***De novo* generation in MS/MS-based structure elucidation.** Since Stravs et al. (2022) introduced the spectrum–fingerprint–molecule paradigm, subsequent work has substantially improved fingerprint-to-structure generation. DiffMS (Bohde et al., 2025) demonstrated non-zero *de novo* accuracy on MassSpecGym, overcoming earlier pessimistic conclusions (Le et al., 2020), while recent models (Bohde et al., 2026; Han et al., 2025; Zhao et al., 2026) leverage pretraining on up to one billion structure–fingerprint pairs. Consequently, the spectrum-to-fingerprint encoder (Dührkop et al., 2015; Goldman et al., 2023; Kalia et al., 2025) has emerged as a primary bottleneck, with imperfect retrieval performance limiting downstream generation (Liu et al., 2026). Motivated by the long-standing success of spectral networking in MS/MS (van der Hooft et al., 2016), retrieval-based augmentation has improved forward spectrum prediction (Shahneh et al., 2024; Wang et al., 2025b).

However, exploiting query-relevant molecular neighborhoods for *de novo* elucidation, particularly in conjunction with learned forward oracles, remains comparatively underexplored.

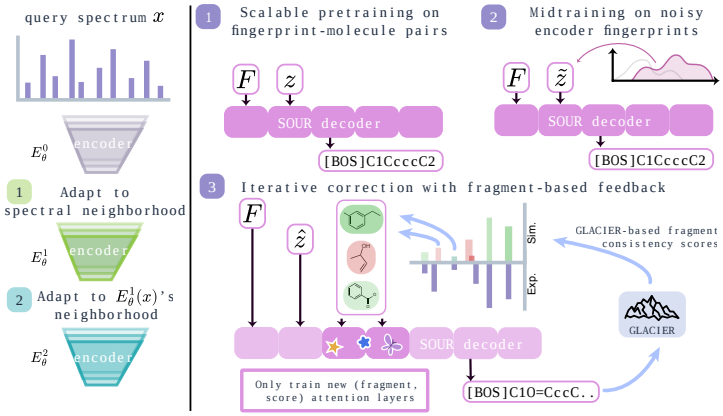

Figure 3: SOUR overview. Left: Encoder learns from the query's synthetic spectral neighborhood, followed by fingerprint-side neighborhood adaptation. Right: Decoder pre-training is only conditioned on ground-truth fingerprints and chemical formulae; mid-training learns to adapt to encoder prediction imperfections; post-training allows for GLACIER-guided iterative structural revision.

3 Scaling Synthetic Supervision for Spectral Encoding

It follows that training a model on synthetically generated data should improve spectral encoders which are bottlenecked by the relative scarcity of high-quality experimental data. Yet, training a model on simulated spectra towards the scale of the entire PubChem database only marginally improves this performance.

While this first step requires building an entire synthetic spectrum library for fast consultation of spectral matching, we posit that a query spectrum, or set thereof, can usefully prioritize which molecules are the most informative to learn from.

We first retrieve molecules by spectral agreement and train the encoder on their simulated spectra with exact fingerprints. We then combine spectral agreement with the resulting, improved fingerprint predictions to construct pseudo-labels for the experimental queries themselves.

Let $\mathcal{D}_{\text{real}}$ contain labeled training spectra, and let $\mathcal{Q} = \{x_i\}_{i=1}^n$ be a set of unlabeled queries. Each query comprises a spectrum x with molecular formula F, adduct a, and acquisition conditions u. For molecule y, $z(y) \in \{0,1\}^d$ is its radius-two Morgan fingerprint with $d = 4096$ (Rogers & Hahn, 2010); we write z when the molecule is clear from context. We use GLACIER (Wang et al., 2026) as the simulator, denoted as $G(y; a, u)$, which first performs bond-breaking predictions then predicts the intensities of each fragment predicted. While our strategy supports any choice of spectral encoder E_θ, we demonstrate the majority of our experiments on MIST (Goldman et al., 2023), an empirically scalable architecture, which produces a fingerprint $E_\theta^{\text{prob}}(x) \in [0,1]^d$, reflecting a set of per-bit predicted probabilities.

3.1 Phase 1: Spectral-side synthetic augmentation

We first filter the molecular library by precursor mass, then rank the remaining candidates by similarity between their simulated spectra and the querying spectrum. Let $\mathcal{R}(x_i)$ be the retained structures and $\mathcal{U}_i$ the simulation conditions associated with x_i, including several collision energies. Writing $\widetilde{x}_i(y,u) = G(y; a_i, u)$ with metadata $(F(y), a_i, u)$, we construct

$$\mathcal{D}_{\text{spec}}(\mathcal{Q}) = \bigcup_{i=1}^{n} \left\{ (\widetilde{x}_i(y,u), z(y)) : y \in \mathcal{R}(x_i),\ u \in \mathcal{U}_i \right\}. \tag{1}$$

Each simulated spectrum is paired with the exact fingerprint of its source molecule, while overlapping molecules and acquisition conditions allow simulations to be shared across queries. With large-scale query-based retrieval and augmentation, synthetic supervision can provide comprehensive coverage of the chemical space and fragmentation patterns relevant to the observed queries.

To prevent overfitting to the test queries and provide more robust supervision across the broader chemical space, we also simulate fingerprint neighbors of labeled training molecules and retain the real training pairs. With s indexing these three sources, training minimizes

$$\mathcal{L}_{\text{spec}}(\theta; \mathcal{Q}) = \sum_s w_s \, \mathbb{E}_{(x,z)\sim\mathcal{D}_s} \mathbb{E}_{x'\sim\mathcal{A}(x)} \big[\ell_{\text{fp}}(E_\theta^{\text{prob}}(x'), z)\big], \qquad \sum_s w_s = 1, \tag{2}$$

where $\mathcal{A}$ perturbs spectral peaks and intensities. The final fingerprint loss combines cosine and soft Tanimoto distances. For MIST, intermediate fingerprint predictions receive additional supervision (Appendix B). Source weights control how often each type of supervision is sampled.

3.2 Phase 2: Fingerprint-side synthetic augmentation

Once Phase 1 selects and augments training molecules by spectral similarity, the improved encoder provides an additional structural clue through its fingerprint predictions. Depending on the base strength of the encoder, the starting $E_{\theta^{(0)}}^{\text{prob}}(x)$ before Phase 1 augmentation may be too weak to retrieve molecules to learn a useful spectrum-fingerprint relationship. Phase 2 combines these two signals to construct fingerprint targets for the observed queries and trains the encoder directly on these experimental spectra. After Phase 1, the encoder provides a second retrieval signal: a probabilistic fingerprint prediction for the query. We rank candidates by soft Tanimoto similarity to this prediction and combine that ranking with forward spectral agreement using equal reciprocal-rank fusion (RRF): $s_i(y) = \frac{1}{\kappa + r_{\text{spec},i}(y)} + \frac{1}{\kappa + r_{\text{fp},i}(y)}$. Here $r_{\text{spec},i}$ and $r_{\text{fp},i}$ denote the spectral and fingerprint ranks. The constant κ controls how quickly each contribution decreases with rank; the two rankings receive equal weight. Among the leading fused candidates, we select a fingerprint medoid using mean pairwise fingerprint similarity and a small preference for higher fused ranks. The pseudo-label $z_i^\star$ is the selected molecule's fingerprint, so target selection accounts for similarity among the retrieved structures as well as their ranks.

This produces a second supervision set, $\mathcal{D}_{\text{fp}}(\mathcal{Q}) = \{(x_i, z_i^\star)\}_{i=1}^n$, where the shared-encoder recipe uses $z_i^\star$. We adapt the encoder by minimizing

$$\begin{aligned}\mathcal{L}_{\text{adapt}}(\theta; \mathcal{Q}) = \frac{1}{n} \sum_{i=1}^{n} & \mathbb{E}_{x_i'\sim\mathcal{A}(x_i)}[1 - T(E_\theta^{\text{prob}}(x_i'), z_i^\star)] \\ & + \lambda_{\text{real}} \, \mathbb{E}_{(x,z)\sim\mathcal{D}_{\text{real}}} \mathbb{E}_{x'\sim\mathcal{A}(x)}[1 - T(E_\theta^{\text{prob}}(x'), z)],\end{aligned} \tag{3}$$

where T is soft Tanimoto similarity. We drop peaks from each query and train the encoder to predict the same pseudo-label from the resulting partial spectra. These examples are mixed with labeled training pairs. Both phases update all encoder parameters. The queries' true fingerprints are not used to score retrieval candidates, construct pseudo-labels, or compute training losses.

3.3 Encoder architectures and test-time demonstrations

As previously noted, since our strategy admits any spectral encoder producing a molecular representation $\mathbf{h}$, we use MIST (Goldman et al., 2023) as our main spectral encoder due to its empirical scalability and denote its augmented version by SOUR-MIST. We also demonstrate our same query-conditioned adaptation strategy to augment several other spectral encoders, including MVP and FLARE (Chen & Hassoun, 2026; Chen et al., 2026) using their respective representations and training objectives; we denote these variants by SOUR-MVP and SOUR-FLARE. Appendix B.5 describes the inputs and prediction targets, as well as detailed training implementations for these architectures.

We collect relevant neighbors in a pooled manner to train one shared encoder, but this collection can also occur on a per-sample basis, $\mathcal{Q} = \{x\}$, to refine one encoder per query. Each encoder per query starts from the same pretrained initialization and requires a separate adaptation run, with no parameter transfer between queries. We denote this setting by SOUR-MIST (single-query). As we will demonstrate, both settings yield comparable performance, suggesting that our strategy remains effective without pooling across query spectra, even though pooled queries could in principle provide complementary evidence. Therefore, only to simplify the adaptation budget, we report all major results on the pooled-query adapted encoders. Appendix B specifies the initialization, data mixtures, and update budgets.

4 SYNTHETIC SUPERVISION FOR MOLECULAR GENERATION

Following recent work on scaling molecular pretraining (Bohde et al., 2026; Zhao et al., 2026), we first pretrain an autoregressive decoder on large molecule-only collections with structure-derived formulae and fingerprints. However, even though these decoders have not yet seen any synthetic data, utilizing upstream encoders that have been adapted on synthetic data can make decoding liable to the errors arising from this fuzzy adaptation. Therefore, we additionally prepare the decoder to learn from the distribution of imperfect fingerprints the encoder generates on experimental training spectra; this enables more robust decoding of experimental or synthetic inputs.

We finally post-train the decoder on virtual refinement trajectories, using forward simulation to turn discrepancies between simulated and observed spectra into supervision for structural correction.

4.1 PRETRAINING ON STRUCTURE-DERIVED CONDITIONS

We derive the formula and fingerprint directly from each training molecule, allowing pretraining to use molecule-only collections without paired experimental spectra. We represent each molecule as a SAFE sequence $y = (y_1, \ldots, y_L)$, including its end token (Noutahi et al., 2024). The SOUR decoder D_ϕ is a transformer decoder prefilled with formula and active fingerprint bits, trained with

$$\mathcal{L}_{\text{pre}}(\phi) = -\mathbb{E}_{y \sim \mathcal{D}_{\text{mol}}} \sum_{\ell=1}^{L} \log D_\phi(y_\ell \mid F(y), z(y), y_{<\ell}). \tag{4}$$

We supervise only the molecular sequence and end token, masking the conditioning prefix in loss.

4.2 MID-TRAINING ON IMPERFECT FINGERPRINTS

While decoder pretraining uses exact fingerprints, the encoder can omit active bits or predict spurious ones, thanks to spectral-fingerprint imperfect patterns introduced by synthetic supervision. We align the decoder to match the error rates of the predicted fingerprints in the molecular corpus, as estimated from encoder predictions on labeled, non-synthetic spectra. For bit j and threshold setting $r \in \{1, \ldots, R\}$, let k_{rj} and a_{rj} denote the probabilities of retaining an active bit and activating an inactive bit, respectively. We estimate these rates on training spectra and calibrate them on validation spectra. The resulting perturbation distribution is

$$Q(\widetilde{z} \mid z) = \rho\, \delta_z(\widetilde{z}) + \frac{1-\rho}{R} \sum_{r=1}^{R} \prod_{j=1}^{d} \text{Bern}(\widetilde{z}_j; z_j k_{rj} + (1 - z_j) a_{rj})\,. \tag{5}$$

We retain the exact fingerprint with probability ρ. Otherwise, we sample r once per molecule and perturb its bits independently. Keeping the molecular target unchanged, mid-training minimizes

$$\mathcal{L}_{\text{mid}}(\phi) = -\mathbb{E}_{y,\, \widetilde{z} \sim Q(\cdot \mid z(y))} \sum_{\ell=1}^{L} \log D_\phi(y_\ell \mid F(y), \widetilde{z}, y_{<\ell}). \tag{6}$$

We apply this corrective midtraining procedure by training on 100 million molecules uniformly downsampled from the SAFE-GPT dataset. Appendix C gives the calibration procedure and training settings.

4.3 POST-TRAINING ON VIRTUAL REFINEMENT TRAJECTORIES

Forward simulation lets us construct correction supervision beyond the mistakes produced by a particular decoder. For a labeled training spectrum x with structure $y^\star$, we retrieve a plausible alternative y from the fingerprint neighborhood of $y^\star$. We compare GLACIER's simulated spectrum for y with the observed spectrum to obtain feedback $e(x, y)$, and use $y^\star$ as the corrected target. Each virtual refinement trajectory contains one such supervised correction step.

GLACIER's fragment assignments map matched and unmatched predicted peaks to candidate atoms, suggesting substructures to preserve or revise. Formulae of unmatched experimental peaks record

Table 1: Formula-known retrieval on MassSpecGym. Hit rates are percentages; MCES@1 is the mean top-candidate MCES distance. Parentheses give 99.9% BCa intervals with 20,000 resamples. Baseline results are from Liu et al. (2026). See Appendix B.5 for details.

Encoder / setting	Hit rate@1 ↑	Hit rate@5 ↑	Hit rate@20 ↑	MCES@1 ↓
DeepSets	4.42 (3.92–4.97)	14.46 (13.58–15.36)	30.76 (29.67–31.93)	15.04 (14.89–15.19)
Fingerprint FFN	5.09 (4.57–5.66)	14.69 (13.83–15.56)	31.97 (30.86–33.10)	14.94 (14.79–15.09)
DeepSets + Fourier	6.56 (5.95–7.16)	16.46 (15.58–17.35)	33.46 (32.39–34.59)	14.14 (13.98–14.31)
MVP	13.96 (13.10–14.82)	36.88 (35.71–38.02)	68.12 (67.00–69.26)	10.36 (10.19–10.53)
SOUR-MVP	19.16 (18.22–20.15)	43.19 (42.00–44.37)	72.04 (70.92–73.16)	9.57 (9.39–9.74)
FLARE	22.66 (21.63–23.74)	50.00 (48.78–51.22)	75.15 (74.10–76.22)	9.00 (8.82–9.18)
SOUR-FLARE	23.74 (22.70–24.85)	49.82 (48.57–51.11)	77.23 (76.23–78.26)	8.57 (8.38–8.76)
No-aug. MIST	12.16 (11.35–12.99)	22.00 (20.98–23.02)	40.08 (38.86–41.28)	12.45 (12.29–12.62)
4m-aug. MIST	40.87 (39.63–42.07)	59.27 (58.07–60.47)	73.43 (72.30–74.46)	7.18 (7.00–7.37)
SOUR-MIST	43.04 (41.86–44.32)	64.09 (62.92–65.27)	77.74 (76.72–78.72)	5.83 (5.65–6.00)

evidence missing from the simulation. Together, the formula, predicted fingerprint $\hat{z} = E_\theta(x)$, candidate, and feedback form a prefix for autoregressive prediction of the corrected sequence:

$$\mathcal{L}_{\text{corr}}(\psi) = -\mathbb{E}_{(x,y,y^\star)} \log D_{\phi,\psi}(y^\star \mid F, \hat{z}, y, e(x,y)) \,. \tag{7}$$

To accommodate these new inputs, we introduce two appended transformer blocks and feedback-token embeddings, denoted by trainable correction parameters ψ, to the decoder architecture while freezing all other mid-trained weights. We implement an auxiliary loss that can increase likelihood for accurate correction cues over cues constructed to be faulty, such that model training is encouraged to directly discriminate genuinely helpful feedback from misleading ones. Appendix C.2 details the synthetic data and objective.

4.4 Iterative correction at test time

Given the adapted fingerprint and molecular formula, the decoder generates an initial candidate pool. We simulate selected candidates, compare their spectra with the observed query, and use the resulting feedback to generate revisions. We also simulate the revised candidates, then rank them together with the original pool using spectral agreement, fingerprint agreement, generation rank, and a dataset-specific revision term. Further rounds use the same observed spectrum to compute feedback for the current candidates. Decoder parameters remain fixed throughout. Generation and ranking settings are given in Appendix C.3.

5 Experiments

5.1 Benchmarks and evaluation setup

Following prior works, we evaluate on MassSpecGym and NPLIB1 and focus on the setting where molecular formula is provided for simplicity. We evaluate both encoder improvements, by measuring Tanimoto similarity of predictions and candidate retrieval; as well as coupled decoder improvements via *de novo* evaluation, including exact structure recovery, generated structures' fingerprint similarity, and graph edit distance (MCES). To demonstrate our encoder augmentation strategy, we focus on three different evaluation settings: (1) comparing SOUR-MIST across different base training repertoires; (2) shared SOUR-MIST training versus independent single-query adaptation; and (3) the generality of our encoder-based spectrum-side augmentation strategy with other encoders, namely MVP and FLARE. In all settings, simulators are only trained on the corresponding labeled training fold, and all generated spectra consulted exclude test structures. Further details on data splits, exclusions, and metric definitions are provided in Appendix A.

To demonstrate the strength of our error- and feedback-adapted decoder over other previous decoders, we consider a fourth evaluation setting where we fix the choice of encoder model to measure the benefit of our decoder (Table 3). To address previously identified errors in MIST's latest checkpoint (Liu et al., 2026), for two recent MIST-dependent models, FRIGID and MS-GPT, we reimplement both models using an encoder with query-independent augmentation, termed *4M/2M-aug.*

MIST (see Appendix A.4 for details). This reimplementation does not change the relative ranking on benchmarks and was confirmed by the original authors.

5.2 Spectral encoding and retrieval-evaluated architecture transfer

We observe our two-stage, query-driven alignment of the encoder model, SOUR-MIST, drastically improves the prediction objective of fingerprint reconstruction, rising to 0.527 and 0.606 Tanimoto similarity on MassSpecGym and NPLIB1, respectively (Table 2). Comparing the relative strengths of both training phases, we observe that the second, fingerprint-side augmentation phase assists SOUR-MIST improvements more in MassSpecGym due to the design of the graph-edit split. Measuring the utility of this improvement toward candidate retrieval, we observe SOUR-MIST achieves state-of-the-art retrieval on MassSpecGym, improving hit rates by 5-8% to the next best model and reducing Top 1 MCES to below 6 graph edits on average (Table 1). We also demonstrate the spectrum-side alignment benefits other spectral encoders, improving Top 1 retrieval in all cases. While the relative improvement is most pronounced for MVP at 37%, FLARE demonstrates smaller benefits. Finally, as a separate control, we apply the real-neighbor tuning strategy of Mismetti et al. (2025) to MIST, which produces almost negligible fingerprint changes compared to SOUR (Appendix D.3).

Table 2: Fingerprint Tanimoto; relative gains use paired baselines (Appendix B.4). †: 1000 representative subset; otherwise full test sets.

	NPLIB1		MassSpecGym	
Setting	Mean Tani	Δ (%)	Mean Tani	Δ (%)
No-aug. MIST	0.468		0.264	
+ SOUR † (single-query)	0.495	+5.58	0.352	+34.15
4M/2M-aug. MIST	0.533		0.433	
+ SOUR † (single-query)	0.563	+5.64	0.535	+23.88
SOUR-MIST	0.606		0.527	

Adaptation to a single query: SOUR can also adapt to an individual measurement, training an encoder with roughly 1000 neighbors from that query's neighborhood. Starting from *FP-aug.* MIST, single-query adaptation reaches comparable fingerprint Tanimoto similarities of 0.563 on NPLIB1 and 0.535 on a representative subset of the MassSpecGym test set, improving over the paired baselines by 5.64% and 23.88% (Table 2). The *no-aug.* MIST comparisons also show gains from a weaker initialization. While a single query might offer more custom retrieval evidence than a wide range of spectra, shared adaptation can pool information across related measurements and amortize total training cost. Other test settings where totals vary or compute constraints may therefore motivate the empirical prioritization of one method over another, with little tradeoff in performance.

5.3 Decoder adaptation results for *de novo* structure elucidation

We demonstrate that the improved fingerprints from SOUR-MIST benefit FRIGID, MS-GPT, and our novel molecular decoder (Table 3): Replacing *4M/2M-aug* MIST with SOUR-MIST introduces a 55% relative Top 1 exact match (Acc@1) recovery improvement for FRIGID-base on NPLIB1 and 47% relative improvement on MassSpecGym. Considering structural similarity improvements and Top 10 exact match rates, similar gains are observed for MS-GPT. Fixing our novel decoder (denoted SOUR-base to exclude test-time iterative correction), our encoder-side adaptation improves 8.42 and 6.22 percentage points to one-shot Acc@1. The generality of this improved encoder therefore provides benefits to downstream molecular generation. When we permit a round of test-time feedback and subsequent correction, SOUR-full achieves state-of-the-art exact-recovery rates among the evaluated methods: 27.35%/40.42% at Top-1/Top-10 on NPLIB1 and 17.98%/24.70% on MassSpecGym. Compared to previously introduced decoder choices paired with SOUR-MIST, SOUR-full's Top-1 gains over the strongest alternatives are 6.84 points over FRIGID on NPLIB1 and 2.82 points over MS-GPT on

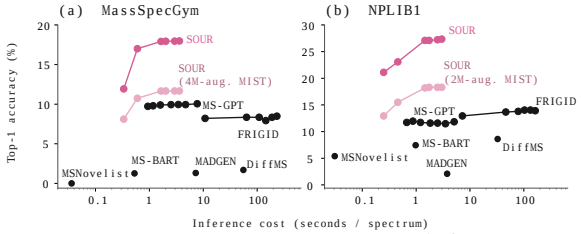

Figure 4: Top-1 structure recovery versus inference cost on MassSpecGym (a) and NPLIB1 (b). Curves vary the test-time generation and correction budgets. The two SOUR curves use *4M/2M-aug.* MIST and SOUR-MIST.

Table 3: *De novo* structure elucidation with supplied formulae. Acc@k is exact recovery (%); Tani@k is mean best fingerprint similarity. Bold and underlined entries indicate the best and second-best values. Entries marked with † are quoted from existing publications, the others are re-evaluated. SOUR-base uses one-shot generation; SOUR-full incorporates synthetic-guided correction. Appendix A.4 specifies candidate ranking and generation budgets.

	NPLIB1				MassSpecGym			
Model	Acc@1	Tani@1	Acc@10	Tani@10	Acc@1	Tani@1	Acc@10	Tani@10
Spec2Mol†	0.00	0.12	0.00	0.16	0.00	0.12	0.00	0.16
MIST + Neuraldecipher†	2.32	0.35	6.11	0.43	0.00	0.14	0.00	0.16
MIST + MSNovelist†	5.40	0.34	11.04	0.44	0.00	0.06	0.00	0.15
MADGEN†	2.10	0.22	2.39	0.27	1.31	0.20	1.54	0.26
MS-BART†	7.45	0.44	10.99	0.51	1.26	0.23	1.28	0.28
MSAnchor†	8.51	0.38	16.90	0.49	2.68	0.32	4.67	0.41
DiffMS†	8.34	0.35	15.44	0.47	2.30	0.28	4.25	0.39
FRIGID-base (w/ *4M/2M-aug.* MIST)	12.94	0.50	18.93	0.56	8.21	0.37	9.82	0.42
MS-GPT (w/ *4M/2M-aug.* MIST)	11.72	0.41	17.09	0.44	9.73	0.35	11.75	0.38
SOUR-base (w/ *4M/2M-aug.* MIST)	14.77	0.53	27.11	0.59	10.87	0.42	14.56	0.45
SOUR-full (w/ *4M/2M-aug.* MIST)	18.32	0.53	28.57	0.59	11.67	0.40	14.94	0.43
With SOUR-MIST adapted using synthetic supervision								
FRIGID-base (w/ SOUR-MIST)	20.17	0.56	27.65	0.63	12.14	0.45	14.79	0.50
FRIGID (w/ SOUR-MIST)	20.51	<u>0.58</u>	34.31	<u>0.65</u>	12.82	0.45	17.48	0.49
MS-GPT (w/ SOUR-MIST)	20.27	0.52	27.35	0.56	15.16	0.46	19.51	0.49
SOUR-base (w/ SOUR-MIST)	<u>23.20</u>	**0.61**	<u>37.36</u>	**0.67**	<u>17.09</u>	**0.52**	<u>24.06</u>	**0.56**
SOUR-full (w/ SOUR-MIST)	**27.35**	**0.61**	**40.42**	**0.67**	**17.98**	<u>0.50</u>	**24.70**	<u>0.54</u>

MassSpecGym (33.3% and 18.6% relative).One-shot SOUR-base already reaches 23.20% and 17.09% Top-1 accuracy on these benchmarks; spectral correction rounds most dramatically improve the Top-1 accuracy rate. SOUR-decoder also achieves strong inference-time efficiency. As shown in Figure 4, it establishes a new Pareto front in the performance–cost trade-off while scaling more efficiently than prior models. As noted in Table 10, even when SOUR-MIST adds more test-time synthetic-data generation and training cost compared to previous models, by amortizing to a per-test-sample basis, SOUR still achieves similar overall inference cost, outperforming diffusion-based models like FRIGID and DiffMS.

Decoder perturbation and spectral correction: We evaluate the mid-training decoder error preparation in a controlled NPLIB1 comparison and identify it substantially improves generation of the true structure in the Top 10 candidates from 27.47% to 36.14% (33% relative improvement) (Table 8).With the final SOUR-MIST encoders, one feedback-aware correction round raises Acc@1/10 from 23.20%/37.36% to 27.35%/40.42% on NPLIB1 and from 17.09%/24.06% to 17.98%/24.70% on MassSpecGym. Filtering and reranking account for part of this gain, while a second round adds little (Appendix D). Forward simulation thus helps recover structures absent from the initial pool, with diminishing returns from further correction.

6 Discussion and conclusion

SOUR demonstrates a novel, query-conditioned synthetic scaling strategy to guide the usage of relevant synthetic data, as we demonstrate with improvements on the accuracy frontier for MS/MS-based structure elucidation as a representative scientific inverse problem. Our query-conditioned augmentation can improve spectral encoding across different architectures, readily improving auxiliary molecular decoding strategies that rely on these otherwise data-constrained encoders. We further introduce a scalable autoregressive decoder that learns to adapt to imperfect predicted fingerprints and iteratively refine candidates through forward-model feedback.

Our results highlight promising gains on retrieval and *de novo* evaluation while maintaining tractable inference-time compute costs. Looking forward, we anticipate methods to build upon SOUR by further considering simulator uncertainty; and expect SOUR to readily apply to other scientific inverse problems where scarce experimental labels are a limiting factor.

AI USE STATEMENT

AI-assisted tools were used for language editing, implementation review, and evaluation scripting.

REPRODUCIBILITY STATEMENT

Appendix A describes the data and evaluation protocols. Appendices B and C specify encoder and decoder training. Appendix D reports supplementary results and computational costs. The entire codebase could be found at `https://anonymous.4open.science/r/MSReasoner-0B86`.

REFERENCES

Marvin Alberts, Oliver Schilter, Federico Zipoli, Nina Hartrampf, and Teodoro Laino. Unraveling molecular structure: A multimodal spectroscopic dataset for chemistry. *arXiv [physics.chem-ph]*, July 2024.

Montgomery Bohde, Mrunali Manjrekar, Runzhong Wang, Shuiwang Ji, and Connor W. Coley. DiffMS: Diffusion Generation of Molecules Conditioned on Mass Spectra. In *Proceedings of the International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 4737–4756, 2025. URL `https://proceedings.mlr.press/v267/bohde25a.html`.

Montgomery Bohde, Hongxuan Liu, Mrunali Manjrekar, Magdalena Lederbauer, Shuiwang Ji, Runzhong Wang, and Connor W. Coley. Scaling Diffusion-Based Molecular Generation from Mass Spectra at Training and Inference Time. *arXiv preprint arXiv:2604.16648*, 2026. URL `https://arxiv.org/abs/2604.16648`.

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 Röst, 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. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL `https://arxiv.org/abs/2410.23326`. Datasets and Benchmarks Track.

Roman Bushuiev, Anton Bushuiev, Raman Samusevich, Corinna Brungs, Josef Sivic, and Tomáš Pluskal. Self-supervised learning of molecular representations from millions of tandem mass spectra using DreaMS. *Nature Biotechnology*, 44:630–640, 2026. doi: 10.1038/s41587-025-02663-3. URL `https://doi.org/10.1038/s41587-025-02663-3`.

Yan Zhou Chen and Soha Hassoun. Learning from all views: A multiview contrastive framework for metabolite annotation. *Analytical Chemistry*, 98(9):6598–6606, 2026. doi: 10.1021/acs.analchem.5c05675. URL `https://pubs.acs.org/doi/10.1021/acs.analchem.5c05675`.

Yan Zhou Chen, Blake Rushing, and Soha Hassoun. FLARE: Fine-grained learning for alignment of spectra-molecule REpresentation enhances metabolite annotation. *bioRxiv*, 2026. doi: 10.64898/2026.01.27.702086. URL `https://www.biorxiv.org/content/10.64898/2026.01.27.702086v1`.

Giannis Daras, Jeffrey Ouyang-Zhang, Krithika Ravishankar, Constantinos Daskalakis, Adam Klivans, and Daniel Diaz. Ambient Proteins: Training diffusion models on noisy structures. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2197. URL `https://proceedings.neurips.cc/paper_files/paper/2025/hash/5e9a29833416e5be3ef3a920489e3375-Abstract-Conference.html`.

Bozheng Dou, Zailiang Zhu, Ekaterina Merkurjev, Lu Ke, Long Chen, Jian Jiang, Yueying Zhu, Jie Liu, Bengong Zhang, and Guo-Wei Wei. Machine learning methods for small data challenges in molecular science. *Chemical Reviews*, 123(13):8736–8780, 2023. doi: 10.1021/acs.chemrev.3c00189. URL `https://doi.org/10.1021/acs.chemrev.3c00189`.

Kai Dührkop, Huibin Shen, Marvin Meusel, Juho Rousu, and Sebastian Böcker. Searching molecular structure databases with tandem mass spectra using CSI:FingerID. *Proceedings of the National Academy of Sciences*, 112(41):12580–12585, 2015. doi: 10.1073/pnas.1509788112. URL https://doi.org/10.1073/pnas.1509788112.

Matthias Gerstgrasser, Rylan Schaeffer, Apratim Dey, Rafael Rafailov, Henry Sleight, John Hughes, Tomasz Korbak, Rajashree Agrawal, Dhruv Pai, Andrey Gromov, Daniel A. Roberts, Diyi Yang, David L. Donoho, and Sanmi Koyejo. Is model collapse inevitable? breaking the curse of recursion by accumulating real and synthetic data. In *Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=5B2K4LRgmz.

Samuel Goldman, Jeremy Wohlwend, Martin Stražar, Guy Haroush, Ramnik J. Xavier, and Connor W. Coley. Annotating metabolite mass spectra with domain-inspired chemical formula transformers. *Nature Machine Intelligence*, 5(9):965–979, 2023. doi: 10.1038/s42256-023-00708-3. URL https://doi.org/10.1038/s42256-023-00708-3.

Samuel Goldman, Janet Li, and Connor W. Coley. Generating Molecular Fragmentation Graphs with Autoregressive Neural Networks. *Analytical Chemistry*, 96(8):3419–3428, 2024. doi: 10.1021/acs.analchem.3c04654. URL https://doi.org/10.1021/acs.analchem.3c04654.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023. URL https://arxiv.org/abs/2306.11644.

Yang Han, Pengyu Wang, Kai Yu, Xin Chen, and Lu Chen. MS-BART: Unified modeling of mass spectra and molecules for structure elucidation. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://papers.neurips.cc/paper_files/paper/2025/hash/a1b71d48d4806ecbe5a9e19fa3f10fdc-Abstract-Conference.html.

Moritz Hardt and Yu Sun. Test-time training on nearest neighbors for large language models. *arXiv [cs.CL]*, May 2023.

Chloe Hsu, Robert Verkuil, Jason Liu, Zeming Lin, Brian Hie, Tom Sercu, Adam Lerer, and Alexander Rives. Learning inverse folding from millions of predicted structures. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 8946–8970, 2022. URL https://proceedings.mlr.press/v162/hsu22a.html.

Jonas Hübotter, Sascha Bongni, Ido Hakimi, and Andreas Krause. Efficiently Learning at Test-Time: Active Fine-Tuning of LLMs. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/ba942323c447c9bbb9d4b638eadefab9-Abstract-Conference.html.

Apurva Kalia, Yan Zhou Chen, Dilip Krishnan, and Soha Hassoun. JESTR: Joint embedding space technique for ranking candidate molecules for the annotation of untargeted metabolomics data. *Bioinformatics*, 41(7):btaf354, 2025. doi: 10.1093/bioinformatics/btaf354. URL https://doi.org/10.1093/bioinformatics/btaf354.

Feiyang Kang, Newsha Ardalani, Michael Kuchnik, Youssef Emad, Mostafa Elhoushi, Shubhabrata Sengupta, Shang-Wen Li, Ramya Raghavendra, Ruoxi Jia, and Carole-Jean Wu. Demystifying synthetic data in LLM pre-training: A systematic study of scaling laws, benefits, and pitfalls. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 10739–10758, 2025. doi: 10.18653/v1/2025.emnlp-main.544. URL https://aclanthology.org/2025.emnlp-main.544/.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D. Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, Lei M. Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal Behbahani, and Aleksandra Faust. Training Language Models to Self-Correct via Reinforcement Learning. In *International*

Conference on Learning Representations, 2025. URL https://arxiv.org/abs/2409.12917.

Tuan Le, Robin Winter, Frank Noé, and Djork-Arné Clevert. Neuraldecipher – reverse-engineering extended-connectivity fingerprints (ECFPs) to their molecular structures. *Chemical Science*, 11 (38):10378–10389, 2020. doi: 10.1039/d0sc03115a. URL https://doi.org/10.1039/d0sc03115a.

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, Tomáš Pluskal, and Connor W. Coley. MassSpecGym in the Wild: Uncovering and Correcting Evaluation Pitfalls in AI-Driven Molecule Discovery. *arXiv preprint arXiv:2606.19624*, 2026. URL https://arxiv.org/abs/2606.19624.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative Refinement with Self-Feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2303.17651.

Mrunali Manjrekar, Runzhong Wang, Samuel Goldman, Jenna C. Fromer, and Connor W. Coley. Generative Structural Elucidation from Mass Spectra as an Iterative Optimization Problem. *arXiv preprint arXiv:2602.07709*, 2026. URL https://arxiv.org/abs/2602.07709.

Laura Mismetti, Marvin Alberts, Andreas Krause, and Mara Graziani. Test-Time Tuned Language Models Enable End-to-end De Novo Molecular Structure Generation from MS/MS Spectra. *arXiv preprint arXiv:2510.23746*, 2025. URL https://arxiv.org/abs/2510.23746.

Michael Murphy, Stefanie Jegelka, Ernest Fraenkel, Tobias Kind, David Healey, and Thomas Butler. Efficiently predicting high resolution mass spectra with graph neural networks. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 25549–25562, 2023. URL https://proceedings.mlr.press/v202/murphy23a.html.

Emmanuel Noutahi, Cristian Gabellini, Michael Craig, Jonathan S. C. Lim, and Prudencio Tossou. Gotta be SAFE: A New Framework for Molecular Design. *Digital Discovery*, 3:796–804, 2024. doi: 10.1039/D4DD00019F. URL https://doi.org/10.1039/D4DD00019F.

David Rogers and Mathew Hahn. Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, 2010. doi: 10.1021/ci100050t. URL https://doi.org/10.1021/ci100050t.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. *arXiv [cs.CL]*, November 2015.

Mohammad Reza Zare Shahneh, Michael Strobel, Giovanni Andrea Vitale, Christian Geibel, Yasin El Abiead, Neha Garg, Berenike Wagner, Karl Forchhammer, Allegra Aron, Vanessa V Phelan, Daniel Petras, and Mingxun Wang. ModiFinder: Tandem mass spectral alignment enables structural modification site localization. *J. Am. Soc. Mass Spectrom.*, 35(11):2564–2578, November 2024.

Lianghe Shi, Meng Wu, Huijie Zhang, Zekai Zhang, Molei Tao, and Qing Qu. A closer look at model collapse: From a generalization-to-memorization perspective. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/39d3b980155d3d52613e412b018004a4-Abstract-Conference.html.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/1b623663fd9b874366f3ce019fdfdd44-Abstract-Conference.html.

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. URL https://doi.org/10.1038/s41592-022-01486-3.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-Time Training with Self-Supervision for Generalization under Distribution Shifts. In *Proceedings of the International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9229–9248, 2020. URL https://proceedings.mlr.press/v119/sun20b.html.

Mathias Unberath, Jan-Nico Zaech, Sing Chun Lee, Bastian Bier, Javad Fotouhi, Mehran Armand, and Nassir Navab. DeepDRR – a catalyst for machine learning in fluoroscopy-guided procedures. *arXiv [physics.med-ph]*, March 2018.

Boris van Breugel, Tennison Liu, Dino Oglic, and Mihaela van der Schaar. Synthetic data in biomedicine via generative artificial intelligence. *Nature Reviews Bioengineering*, 2:991–1004, 2024. doi: 10.1038/s44222-024-00245-7. URL https://www.nature.com/articles/s44222-024-00245-7.

Justin Johan Jozias van der Hooft, Joe Wandy, Michael P Barrett, Karl E V Burgess, and Simon Rogers. Topic modeling for untargeted substructure exploration in metabolomics. *Proc. Natl. Acad. Sci. U. S. A.*, 113(48):13738–13743, November 2016.

Rui-Xi Wang, Runzhong Wang, and Connor W. Coley. Rethinking Mass Spectrum Prediction as an Object Detection Problem. *arXiv preprint arXiv:2606.29161*, 2026. URL https://arxiv.org/abs/2606.29161.

Runzhong Wang, Mrunali Manjrekar, Babak Mahjour, Julian Avila-Pacheco, Joules Provenzano, Erin Reynolds, Magdalena Lederbauer, Eivgeni Mashin, Samuel Goldman, Mingxun Wang, Jing-Ke Weng, Desirée L. Plata, Clary B. Clish, and Connor W. Coley. Neural Spectral Prediction for Structure Elucidation with Tandem Mass Spectrometry. *bioRxiv*, 2025a. doi: 10.1101/2025.05.28.656653. URL https://doi.org/10.1101/2025.05.28.656653.

Runzhong Wang, Rui-Xi Wang, Mrunali Manjrekar, and Connor W Coley. Neural graph matching improves retrieval augmented generation in molecular machine learning. *arXiv [cs.LG]*, February 2025b.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-Instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13484–13508, 2023. doi: 10.18653/v1/2023.acl-long.754. URL https://aclanthology.org/2023.acl-long.754/.

Adamo Young, Hannes Röst, and Bo Wang. Tandem mass spectrum prediction for small molecules using graph transformers. *Nature Machine Intelligence*, 6:404–416, 2024. doi: 10.1038/s42256-024-00816-8. URL https://doi.org/10.1038/s42256-024-00816-8.

Adamo Young, Fei Wang, David S. Wishart, Bo Wang, Hannes Röst, and Russell Greiner. FraGNNet: A deep probabilistic model for mass spectrum prediction. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=UsqeHx9Mbx.

Xin Zhao, Yumin Liu, Zhuo Li, Weichu Zheng, Feng Zhu, Xiaokang Yang, Yaohui Jin, and Yanyan Xu. Rethinking MS/MS De Novo Structure Elucidation as Spectrum-Induced Posterior Querying of a Molecule-Language Model. *arXiv preprint arXiv:2607.23607*, 2026. URL https://arxiv.org/abs/2607.23607.

A Data and evaluation

A.1 Benchmarks and metrics

We use the supplied molecular formula and the official splits of MassSpecGym and NPLIB1. MassSpecGym contains 194,119 training, 19,429 validation, and 17,556 test spectra; NPLIB1 contains 9181, 401, and 819, respectively. Results are averaged over spectra, so molecules with repeated measurements retain their benchmark weighting. Our de novo evaluations keep the full test denominator and count unavailable predictions as failures.

Encoder Tanimoto compares the predicted binary fingerprint with the target fingerprint. For formula-based retrieval, Hit@k measures recovery within the first k candidates, MRR is mean reciprocal rank, and MCES@1 is the maximum common edge subgraph distance to the top candidate. For generation, Acc@k measures exact recovery and Tani@k is the highest fingerprint similarity among the first k candidates, averaged over queries. Exact recovery in our evaluations is defined by molecular connectivity, using the first 14 InChIKey characters; it does not require stereochemical recovery. Empty candidate lists receive zero accuracy and similarity. Molecular similarity uses radius-two Morgan fingerprints, with the dimensions specified below.

Retrieval confidence intervals follow the MassSpecGym 99.9% BCa bootstrap protocol: 20,000 resamples and seed 20260505. Each resampled unit is a test spectrum. These intervals describe test-spectrum variability for a fixed model; they do not capture variation across training runs or account for dependence between measurements of the same molecule. MCES uses threshold 15, the stronger bound, and a 600-second solver limit. The threshold is an algorithm setting, not a cap on reported distances.

A.2 Molecular collections and synthetic spectra

We distinguish query-independent augmentation, which samples molecules without consulting test spectra, from the two sources used by SOUR: neighbors of labeled training molecules and candidates selected by the observed queries. The query-independent MIST baselines use the refined four-million-molecule MassSpecGym collection and two million PubChem molecules for NPLIB1. We call these encoders *4M-aug.* MIST and *2M-aug.* MIST, or *4M/2M-aug.* MIST when discussing both datasets. These counts refer to molecules, not repeated training presentations.

Query-conditioned retrieval searches PubChem catalogs of approximately 104 million structures, filtering out MassSpecGym/NPLIB1 test set InChIKey-14 matched structures in advance, guaranteeing that all of the subsequent synthetic corpus are exact-match leakage-free. Candidates are initially filtered to be within ± 0.01 Da of the neutral precursor mass for MassSpecGym and ± 0.1 Da for NPLIB1. We compare simulated and experimental spectra under the same adduct and instrument conditions using cosine similarity of normalized, approximately 0.1-Da binned spectra, with the precursor region removed. We retain up to 64 candidates per MassSpecGym query and 2048 per NPLIB1 query, then simulate them at 10, 20, 30, 40, and 50 eV. Pooling repeated molecule–adduct–instrument–energy combinations yields 2,538,905 and 7,654,065 synthetic spectra, respectively.

Training-neighbor augmentation retrieves structures by fingerprint similarity to labeled training molecules. We collect neighbors one rank at a time across training molecules and deduplicate structures, yielding about one million molecules per dataset and 20,083,880 MassSpecGym or 5,551,565 NPLIB1 spectra. Augmentation for *4M-aug.* MIST and *2M-aug.* MIST contains 78,450,540 and 40 million spectra, with each molecule using two adducts, two instrument settings, and five collision energies to simulate spectra.

Decoder pretraining uses the SAFE molecular corpus, whose training split contains approximately 933 million records (Noutahi et al., 2024). Formulae, Morgan fingerprints, and target sequences are derived from the molecules. Benchmark validation and test connectivities are excluded, and invalid conversions or sequences exceeding the context limit are discarded. Correction targets come from labeled training structures; their candidate neighbors are described in Appendix C.2.

A.3 FORWARD SIMULATOR

GLACIER predicts fragments and intensities from a molecular graph, conditioned on formula, adduct, instrument, and collision energy (Wang et al., 2026). We train one simulator on each benchmark's labeled training split and keep it fixed during encoder adaptation and decoder training. Each model uses hidden width 256, five graph-encoding layers, four fragment-decoder layers, and three layers in each intensity module. The fragment decoder has 200 slots and permits up to six broken bonds per fragment. Fragment atom assignments supply the structural evidence used in correction.

Training combines cosine spectral loss with MAGMa fragment supervision weighted by 0.1. AdamW uses learning rate 4×10^{-4}, zero weight decay, and per-device batch size 32. Annotated fragments supply intensity training during the first 2500 updates; their contribution then decays by 0.9 every 2500 updates. The learning rate warms up over 2000 updates and decays by 0.9 every 5000 updates thereafter. Training runs for at most 200 epochs, with validation-loss selection and early-stopping patience five after epoch 20. NPLIB1 adds retrieval fine-tuning against formula-matched and graph-mutated decoys, using learning rate 2.7×10^{-4} and an entropy-based ranking loss. The simulator remains an imperfect predictor, so agreement with its fragments is evidence for a candidate rather than a guarantee of correctness.

A.4 BASELINE ENCODERS AND GENERATION PROTOCOLS

Our FRIGID and MS-GPT re-evaluations share the corrected MIST implementation described by Liu et al. (2026). This removes the inherited attention-mask error that made predictions depend on batch padding. We evaluate each decoder with *4M/2M-aug.* MIST and with SOUR-MIST. Published reference results are identified separately in Table 3 and retain their original evaluation conventions. All similarity uses 2048-bit ECFP4.

MS-GPT draws 100 candidates per query and uses its posterior-based ranking and formula filtering (Zhao et al., 2026). Its continuation checkpoint is selected on validation spectra, and similarity uses 2048-bit Morgan fingerprints. FRIGID-base reports the initial 128 proposals; FRIGID adds 24 ICEBERG-guided rounds, with up to 12 parents, eight masked variants per parent, and masking probability 0.75 (Bohde et al., 2026). Encoder comparisons use the two FRIGID-base rows at the same generation budget. Comparisons with corrected FRIGID concern the complete systems, including their different forward simulators.

SOUR-base and SOUR-full use the same dataset-specific molecular decoders in both encoder settings. Fingerprint prefixes from *4M-aug.* MIST and *2M-aug.* MIST use thresholds 0.1758151 and 0.2010106, respectively; prefixes from SOUR-MIST use 0.213. Generation budgets and ranking are given in Appendix C.3.

B SPECTRAL ENCODER TRAINING

B.1 REPRESENTATION AND OBJECTIVE

MIST represents a spectrum with up to ten fragment-formula peaks and a precursor token (Goldman et al., 2023). We use hidden width 1024, four peak-attention layers, 16 heads, and dropout 0.1, without the optional instrument embedding or MAGMa head. Intermediate fingerprint predictions have widths 256, 512, 1024, and 2048, followed by the 4096-bit output. Let $p = E_\theta^{\mathrm{prob}}(x)$ denote the output probabilities and $z(y)$ the exact molecular fingerprint. The binary readout is $[E_\theta(x)]_j = \mathbf{1}\{p_j \geq \tau\}$; thresholding is used for prediction, not in the training loss.

Writing C for cosine similarity and $T(p, z) = \sum_j p_j z_j / \max\{\sum_j (p_j + z_j - p_j z_j), 10^{-8}\}$ for soft Tanimoto, the spectrum-side objective is

$$\begin{aligned}\ell_{\mathrm{fp}}(E_\theta^{\mathrm{prob}}(x), z) = 1 - C(E_\theta^{\mathrm{prob}}(x), z) + 0.1[1 - T(E_\theta^{\mathrm{prob}}(x), z)] \\ + 0.4 \sum_{d' \in \{256,512,1024,2048\}} [1 - C(E_{\theta,[d']}^{\mathrm{prob}}(x), z_{[d']})].\end{aligned} \tag{8}$$

Here $z_{[d']}$ folds the target fingerprint to width d' by a bitwise OR over indices with the same remainder modulo d'. The intermediate losses are summed. Phase 2 uses only $1 - T(p, z)$, with structural pseudo-labels for queries and true fingerprints for experimental training examples.

During initial training and Phase 1, spectra are augmented with probability 0.5 by dropping each fragment with probability 0.1 and multiplying intensities by log-normal noise with log standard deviation 0.2. Intensities are renormalized, and the precursor token is retained. Phase 2 instead drops 70% of fragment peaks on every view, without intensity noise, while keeping the fingerprint target fixed.

B.2 SHARED ENCODER

The query-independent baselines train from random initialization with the settings in Table 4. Validation spectra select their EMA checkpoint and fingerprint threshold. For SOUR-MIST, Phase 1 also starts from random initialization but uses the query-conditioned, training-neighbor, and experimental sources in Table 5.

Table 4: *4M/2M-aug.* MIST training with query-independent synthetic augmentation. Batch sizes are per device, with four devices per run.

Setting	MassSpecGym	NPLIB1
Synthetic / real batch	2048 / 75	2048 / 225
Optimizer	Adam	AdamW
Peak learning rate	2×10^{-4}	2×10^{-4}
Weight decay	10^{-6}	10^{-6}
Training steps	478,850	244,150
Warmup steps	9577	4883
EMA decay	0.997497	0.999

Table 5: Shared SOUR-MIST training settings. Each phase uses four accumulation steps. Phase 1 query and training-neighbor entries are synthetic spectra; Phase 2 query entries are experimental spectra with pseudo-labels.

Setting	MassSpecGym	NPLIB1
Phase 1: synthetic / real batch	619 / 77	653 / 115
Phase 1 optimizer	AdamW	AdamW
Phase 1 steps	80,000	30,000
Phase 1 learning rate	3.75×10^{-4}	3.75×10^{-4}
Phase 1 weight decay	10^{-6}	10^{-6}
Phase 1 warmup / schedule horizon	250 / 120,000	250 / 120,000
Phase 1 EMA decay	0.9995	0.999
Phase 2 query / real batch	245 / 5	24 / 1
Phase 2 optimizer	Adam	Adam
Phase 2 steps	500	400
Phase 2 learning rate	2×10^{-5}	2×10^{-5}
Phase 2 weight decay	10^{-7}	10^{-7}
Phase 2 warmup / schedule horizon	400 / 4000	400 / 4000
Phase 2 EMA decay	0.98061	0.98061

Phase 1 runs for 80,000 updates on MassSpecGym and 30,000 on NPLIB1. Phase 2 starts from its EMA weights and runs for 500 and 400 updates, respectively. Both phases use fixed budgets and return their final EMA weights. Both phases clip gradients at norm 5. Phase 1 uses bfloat16 and Phase 2 uses float32. The second phase resets the optimizer and EMA and follows the initial portion of a 4000-step cosine schedule.

To construct pseudo-labels, we rank retrieved structures by spectral agreement and by fingerprint similarity to the Phase 1 prediction, then combine ranks by equal reciprocal-rank fusion with constant $\kappa = 60$. We retain 64 candidates and choose a medoid from the first $K \leq 10$. For candidates

$y_{i1}, \ldots, y_{iK}$ in fused order,

$$j_i^\star = \arg\max_{1 \le j \le K} \left[\frac{\sum_{\ell \neq j} T(z(y_{ij}), z(y_{i\ell}))}{\max(K-1, 1)} + \frac{0.02}{j} \right], \qquad z_i^\star = z(y_{i,j_i^\star}). \tag{9}$$

The rank term favors earlier candidates when their structural centrality is similar. Pseudo-labels are constructed once and remain fixed throughout Phase 2. Experimental training spectra provide rehearsal alongside the pseudo-labeled queries. Query structures and target fingerprints do not enter the ranking or adaptation loss.

For encoder evaluation, we calibrate a single threshold on validation spectra by matching the total predicted active-bit count to the target count. The shared SOUR-MIST thresholds are 0.1480614543 for MassSpecGym and 0.1789445579 for NPLIB1. These thresholds remain fixed on test spectra and are separate from the decoder-prefix thresholds in Appendix A.4.

B.3 Synthetic-data scaling study

Figure 2 compares query-independent random augmentation with the two-phase SOUR augmentation procedure on MassSpecGym. All models use the same corrected MIST architecture with 72,296,996 parameters and predict 4096-bit, radius-two Morgan fingerprints without chirality. Evaluation uses all 17,556 test spectra, representing 2998 molecular connectivities, and reports binary fingerprint Tanimoto averaged over spectra. For each evaluated checkpoint, a single prediction threshold is calibrated on the 19,429 validation spectra by matching the total predicted active-bit count to the target count, then held fixed on the test set.

Synthetic-data budget. The horizontal axis counts Phase 1 synthetic spectrum–fingerprint records, including both query-conditioned and training-neighbor records for SOUR. Different acquisition conditions for the same molecule count as separate records. This count excludes experimental training spectra, repeated presentations during optimization, and the experimental query–pseudo-label pairs used in Phase 2. The displayed positive budgets are 1000, 5000, 10,000, 62,500, 250,000, one million, five million, and ten million records. The no-augmentation point represents zero synthetic records and is placed separately from the logarithmic positive-count axis.

Random augmentation. We construct nested random subsets from a PubChem atlas containing 104,300,613 molecules. Before sampling, we exclude the union of MassSpecGym and NPLIB1 validation and test connectivities, identified by the first 14 InChIKey characters, together with failed records and structures incompatible with the model's atom-count or formula representation limits. Molecules are ordered by a deterministic randomized hash with seed 20260921. Each selected molecule contributes 20 spectra generated by the MassSpecGym GLACIER model: two adducts, $[\mathrm{M}+\mathrm{H}]^+$ and $[\mathrm{M}+\mathrm{Na}]^+$; two instrument settings, Orbitrap and QTOF; and five collision energies, 10, 20, 30, 40, and 50 eV. Complete molecule blocks are retained, so the largest displayed random budget contains 500,000 molecules and ten million spectrum–fingerprint pairs. This control uses the filtered full PubChem atlas rather than the refined four-million-molecule collection used by the *4M-aug.* MIST baseline elsewhere in the paper.

SOUR-selected augmentation. We form nested subsets of the query-conditioned and training-neighbor pools described in Appendix A.2. Query-conditioned records are selected using the observed test spectra, while training-neighbor records are selected using fingerprints of labeled training molecules. Within each source pool, records receive a fixed randomized ordering, with seeds 20261131 and 20261227, respectively. For a requested budget N, we allocate approximately $538N/691$ records to the query-conditioned source and the remainder to training neighbors. Once the query-conditioned pool is exhausted, all of its 2,538,905 records are retained and further growth comes from training neighbors. Thus, the five-million and ten-million budgets contain 2,461,095 and 7,461,095 training-neighbor records, respectively. The training batch proportions remain fixed across budgets even when these pool-size proportions change.

Phase 1 training. Both augmented conditions train from random initialization for 30,000 optimizer updates with seed 20260920. Each update accumulates four microbatches. A random-augmentation microbatch contains 691 synthetic records and 77 experimental training records; an

SOUR microbatch contains 538 query-conditioned synthetic records, 153 training-neighbor synthetic records, and the same 77 experimental records. Both conditions use the objective in Eq. 8, AdamW with peak learning rate 3.75×10^{-4} and weight decay 10^{-6}, 250 warmup updates, a cosine schedule with a 120,000-update horizon, gradient clipping at norm 5, bfloat16 precision, and EMA decay 0.9995. Input augmentation is applied with probability 0.5, independently dropping fragment peaks with probability 0.1 and adding log-normal intensity noise with log standard deviation 0.2. Validation is performed every 3000 updates, and the EMA checkpoint with the highest validation mean Tanimoto is selected. These scaling runs therefore use a shorter Phase 1 budget than the 80,000-update MassSpecGym encoder used for the main benchmark results.

Phase 2 for SOUR. Each SOUR point additionally applies 500 updates of the shared Phase 2 procedure, initialized from that budget's validation-selected Phase 1 EMA checkpoint. Pseudo-labels are constructed once from the complete observed test spectra using equal reciprocal-rank fusion with $\kappa = 60$ and the top-ten fingerprint medoid with rank preference $0.02/rank$. All 17,556 test queries participate in this shared adaptation. Each microbatch contains 245 query–pseudo-label pairs and five labeled experimental training pairs, with four accumulation steps per update. We use seed 20260913, Adam with learning rate 2×10^{-5} and weight decay 10^{-7}, 400 warmup updates, a 4000-update cosine-schedule horizon, EMA decay 0.9806102, and the soft Tanimoto loss. Every training view independently drops each fragment peak with probability 0.7, retains the precursor token, and uses no intensity noise. We evaluate the final EMA after 500 updates. This adds 490,000 query–pseudo-label presentations at every synthetic-data budget; these presentations are excluded from the horizontal axis. Query target fingerprints do not enter candidate scoring, pseudo-label construction, or adaptation losses.

No-augmentation reference. The reference value of 0.26434 comes from a separately trained real-only MIST model with the same architecture. It uses Adam with learning rate 5×10^{-5}, batch size 1024, and no peak dropping or intensity jitter. Within each of three confirmation runs, the checkpoint is selected by validation performance; the run with the median best validation score supplies the reference model, selected at update 5000. This point is therefore a separately selected real-only reference rather than the zero-data endpoint of the augmented training schedule.

B.4 Single-query adaptation

Single-query adaptation starts each query from the same *4M-aug.* MIST or *2M-aug.* MIST checkpoint and discards the updates after producing its fingerprint. The recipe and update budgets in Table 6 are fixed across queries. Phase 1 uses AdamW with learning rate 3.75×10^{-4}, weight decay 10^{-6}, EMA decay 0.99, batch size 64, and the spectrum-side loss. NPLIB1 combines a local training-neighbor pool with a broader background sample; its local pool is selected by spectral and fingerprint ranks fused with $\kappa = 60$.

Let $\theta^{(0)}$ be the initialization and $\theta^{(1)}$ the Phase 1 EMA. Candidate fingerprint ranking uses $0.25E^{\mathrm{prob}}_{\theta^{(0)}}(x_i) + 0.75E^{\mathrm{prob}}_{\theta^{(1)}}(x_i)$. After rank fusion, the local medoid maximizes $K^{-1}\sum_{\ell=1}^{K} T(z(y_{ij}), z(y_{i\ell})) + 0.02/j$. The Phase 2 target blends this medoid with the probability prediction:

$$\bar{z}_i = (1-\gamma)\left[0.25E^{\mathrm{prob}}_{\theta^{(0)}}(x_i) + 0.75E^{\mathrm{prob}}_{\theta^{(1)}}(x_i)\right] + \gamma z_i^{\star}, \tag{10}$$

Phase 2 initializes from the Phase 1 EMA and uses soft-Tanimoto loss on augmented query views with real-spectrum rehearsal. The final EMA prediction is thresholded without a further blend. If query-synthetic support is unavailable, the initial prediction is retained and the query remains in the evaluation. For adaptation from *4M-aug.* MIST, MassSpecGym evaluation uses a fixed 1000-spectrum subset matched to the full test set's baseline-score distribution and molecular and spectral covariates, without using adapted outcomes. NPLIB1 uses all 819 test spectra. The paired initialization/adaptation means are 0.432085/0.535288 and 0.532693/0.562741, giving relative gains of 23.88% and 5.64%. Standalone initialization rows in Table 2 use the full test set and therefore need not equal the subset baseline. The weaker *no-aug.* controls use their own fixed evaluation cohorts. Shared and single-query adaptation use different initializations and mixtures, so their comparison describes two deployment settings rather than isolating the effect of pooling.

Table 6: Single-query adaptation from *4M/2M-aug*. MIST. Pool sizes are maximum numbers of spectral rows, not necessarily distinct structures. Both stages update all encoder parameters.

Setting	MassSpecGym	NPLIB1
Query-synthetic pool / batch	320 / 45	8192 / 12
Training-neighborhood pool / batch	10,000 / 13	2048 / 24 (retrieved)
Additional global-neighborhood pool / batch	–	100,000 / 19
Real-training pool / batch	1000 / 6	1000 / 9
First-stage updates	128	384
First-stage warmup / schedule horizon	8 / 512	8 / 1024
Hard-medoid weight γ	0.10	0.25
Second-stage query / real batch	49 / 1	24 / 1
Second-stage accumulation	4	4
Second-stage updates	16	96
Second-stage optimizer	Adam	Adam
Second-stage learning rate	10^{-4}	2×10^{-5}
Second-stage weight decay	10^{-7}	10^{-7}
Second-stage warmup / schedule horizon	4 / 4000	4 / 4000
Second-stage EMA decay	0.98061	0.98061
Prediction threshold	0.050698	0.010435

B.5 Transfer to MVP and FLARE

We train MVP and FLARE with spectrum-side synthetic pairs while retaining their native representations and contrastive objectives (Chen & Hassoun, 2026; Chen et al., 2026). MVP uses 250,000 synthetic rows and FLARE uses 62,500, combining query-conditioned and training-neighbor rows in a 538:153 ratio with validation and test connectivities excluded. Each minibatch samples distinct structures and then one spectrum per structure, preventing replicate spectra from becoming contrastive negatives. Neither model receives the MIST fingerprint-side objective.

MVP jointly trains molecular-graph, molecular-fingerprint, individual-spectrum, and consensus-spectrum views with all six pairwise contrastive losses, at temperature 0.05. Formula encoders use two transformer layers and 512-dimensional outputs. Training starts from random initialization and uses 400,000 Adam updates, learning rate 7×10^{-5}, zero weight decay, and batches of 48 real plus 16 synthetic structures. The reported consensus view pools spectra sharing a molecular identity in the benchmark metadata; it requires known replicate groups. Retrieval uses cosine similarity to the molecular-graph embedding and retains all 2,909,859 official query–candidate pairs, with ties broken by candidate order.

FLARE retains its bidirectional peak–atom alignment objective and released initialization. We update all parameters for 120,000 Adam steps with batch size 512, learning rate 2.88134×10^{-6}, weight decay 1.83762×10^{-5}, and a 691:77 synthetic-to-real sampling ratio. Both encoders use these fixed endpoints. FLARE evaluation follows its released subformula assignment and candidate processing: up to 60 peaks, 20-ppm tolerance, relative-intensity threshold 0.001, disconnected candidates removed, and worst-rank handling of ties. It evaluates individual spectra and retains 2,823,799 query–candidate pairs. The reported results retain these backbone-specific evaluation conventions.

C Molecular decoder training and inference

C.1 Pretraining and fingerprint perturbation

The molecular decoder is a 208-million-parameter causal transformer with 16 layers, hidden width 1024, 16 attention heads, feed-forward width 4096, and context length 512. It uses GELU, learned positional embeddings, pre-layer normalization, tied input/output embeddings, and dropout 0.1. We extend the SAFE tokenizer with 4096 fingerprint-bit tokens and three conditioning markers. The prefix contains the molecular formula and active fingerprint bits; only the target SAFE sequence and end token contribute to the likelihood loss.

Pretraining and mid-training use molecule-derived formulae, fingerprints, and sequences. Experimental spectra are used to estimate encoder errors, not as decoder training inputs. For each of 20 fingerprint thresholds, we estimate bitwise true-positive retention and false-positive rates from labeled training spectra. Sparse counts are smoothed toward the global rates, with smoothing strength selected from $\{1, 5, 20, 100\}$ by balanced validation negative log-likelihood. Separate logit offsets then calibrate the aggregate retention and false-positive rates on validation spectra.

For retention rate k_{rj} and false-positive rate a_{rj}, mid-training samples $\widetilde{z}_j \sim \mathrm{Bernoulli}(z_j k_{rj} + (1 - z_j)a_{rj})$. A threshold regime r is sampled uniformly once per molecule, and bits are sampled independently conditional on that regime. Clean fingerprints are retained with probability 0.2. The molecular target is unchanged. The shared regime varies error severity; conditional bit sampling does not model correlations between individual errors. All decoder stages use AdamW with $\beta_1 =$ 0.9, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, cosine learning-rate decay, bfloat16, and gradient clipping at norm 1. Pretraining makes 960 million molecular presentations and each mid-training run makes 100 million, including repeated sampling. The two datasets share the pretrained initialization and use separate mid-training and post-training models. We use the terminal checkpoints at the budgets in Table 7.

Table 7: SOUR decoder training. Mid-training runs share the pretrained initialization and use dataset-specific encoder error models. Post-training starts from the corresponding mid-trained decoder.

Setting	Pretraining	Mid-training	Post-training
Per-device batch	64	32	16
Gradient accumulation	5	10	2
Global batch	1920	1280	128
Optimizer steps	500,000	78,125	12,000 / 8000
Peak learning rate	3×10^{-4}	3×10^{-5}	10^{-3}
Warmup steps	50,000	2000	600 / 400
Weight decay	0	0	0
Trainable layers	16	16	2 appended

C.2 Learning from synthetic correction pairs

Correction training pairs an experimental training spectrum with an incorrect molecular neighbor and its known target structure. We retrieve up to 64 neighbors per MassSpecGym training reference or 176 per NPLIB1 reference, remove the target connectivity, and retain candidate–target fingerprint similarities in $[0.2, 0.98]$. Candidates need not share the target formula. Each candidate is simulated and compared with the observed spectrum to construct feedback, giving 1,648,746 MassSpecGym and 1,030,051 NPLIB1 examples. Each example teaches one correction step; repeated correction at test time composes the same learned operation.

Observed and simulated spectra are normalized to their maximum intensities. We retain up to 100 peaks above relative intensity 0.01, exclude the precursor region, and match peaks one-to-one within 20 ppm. For matched fragments M, unmatched simulated fragments U, intensities I_q, and atom masks A_q, atom b receives score $v_b = \sum_{q \in M} I_q \mathbf{1}\{b \in A_q\} - \sum_{q \in U} I_q \mathbf{1}\{b \in A_q\}$. Scores are normalized by the largest absolute value. Unmatched observed peaks contribute their formulae and binned intensities. The feedback contains up to 16 edit cues, 12 keep cues, and 12 missing peaks. The decoder predicts a complete molecular sequence. Target fingerprints and target-derived edits are absent from its feedback.

Post-training appends two transformer blocks and 308 feedback tokens. The original layers and embedding rows remain frozen; only the added blocks and token rows are trained. The new residual projections start at zero, making the added blocks initially the identity. In addition to Equation 7, we compare the corrected sequence under genuine feedback and a control that reverses edit/keep directions and perturbs missing-peak cues. With their mean token negative log-likelihoods denoted by n^+ and n^-, the loss is $\mathcal{L}_{\mathrm{post}} = \mathcal{L}_{\mathrm{corr}} + 0.25\,\mathbb{E}[\mathrm{softplus}(0.2 + n^+ - n^-)]$. The auxiliary term is evaluated on half of each batch. Decoder parameters remain fixed at test time.

C.3 Generation and candidate ranking

SOUR-base uses deterministic beam search with 50 beams and at most 256 new tokens. Candidates are parsed, canonicalized without stereochemistry, and deduplicated by connectivity. Formula-matching candidates are preferred; if none match, valid candidates provide a fallback. SOUR-full further removes disconnected structures and applies one correction round to up to eight parents in MassSpecGym and NPLIB1, with eight beams per parent. Revised structures are simulated and merged with the parent pool.

We rank this union in ascending order of

$$s_{\mathrm{rank}}(y) = \frac{a(y)}{\max(1, K-1)} + w_{\mathrm{spec}} d_{\mathrm{spec}}(x, G(y)) + w_{\mathrm{fp}}[1 - T(\hat{z}, z(y))] + b_{\mathrm{rev}} \mathbf{1}\{y \text{ is revised}\}, \tag{11}$$

Here $K = 50$, $\hat{z}$ is the generation fingerprint, and $a(y)$ is the original zero-based rank or, for a revision, its parent's rank plus $(r_{\mathrm{beam}} + 1)/5$. For normalized binned spectra, $d_{\mathrm{spec}}(x, x') = [2H((x + x')/2) - H(x) - H(x')]/\log 4$, where H is Shannon entropy. The weights $(w_{\mathrm{spec}}, w_{\mathrm{fp}}, b_{\mathrm{rev}})$ are $(0.2, 2, 0)$ for MassSpecGym and $(0.5, 0.5, -0.05)$ for NPLIB1. We retain at most 50 candidates. All weights and budgets are fixed during evaluation. Because ranking and truncation act on the merged pool, an additional round need not improve recovery.

D Supplementary results and computation

D.1 Decoder mid-training

Table 8 holds the NPLIB1 encoder, predicted fingerprints, formulae, beam-50 generation, and formula-preferred ranking fixed. The fingerprint threshold is 0.1947156, calibrated on validation spectra. After 5000 mid-training updates, Top-10 recovery improves by 8.67 percentage points, while Top-1 similarity remains nearly unchanged. This comparison measures the benefit of continuation with perturbed conditions.

Table 8: Controlled NPLIB1 decoder mid-training comparison. The encoder, fingerprint inputs, ranking rule, and generation budget are fixed. Accuracy is in percent.

Decoder stage	Acc@1	Tani@1	Acc@10	Tani@10
Molecular pretraining only	22.22	0.5778	27.47	0.6104
+ 5000 mid-training updates	23.81	0.5770	36.14	0.6502

D.2 Correction and candidate selection

Table 9 separates impacts of performance gain from one-shot SOUR to SOUR-full. The overall pipeline trades some fingerprint similarity for exact recovery. Second-round refinement brings very marginal performance gain while adding a great amount of compute cost.

Table 9: Correction ablation with the final encoders. Accuracy is in percent. The middle two rows use the same parent pool and hybrid ranking as correction. Each correction round decodes eight beams per selected parent.

	NPLIB1				MassSpecGym			
Setting	Acc@1	Tani@1	Acc@10	Tani@10	Acc@1	Tani@1	Acc@10	Tani@10
One-shot SOUR-base	23.20	0.6067	37.36	0.6720	17.09	0.5230	24.06	0.5640
Parents, hybrid ranking	25.64	0.6149	38.10	0.6751	16.75	0.5223	24.15	0.5646
SOUR-full (one round)	27.35	0.6070	40.42	0.6673	17.98	0.5038	24.70	0.5419
+ second round	27.35	0.6074	40.54	0.6676	17.98	0.5039	24.82	0.5423

Table 10: Estimated core computational cost per test spectrum on an NVIDIA RTX PRO 6000, in seconds. SOUR-full includes inference, test-conditioned synthetic generation, and encoder training. Appendix D.4 details the calculation and scope.

	MSNovelist	MS-BART	MADGEN	DiffMS (100 candidates)	FRIGID (25 rounds)	MS-GPT (1,500 candidates)	SOUR-full			
Dataset							Inference and correction	Synthetic generation	Phase 1 and Phase 2 training	**Total**
NPLIB1	0.031	0.98	3.77	32.58	162.40	5.08	2.99	5.52	5.60	**14.11**
MassSpecGym	0.036	0.53	7.25	55.59	232.68	7.74	3.61	0.097	0.70	**4.41**

D.3 Adaptation with experimental neighbors

As a control, we apply the real-neighbor tuning strategy of Mismetti et al. (2025) to MIST. We cluster normalized test fingerprint predictions into 500 groups and choose the spectrum nearest each centroid. For each representative, the encoder retrieves 64 experimental training spectra by cosine similarity and takes one supervised update. Representations are refreshed every 50 updates. AdamW uses learning rate 5×10^{-5} with decay 0.995 per update, zero weight decay, and the native fingerprint loss. The run ends after 500 updates; thresholds are calibrated on validation spectra.

Figure 5 shows the full-test trajectory on MassSpecGym. At the fixed 500-update endpoint, mean fingerprint Tanimoto changes from 0.264340 to 0.266623 for *no-aug.* MIST and from 0.433096 to 0.431596 for *4M-aug.* MIST. Under this budget, retrieved experimental neighbors provide little improvement over either initialization.

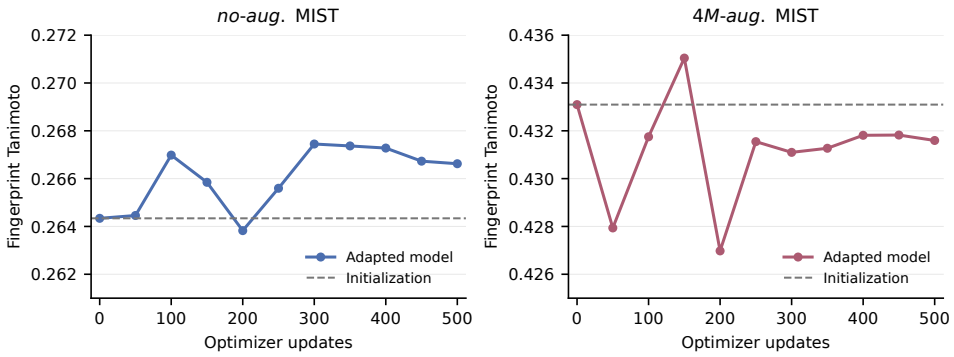

Figure 5: Real-neighbor adaptation on MassSpecGym over a fixed 500-update budget. Each point evaluates all 17,556 test spectra. Dashed lines mark the initialization; the panels have equal vertical spans.

D.4 Computational cost

Table 10 reports estimated core computational cost per test spectrum using an NVIDIA RTX PRO 6000 Blackwell Server Edition as the reference device. For SOUR-full, we include molecular inference and correction, generation of test-conditioned synthetic spectra, and both phases of shared encoder training. Let N_{test} denote the number of test spectra, N_{syn} the number of test-conditioned synthetic spectra, $\bar{t}_{\text{syn}}$ the measured generation time per synthetic spectrum, and T_{train} the total encoder optimization time. We calculate

$$C_{\text{total}} = C_{\text{inference}} + \frac{N_{\text{syn}}\bar{t}_{\text{syn}}}{N_{\text{test}}} + \frac{T_{\text{train}}}{N_{\text{test}}}, \tag{12}$$

where $N_{\text{test}} = 819$ for NPLIB1 and $N_{\text{test}} = 17{,}556$ for MassSpecGym.

Inference and correction. We retain the component timing convention of Figure 4. MSNovelist and MS-BART use 100 beams, MADGEN uses 50 candidates, DiffMS uses 100 candidates, FRIGID uses 128 candidates over 25 generation rounds, and MS-GPT uses 1,500 candidates. For SOUR-full, beam search uses width 50, followed by one correction round with up to eight parents and beam width eight per parent. Its inference costs are 2.990 and 3.612 seconds per spectrum on NPLIB1

and MassSpecGym, respectively. These costs already include forward simulation for candidate feedback, which is therefore not charged again under synthetic training-data generation.

Test-conditioned synthetic generation. The query-conditioned training pools contain 7,654,065 synthetic spectra for NPLIB1 and 2,538,905 for MassSpecGym. We uniformly sample 2,048 physical rows from each pool using seed 20260926 and reconstruct their molecular inputs and acquisition conditions. We time the native forward simulator in FP32 with batch size 256, discard the first full pass for warmup, and average two subsequent passes. The resulting generation times are 0.590738 and 0.671728 milliseconds per synthetic spectrum, respectively. Thus,

$$C_{\text{syn}}^{\text{NPLIB1}} = \frac{7{,}654{,}065 \times 0.000590738}{819} \approx 5.521 \text{ s}, \tag{13}$$

$$C_{\text{syn}}^{\text{MassSpecGym}} = \frac{2{,}538{,}905 \times 0.000671728}{17{,}556} \approx 0.097 \text{ s}. \tag{14}$$

Each generated spectrum is counted once; epochs, repeated minibatch sampling, and stochastic input augmentation do not increase the generation count.

Encoder training. We estimate optimization time by multiplying the measured time per optimizer step by the fixed training budget:

$$T_{\text{train}} = S_1 \bar{t}_1 + S_2 \bar{t}_2, \tag{15}$$

where S_k and $\bar{t}_k$ denote the number of optimizer steps and mean step time in Phase k. The timings include gradient accumulation, forward and backward computation, gradient clipping, optimizer updates, and exponential moving-average updates. NPLIB1 uses 30,000 Phase 1 steps and 400 Phase 2 steps; MassSpecGym uses 80,000 and 500 steps. The resulting totals are 1.2742 and 3.4221 GPU-hours, giving

$$C_{\text{train}}^{\text{NPLIB1}} = \frac{1.2742 \times 3600}{819} \approx 5.601 \text{ s}, \tag{16}$$

$$C_{\text{train}}^{\text{MassSpecGym}} = \frac{3.4221 \times 3600}{17{,}556} \approx 0.702 \text{ s}. \tag{17}$$

These totals include optimization on all training sources used in both phases. Combining the three components yields 14.11 seconds per test spectrum on NPLIB1 and 4.41 seconds on MassSpecGym.