MolSpecFlow: Modality-Incomplete Molecular–Spectral Learning for MS/MS

Yu Wang[1,2,3*] **Fan Yang**[3‡] **Kaikun Xu**[4,5] **Hao Li**[6]
Li Yuan[6,2] **Jun Zhu**[7] **Jingjie Zhang**[8] **Zhenchao Tang**[3]
Yatao Bian[9] **Cheng Chang**[4,5] **Jianhua Yao**[3†]

[1] School of Computer Science, Peking University, China
[2] School of Artificial Intelligence for Science, Peking University, China
[3] AI for Life Sciences Lab, Tencent, China
[4] State Key Laboratory of Medical Proteomics, National Center for Protein Sciences, Beijing, China
[5] Research Unit of Proteomics-driven Cancer Precision Medicine,
Chinese Academy of Medical Sciences, China
[6] School of Electronic and Computer Engineering, Peking University, China
[7] Tsinghua-Peking Joint Center for Life Sciences, School of Life Sciences,
Tsinghua University, China
[8] Department of Computer Science and Engineering,
The Chinese University of Hong Kong, Hong Kong SAR, China
[9] Department of Computer Science, National University of Singapore, Singapore

Abstract

Untargeted metabolomics produces massive numbers of tandem mass spectra, yet most spectra remain difficult to assign to molecular structures because each spectrum is only a partial, instrument-dependent observation of the underlying molecule. We study MS/MS annotation as modality-incomplete molecular–spectral inference: molecules, spectra, and fingerprints provide complementary views of the same chemical entity, but fully paired molecule–spectrum data are limited and many training examples contain only molecular or only spectral information. We introduce MolSpecFlow, a mask-conditioned heterogeneous flow framework that represents these modalities as coupled components of a shared molecular–spectral state. Availability masks encode which components exist in each example, and observation masks specify which components are conditioned on or generated, instantiating de novo annotation, molecular retrieval, and spectrum simulation within one backbone. MolSpecFlow uses a shared Mol-Spec Transformer, hybrid discrete–continuous flow dynamics, and expected mass/formula regularization over soft molecular token distributions. Experiments show consistent gains from modality-incomplete pretraining, multi-interface reuse, and external de novo transfer.

1 Introduction

Untargeted metabolomics produces large numbers of tandem mass spectra, yet assigning these spectra to molecular structures remains a central bottleneck in biological and chemical discovery. In LC–MS/MS, a molecule is ionized and fragmented into sparse peaks whose interpretation depends on adducts, collision energy, instrument type, preprocessing, and library curation [Wang et al.,

*Yu Wang and Fan Yang contributed equally.
†Corresponding authors: Fan Yang (fionafyang@tencent.com) and Jianhua Yao (jianhuayao@tencent.com).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

2016, Bushuiev et al., 2024]. Each tandem spectrum therefore provides a partial, instrument-dependent observation of the underlying molecule. This makes MS/MS annotation a scientific inverse problem: molecular structure must be inferred from noisy, partial physical measurements. This partial observability contributes to unidentified chemical "dark matter" [Alseekh et al., 2021, Bittremieux et al., 2022], and exact structure recovery is further complicated by isomers, near-isobaric compounds, and similar fragmentation patterns [Bushuiev et al., 2024]. Library search, molecular networking, and database retrieval remain effective when close references exist [Aron et al., 2020, Li et al., 2021]; broader metabolite annotation requires models that can integrate molecular and spectral evidence under incomplete observation.

Modern MS/MS annotation now spans several molecular–spectral inference interfaces. A system may infer structures from spectra, retrieve candidate molecules through fingerprint-like representations, or simulate spectra from molecular structures for candidate ranking and pseudo-labeling. MassSpecGym reflects this view by evaluating de novo molecular annotation, molecular retrieval, and spectrum simulation under leakage-aware molecular splits [Bushuiev et al., 2024]. Existing models have advanced each interface separately: spectral representation and fingerprint models support retrieval [Huber et al., 2021b, Goldman et al., 2023], spectrum-to-molecule models support de novo annotation [Stravs et al., 2022, Litsa et al., 2023, Butler et al., 2023, Wang et al., 2025], and forward models support spectrum simulation [Wang et al., 2021, Young et al., 2025]. These interfaces share the same chemical entity viewed through different coordinates: molecular structure, tandem spectra, and retrieval-oriented molecular representations.

A second challenge is uneven supervision across these coordinates. High-quality molecule–spectrum pairs are limited, while molecule-only corpora provide chemical syntax and structural priors, and spectrum-only corpora encode peak statistics and fragmentation regularities [Irwin et al., 2020, Kim et al., 2023, Bushuiev et al., 2024]. This creates a modality-incomplete and geometrically heterogeneous learning problem: molecular strings are discrete and traversal-dependent, spectra are continuous peak sets, and fingerprints are sparse molecular coordinates. We refer to this setting as modality-incomplete because many training examples contain only a subset of the molecular, spectral, and fingerprint components needed by downstream annotation, retrieval, or simulation tasks. This creates a learning problem: molecular strings are discrete and traversal-dependent, spectra are continuous peak sets, and fingerprints are sparse molecular coordinates.

We introduce MolSpecFlow, a modality-incomplete molecular–spectral learning framework for tandem MS annotation. The central algorithmic view is mask-conditioned heterogeneous flow matching: molecule-only, spectrum-only, and paired examples are converted into conditional generation problems over the same molecular–spectral state by changing availability and observation masks. Changing the observation mask instantiates de novo annotation, molecular retrieval, and spectrum simulation as conditional views of one molecular–spectral state. The framework combines a shared Mol-Spec Transformer, hybrid discrete–continuous flow dynamics [Lipman et al., 2022, Liu et al., 2022, Tong et al., 2023, Austin et al., 2021, Gat et al., 2024], and expected mass/formula regularization over soft molecular token distributions [Weininger, 1988, Krenn et al., 2020, Rogers and Hahn, 2010, Dührkop et al., 2019, Goldman et al., 2024]. Across inverse annotation, retrieval, simulation, low-paired-data finetuning, ablations, and external de novo evaluation, MolSpecFlow supports multi-interface reuse and improves learning from molecule-only, spectrum-only, and paired data.

2 Related Work

MS/MS annotation, representation learning, and benchmarks. Small-molecule MS/MS annotation has long relied on spectral-library search, molecular networking, fragmentation trees, and database retrieval [Wang et al., 2016, Aron et al., 2020, Dührkop et al., 2015, 2019]. Learning-based methods extend these priors by mapping spectra to molecular representations: Spec2Vec and MS2DeepScore learn spectral similarities [Huber et al., 2021a,b], MIST and MIST-CF use formula-aware transformers for fingerprint and formula prediction [Goldman et al., 2023, 2024], and DreaMS scales self-supervised spectral representation learning to millions of unannotated spectra [Bushuiev et al., 2026]. MassSpecGym standardizes de novo molecular annotation, molecular retrieval, and spectrum simulation under leakage-aware molecular splits [Bushuiev et al., 2024]. MolSpecFlow follows this multi-interface view and treats retrieval, annotation, and simulation as conditional queries over one molecular–spectral state.

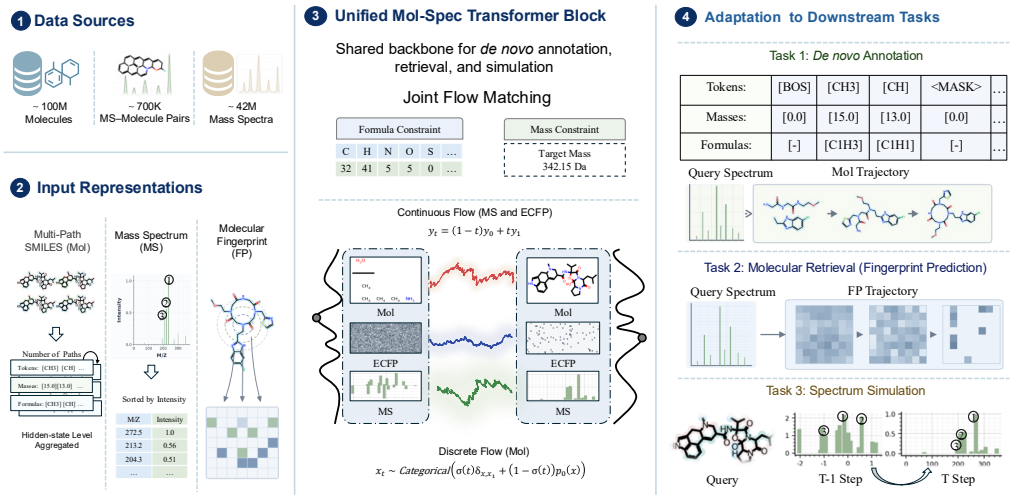


Figure 1: **MolSpecFlow framework.** Molecules, MS/MS spectra, and fingerprints are represented as coupled components of a shared molecular–spectral state. Availability and observation masks specify which components are present, conditioned on, or generated, enabling de novo annotation, retrieval, and spectrum simulation within one backbone. Hybrid discrete–continuous flows model heterogeneous modalities, with expected mass/formula objectives regularizing soft molecular token distributions.

Spectrum-conditioned molecular generation and simulation. De novo molecular annotation from MS/MS has been studied through sequence generation, fingerprint-mediated decoding, and graph generation. Sequence models such as MassGenie, Spec2Mol, and MS2Mol translate spectra into molecular strings [Shrivastava et al., 2021, Litsa et al., 2023, Butler et al., 2023]; MSNovelist and NeuralDecipher decode structures from inferred fingerprints [Stravs et al., 2022, Le et al., 2020]; and scaffold- or formula-constrained graph generators such as MADGEN and DiffMS use structural priors to reduce the generation space [Wang et al., 2025, Bohde et al., 2025]. The forward direction, spectrum simulation from molecular structures, supports candidate ranking and pseudo-labeling through rule-based fragmentation, graph neural networks, fragmentation graphs, and probabilistic fragment generation [Allen et al., 2015, Wang et al., 2021, Murphy et al., 2023, Young et al., 2025, Overstreet et al., 2024]. MolSpecFlow couples inverse and forward directions by switching observation masks within one backbone.

Modality-incomplete and hybrid generative modeling. Incomplete multimodal learning studies settings in which modalities may be unavailable during training or inference, while recent foundation-model and generative approaches further explore missing-modality reconstruction under flexible observed modality sets [Wu et al., 2024, Ke et al., 2025, Dai et al., 2025]. MS/MS annotation is a chemically structured instance of this problem, combining molecule-only, spectrum-only, and paired supervision across heterogeneous geometries. Flow matching and discrete generative modeling provide scalable transport-based objectives for continuous, discrete, and structured variables [Lipman et al., 2022, Liu et al., 2022, Tong et al., 2023, Austin et al., 2021, Gat et al., 2024, Xu et al., 2024]. MolSpecFlow uses availability masks and observation masks to turn molecule-only, spectrum-only, and paired examples into conditional training instances over one heterogeneous molecular–spectral state.

3 Methods

3.1 Modality-incomplete molecular–spectral state

MolSpecFlow formulates tandem MS annotation as mask-conditioned inference over a heterogeneous molecular–spectral state. Each example is represented as

$$\mathbf{z} = (\mathbf{x}, \mathbf{s}, \mathbf{h}),$$

where $\mathbf{x}$ is a molecular token sequence, $\mathbf{s}$ is an MS/MS peak-set representation, and $\mathbf{h}$ is a molecular fingerprint. An availability mask $a \in \{0, 1\}^3$ records which components exist in the example. This

mask is what makes modality-incomplete training explicit: molecule-only examples expose molecular coordinates without spectra, spectrum-only examples expose spectral coordinates without structures, and paired examples expose cross-modal correspondence. For molecule-only data, x is available and h can be computed deterministically; for spectrum-only data, only s is available; for paired data, all available molecular and spectral components are used. The fingerprint is a deterministic, molecule-derived structural coordinate for retrieval rather than an independent experimental modality.

For each training or inference instance, an observation mask $\mathbf{o} \leq \mathbf{a}$ specifies which available components are provided as conditions, and the target mask is

$$\mathbf{r} = \mathbf{a} - \mathbf{o}.$$

MolSpecFlow learns the conditional distribution

$$p_\theta(\mathbf{z_r} \mid \mathbf{z_o}, \mathbf{a}, \mathbf{o}, \mathbf{c}),$$

where $\mathbf{z_o}$ denotes observed components and $\mathbf{c}$ includes available metadata such as precursor m/z, adduct, charge, collision energy, instrument type, and the generative time. Changing $\mathbf{o}$ instantiates different task interfaces: de novo molecular annotation observes spectra and predicts molecules $(o_s = 1, r_x = 1)$, molecular retrieval observes spectra and predicts fingerprints $(o_s = 1, r_h = 1)$, and spectrum simulation observes molecules and predicts spectra $(o_x = 1, r_s = 1)$. Task interfaces are specified by observation and target masks within a shared pretrained architecture. Downstream adaptation additionally involves task-specific fine-tuning, output heads, and inference settings.

3.2 State coordinates

MolSpecFlow uses coordinates matched to the geometry of each component. A molecule is represented as a padded token sequence

$$\mathbf{x} = (x_1, \ldots, x_L), \qquad x_i \in \{1, \ldots, V\}.$$

Each token has an identity embedding and, when chemically defined, a monoisotopic mass $m(v)$ and elemental composition vector $\mathbf{c}(v)$. Structural or syntactic tokens without intrinsic composition are assigned zero entries and excluded by valid-token masks. The molecular embedding combines token identity, mass, formula, and position features,

$$\mathbf{e}_i^{\mathrm{mol}} = \mathbf{E}_{\mathrm{id}}(x_i) + \phi_{\mathrm{mass}}(m(x_i)) + \phi_{\mathrm{formula}}(\mathbf{c}(x_i)) + \mathbf{E}_{\mathrm{pos}}(i).$$

Randomized molecular paths may be sampled as representation augmentation; unless otherwise stated, direct de novo decoding uses a single path, while multi-path effects are analyzed in Appendix G.

An MS/MS spectrum is represented as a fixed-size peak array

$$\mathbf{s} = \{(m_i, I_i, b_i)\}_{i=1}^{N},$$

where m_i is the peak m/z, I_i is normalized intensity, and b_i is a valid-peak indicator. We retain the top N peaks by intensity and pad shorter spectra. Peak m/z, intensity, precursor information, and scalar metadata are encoded with learnable basis encoders. To capture neutral-loss-like relations, the spectral stream adds a learned attention bias indexed by discretized pairwise mass differences $|m_i - m_j|$. The fingerprint component $\mathbf{h} \in \mathbb{R}^D$ is a retrieval-oriented molecular coordinate computed from molecular structures when available and predicted from spectra during retrieval.

3.3 Mask-conditioned heterogeneous flow matching

The state combines discrete molecular tokens with continuous spectra and fingerprints. MolSpecFlow therefore uses a masked hybrid objective: observed components are fixed as conditions, target components are corrupted according to their geometry, and unavailable components are represented by learned null states. Let $t \in [0, 1]$ be a shared generative time and $\mathbf{z}_t = (\mathbf{x}_t, \mathbf{s}_t, \mathbf{h}_t)$ be the partially corrupted state.

For a continuous target $q \in \{s, h\}$, let $\mathbf{y}_1^q$ be the clean coordinate and $\mathbf{y}_0^q \sim \mathcal{N}(0, \mathbf{I})$. We use the linear path

$$\mathbf{y}_t^q = (1 - t)\mathbf{y}_0^q + t\mathbf{y}_1^q,$$

whose target velocity is $\mathbf{y}_1^q - \mathbf{y}_0^q$. The continuous loss is

$$\mathcal{D}_{\mathrm{cont}}^q = \mathbb{E}\left[\|v_\theta^q(\mathbf{z}_t, t, \mathbf{z_o}, \mathbf{a}, \mathbf{o}, \mathbf{c}) - (\mathbf{y}_1^q - \mathbf{y}_0^q)\|_{2\,\mathbf{M}_q}^2\right], \quad q \in \{s, h\},$$

where $\mathbf{M}_q$ masks invalid padded entries.

For molecular targets, the clean token $x_{1,i}$ at position i is corrupted by a masked discrete path

$$p_t(x_i \mid x_{1,i}) = \kappa(t)\delta_{x_{1,i}} + (1-\kappa(t))\,\delta_{[\mathrm{MASK}]}, \qquad \kappa(0)=0,\ \kappa(1)=1.$$

The molecular head predicts clean-token probabilities, and the discrete loss is

$$\mathcal{D}^x_{\mathrm{disc}} = \mathbb{E}\left[-\frac{1}{|\Omega_x|}\sum_{i\in\Omega_x}\log p_\theta(x_{1,i} \mid \mathbf{z}_t, t, \mathbf{z}_{\mathbf{o}}, \mathbf{a}, \mathbf{o}, \mathbf{c})\right],$$

where Ω_x denotes non-padding molecular positions. The mask-conditioned hybrid loss is

$$\mathcal{L}_{\mathrm{MHG}} = \mathbb{E}_{(\mathbf{z},\mathbf{a}),\mathbf{o},t}\left[r_x\mathcal{D}^x_{\mathrm{disc}} + r_s\mathcal{D}^s_{\mathrm{cont}} + r_h\mathcal{D}^h_{\mathrm{cont}}\right].$$

Losses are computed only for available but unobserved components, allowing molecule-only, spectrum-only, and paired examples to share one conditional training procedure.

3.4 Mol-Spec Transformer and chemical regularization

MolSpecFlow parameterizes the conditional dynamics with a shared Mol-Spec Transformer. Molecular tokens, spectral peaks, and fingerprint tokens are embedded with modality, availability, and observation/target-mask embeddings. Each block applies intra-stream self-attention, cross-modal attention, and feed-forward updates, with learned null embeddings for unavailable components. The spectral stream uses the relative-m/z attention bias described above, and a global condition vector containing time and available MS/MS metadata modulates the streams through adaptive normalization. Shared hidden states feed modality-specific heads: a molecular head predicts clean-token distributions, a spectral head predicts peak-coordinate velocities, and a fingerprint head predicts continuous fingerprint velocities.

To encourage chemically consistent molecular predictions before hard decoding, MolSpecFlow adds differentiable expected mass and formula constraints to the molecular head. Let $\mathbf{P} \in \mathbb{R}^{L\times V}$ be the predicted token probability matrix, $\mathbf{C} \in \mathbb{R}^{V\times K_e}$ the token-to-element registry, and $\mathbf{w} \in \mathbb{R}^V$ the token mass vector. The expected formula and mass are

$$\widehat{\mathbf{y}} = \sum_{i=1}^{L}\sum_{v=1}^{V} P_{i,v}\mathbf{C}_v, \qquad \widehat{m} = \sum_{i=1}^{L}\sum_{v=1}^{V} P_{i,v}\mathbf{w}_v.$$

For examples with valid formula or precursor-compatible neutral-mass supervision, we use

$$\mathcal{L}_{\mathrm{formula}} = \mathrm{SmoothL1}(\widehat{\mathbf{y}}, \mathbf{y})\,\mathbb{M}_{\mathrm{valid}}, \qquad \mathcal{L}_{\mathrm{mass}} = \|\widehat{m} - m\|_2^2\,\mathbb{M}_{\mathrm{valid}}.$$

The full objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{MHG}} + \lambda_{\mathrm{formula}}\mathcal{L}_{\mathrm{formula}} + \lambda_{\mathrm{mass}}\mathcal{L}_{\mathrm{mass}}.$$

These constraints act on soft token distributions and complement sequence decoding; syntactic validity, valency validity, and ambiguity are evaluated empirically rather than assumed by construction. The expected mass/formula losses are training-time regularizers applied when valid supervision is available; they are not hard constraints and do not require ground-truth molecular structure at inference. Detailed architecture, scalar encoders, token registries, and relative-m/z attention-bias implementation are provided in Appendix B.

3.5 Task interfaces and inference

At inference time, MolSpecFlow selects the task by choosing observation and target masks. For de novo molecular annotation, the spectrum is observed and the molecular component is initialized with mask tokens. The model iteratively predicts clean-token distributions conditioned on the spectral stream and metadata; sampled sequences are decoded, canonicalized with RDKit, deduplicated, and ranked by model score. Invalid or unparsable outputs are retained for validity diagnostics and excluded from candidate ranking when required by the benchmark protocol.

For molecular retrieval, the spectrum is observed and the fingerprint component is generated by continuous integration. Candidate molecules are ranked by similarity between the predicted fingerprint and precomputed candidate fingerprints. For spectrum simulation, the molecule is observed and

the spectral component is initialized from Gaussian noise and evolved through the continuous path conditioned on molecular and metadata features. When multiple target components are generated jointly, discrete and continuous components use synchronized time steps. Additional task-mask definitions, scheduler details, decoding procedures, and synchronized hybrid-inference algorithms are provided in Appendices C and E.

4 Experiments

4.1 Experimental Setup

We evaluate MolSpecFlow on three tandem-MS interfaces: de novo molecular annotation, molecular retrieval, and spectrum simulation. Throughout this section, *de novo generation* refers specifically to *de novo molecular annotation from tandem mass spectra* (spec→mol), i.e., predicting a molecular structure from an input MS/MS spectrum, rather than unconstrained molecular generation in the standard generative-modeling sense.

Benchmarks. Our primary benchmark is MassSpecGym [Bushuiev et al., 2024], which provides standardized evaluations for de novo annotation, molecular retrieval, and spectrum simulation under leakage-aware molecular splits. We additionally evaluate de novo annotation on NPLIB1 as an external benchmark to assess transfer beyond the main training/evaluation environment.

Pretraining data and leakage control. Because paired molecule–spectrum supervision is the bottleneck, MolSpecFlow pretraining combines paired examples with single-modality molecule-only and spectrum-only data. To reduce pretraining leakage, we apply benchmark-overlap filtering before modality-incomplete pretraining. Held-out MassSpecGym and NPLIB1 molecules are excluded from molecule-only pretraining sources by canonical structure identifiers when available, and spectra linked to held-out benchmark entries are removed from spectrum-only sources when spectrum identifiers, dataset identifiers, metadata, or annotated structures permit matching. We describe this protocol in Appendix A.3; we do not claim exhaustive removal of all possible near-duplicate spectra without reliable metadata.

Metrics. For de novo annotation, we report exact-match Top-k accuracy, MCES distance, and Morgan-fingerprint Tanimoto similarity following prior work. For retrieval, we report hit rate at k (HR@k) and MCES@1. For spectrum simulation, we report cosine similarity, Jensen–Shannon similarity, and retrieval-oriented hit-rate metrics used by MassSpecGym. Additional evaluation protocols, path-count analyses, qualitative trajectories, and implementation details are reported in Appendices D, G, I, and C.

Implementation. Unless otherwise stated, models are fine-tuned on the paired training split of each task after modality-incomplete pretraining. All runtime measurements are conducted on a single NVIDIA V100 GPU using the same evaluation scripts and batch/query setting across methods. Because full repeated pretraining and fine-tuning runs are computationally expensive, we report point estimates for several large-scale experiments and interpret small absolute gains conservatively. Training compute and inference runtime are summarized in Appendix H.

4.2 De Novo Molecular Annotation on MassSpecGym

Table 1 reports the de novo molecular annotation results on MassSpecGym.

Among the predictive baselines included in Table 1, MolSpecFlow achieves the strongest exact-match and structural-similarity results.

Although exact-match accuracy remains low under this leakage-aware split, improvements in MCES and Tanimoto indicate structurally closer candidates even when exact recovery fails. We therefore interpret the results as evidence that a unified modality-incomplete molecular–spectral framework improves spectrum-to-structure inference under ambiguity, rather than as evidence that MS/MS-based structure elucidation is solved.

Because this task predicts molecular structures from spectra, we further evaluate the chemical validity of predictive de novo outputs. RDKit validity reaches 0.9286, and the no-AtomValenceException rate

Table 1: De novo molecular annotation on MassSpecGym. Accuracy values are percentages. Oracle settings are diagnostic only and are not treated as deployable settings.

Model	Top-1			Top-10		
	Acc. ↑	MCES ↓	Tan. ↑	Acc. ↑	MCES ↓	Tan. ↑
Predictive setting						
SMILES Transformer	0.00	79.39	0.03	0.00	52.13	0.10
MIST + MSNovelist	0.00	45.55	0.06	0.00	30.13	0.15
SELFIES Transformer	0.00	38.88	0.08	0.00	26.87	0.13
Spec2Mol	0.00	37.76	0.12	0.00	29.40	0.16
MIST + Neuraldecipher	0.00	33.19	0.14	0.00	31.89	0.16
Random Generation	0.00	21.11	0.08	0.00	18.26	0.11
$\text{MADGEN}_{\text{pred}}$	1.31	27.47	0.20	1.54	16.84	0.26
DiffMS	2.30	18.45	0.28	4.25	14.73	0.39
$\text{MolSpecFlow}_{\text{pred}}$	**3.11**	**17.32**	**0.34**	**5.23**	**12.32**	**0.44**
Oracle diagnostic setting						
$\text{MADGEN}_{\text{oracle}}$	10.50	16.27	0.43	12.40	7.08	0.53
$\text{MolSpecFlow}_{\text{oracle}}$	**13.60**	**16.01**	**0.54**	**17.10**	**6.95**	**0.60**

Table 2: A shared MolSpecFlow backbone supports forward spectrum simulation and fingerprint-based molecular retrieval on MassSpecGym. Simulation-induced HR@k ranks candidates by their simulated spectral similarity, whereas fingerprint-based HR@k ranks candidates by similarity to the predicted structural fingerprint.

Spectrum simulation

Model	Cos. ↑	JS ↑	HR@1 ↑	HR@5 ↑	HR@20 ↑
Precursor m/z	0.15 (0.14–0.17)	0.15 (0.14–0.16)	0.38 (0.21–0.62)	1.72 (1.32–2.18)	7.17 (6.32–8.04)
FFN Fingerprint	0.25 (0.24–0.26)	0.24 (0.23–0.25)	8.44 (7.56–9.34)	21.43 (20.10–22.79)	38.57 (36.99–40.23)
GNN	0.19 (0.18–0.20)	0.20 (0.19–0.20)	3.95 (3.37–4.62)	11.92 (10.87–13.00)	26.27 (24.83–27.82)
FraGNNet	0.52 (0.51–0.53)	0.47 (0.46–0.48)	46.64 (44.98–48.26)	72.56 (71.18–74.00)	83.58 (82.34–84.75)
MolSpecFlow	**0.63 (0.61–0.64)**	**0.59 (0.57–0.59)**	**55.32 (52.97–56.87)**	**79.98 (79.01–81.32)**	**89.54 (88.32–90.76)**

Molecular retrieval

Model	HR@1 ↑	HR@5 ↑	HR@20 ↑	MCES@1 ↓
Random	0.37 (0.24–0.54)	2.01 (1.68–2.39)	8.22 (7.53–8.89)	30.81 (30.40–31.21)
DeepSets	1.47 (1.18–1.77)	6.21 (5.64–6.82)	19.23 (18.24–20.26)	25.11 (24.84–25.39)
Fingerprint FFN	2.54 (2.17–2.99)	7.59 (6.96–8.28)	20.00 (19.01–20.98)	24.66 (24.38–24.94)
DeepSets + Fourier	5.24 (4.71–5.83)	12.58 (11.80–13.42)	28.21 (27.10–29.36)	22.13 (21.85–22.43)
MIST	14.64 (13.82–15.54)	34.87 (33.69–36.10)	59.15 (57.89–60.39)	15.37 (15.12–15.62)
JESTR	15.13	36.75	60.32	–
JESTR_{NR}	15.62	37.47	60.55	–
MolSpecFlow	**17.32 (16.59–19.25)**	**39.93 (38.42–40.42)**	**64.85 (63.16–66.04)**	**14.05 (13.89–14.51)**

JESTR rows are public MassSpecGym retrieval results without confidence intervals or MCES@1 in the available report.

reaches 0.9355; the formula-count error has mean 0.0522 and median 0.0000. Additional validity and mass/formula consistency diagnostics are reported in Appendix F.5. These diagnostics indicate that most decoded molecules are chemically parseable, although the expected mass and formula objectives remain soft regularizers rather than hard validity guarantees.

4.3 A Single Backbone Supports Retrieval and Simulation

Table 2 evaluates whether one modality-incomplete pretrained backbone can support molecular–spectral interfaces beyond de novo annotation. For spectrum simulation, MolSpecFlow improves cosine similarity, Jensen–Shannon similarity, and simulation-induced retrieval hit rates over FraGNNet, indicating that the shared state also supports forward molecular-to-spectral inference. For molecular retrieval, MolSpecFlow improves HR@1/5/20 over MIST and available JESTR variants, while reducing MCES@1 relative to MIST. Because some external retrieval baselines do not report confidence intervals or MCES@1, we treat these results as evidence for a strong retrieval interface rather than an exhaustive retrieval state-of-the-art claim.

Importantly, this evaluation tests interface reuse rather than training separate task-specific systems: de novo annotation, retrieval-oriented fingerprint prediction, and spectrum simulation all start from the same modality-incomplete pretrained MolSpecFlow backbone. Task-specific fine-tuning changes the observation/target masks and output heads, but preserves the shared molecular–spectral state formulation. Appendix F.1 further compares this shared-backbone design with separately trained task-specific variants.

Table 3: External de novo molecular annotation on NPLIB1. Accuracy values are percentages.

Model	Top-1			Top-10		
	Acc. ↑	MCES ↓	Tan. ↑	Acc. ↑	MCES ↓	Tan. ↑
Spec2Mol	0.00	27.82	0.12	0.00	23.13	0.16
MADGEN	2.10	20.56	0.22	2.39	12.64	0.27
MIST + Neuraldecipher	2.32	12.11	0.35	6.11	9.91	0.43
MIST + MSNovelist	5.40	14.52	0.34	11.04	10.23	0.44
DiffMS	8.34	11.95	0.35	15.44	9.23	0.47
MolSpecFlow$_{\text{pred}}$	**10.45**	**10.32**	**0.36**	**17.25**	**8.42**	**0.51**

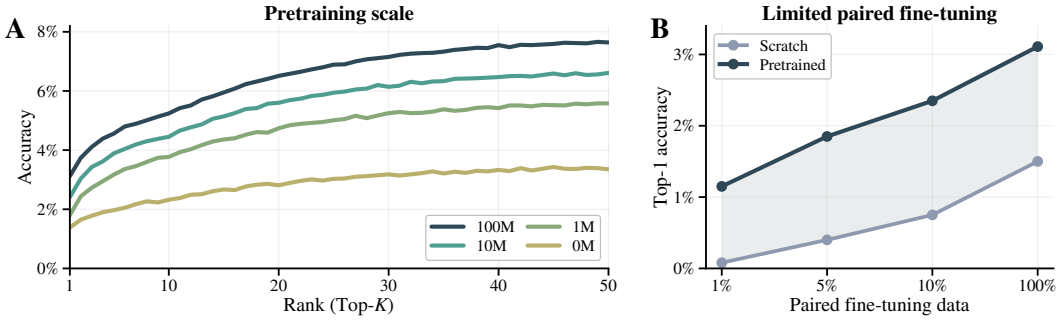


Figure 2: **Effect of modality-incomplete pretraining. A**, De novo molecular annotation improves consistently as the pretraining corpus increases. **B**, Pretraining yields larger gains under limited paired molecule–spectrum fine-tuning data.

4.4 External Transfer on NPLIB1

To evaluate external transfer, we test the spectrum-to-structure interface on NPLIB1. As shown in Table 3, MolSpecFlow achieves 10.45% Top-1 and 17.25% Top-10 accuracy, improving over directly comparable published baselines across exact-match and structural-similarity metrics. This result is consistent with the broader motivation of the model: learning molecular priors from molecule-only data, spectral regularities from spectrum-only data, and cross-modal correspondence from paired data can improve transfer to external de novo annotation settings.

4.5 Effect of Modality-Incomplete Pretraining

Figure 2 examines the effect of pretraining scale and paired-data availability. Figure 2A shows that de novo molecular annotation improves consistently as the amount of modality-incomplete pretraining increases. Figure 2B further shows that pretraining is especially helpful under limited paired fine-tuning. This setting directly tests the modality-incomplete premise: when paired supervision is scarce, single-modality molecular and spectral corpora provide useful complementary supervision. Under 1% paired fine-tuning data, the full modality-incomplete model reaches 1.15 Top-1 and 2.45 Top-10, compared with 0.19 and 0.64 for paired-only training and 0.08 and 0.31 when trained from scratch. Separating the two unpaired sources shows that both are useful: molecule-only + paired reaches 0.67 / 1.73, while spectrum-only + paired reaches 0.49 / 1.51. These results support the main premise of MolSpecFlow: in tandem-MS annotation, paired supervision is limited, but molecule-only and spectrum-only corpora provide complementary information that can be incorporated within a single training framework. Fine-tuning details for the low-paired-data settings are provided in Appendix C.5.

4.6 Ablations, Diagnostics, and Cost

Figure 3 tests whether the main design choices contribute independently to the final model. Panel A isolates the contribution of different pretraining sources under 1% paired supervision. Paired-only pretraining gives limited improvement over scratch training, while molecule-only and spectrum-only pretraining both help; their combination performs best. Panel B shows that the expected mass and formula objectives provide complementary gains: removing either objective degrades de novo annotation, and removing both causes the largest drop from 3.11 / 5.23 to 2.05 / 3.48 in Top-1 /

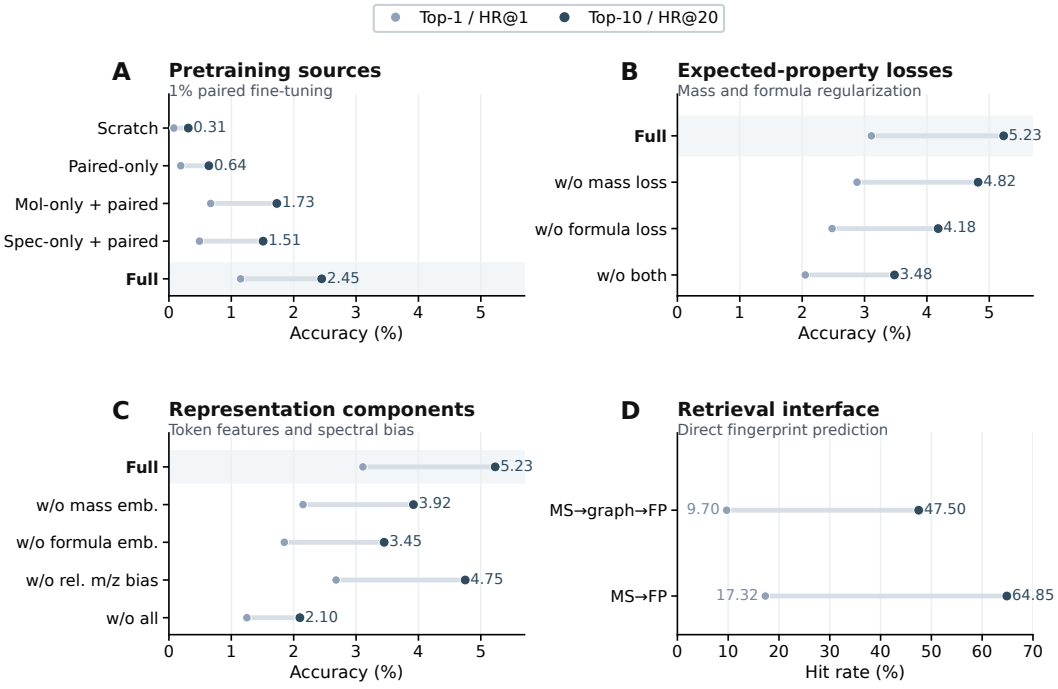


Figure 3: **Main ablations on MassSpecGym.** **A**, Source ablation under 1% paired fine-tuning, isolating paired, molecule-only, and spectrum-only pretraining. **B**, Effect of expected mass and formula objectives on de novo molecular annotation. **C**, Ablation of token-level mass/formula embeddings and relative-m/z spectral bias. **D**, Retrieval-interface comparison between graph-mediated and direct fingerprint prediction.

Top-10. Panel C evaluates representation components. Removing token-level mass embeddings, token-level formula embeddings, or the relative-m/z spectral bias each reduces performance, with the strongest degradation observed when all three are removed. Panel D examines the retrieval interface: direct MS→FP prediction outperforms graph-mediated retrieval, improving HR@1 from 9.70 to 17.32 and HR@20 from 47.50 to 64.85. This supports our use of retrieval as a representation-oriented interface rather than a by-product of graph generation.

Additional numerical ablations and diagnostics are reported in Appendix F. The multi-path study in Appendix G reveals a representation–generation trade-off: increasing the number of randomized molecular paths improves representation-oriented tasks but hurts direct de novo annotation. Accordingly, we use $K = 1$ for de novo annotation and larger K for representation-oriented tasks such as retrieval and spectrum simulation. All runtime measurements are conducted on a single NVIDIA V100 GPU. For de novo annotation, MolSpecFlow with $K = 1$ requires 1042.3 ms/query, compared with 2884.5 ms/query for MADGEN and 7286.2 ms/query for DiffMS. For spectrum simulation, MolSpecFlow requires 1456.6 ms/query, compared with 1849.9 ms/query for FraGNNet. These measurements include model forward passes and decoding/sampling overhead and are reported as practical wall-clock comparisons rather than theoretical complexity claims. Additional runtime and training-compute details are reported in Appendix H.

Qualitative trajectories. Appendix I provides representative generation trajectories. These examples visualize how molecular tokens, spectral peaks, and fingerprints evolve during inference, but they should be interpreted as qualitative illustrations rather than evidence of guaranteed synchronized reasoning.

5 Discussion and Limitations

MolSpecFlow is a step toward unified molecular–spectral modeling under incomplete supervision, not a complete solution to MS/MS-based structure elucidation. Its performance is limited by scarce license-compatible molecule–spectrum corpora and by measurement heterogeneity across instruments, collision energies, adducts, ionization modes, preprocessing pipelines, and curation protocols. Although MassSpecGym and NPLIB1 provide important open benchmarks, they cover only part of real LC–MS/MS chemical and instrumental diversity.

MS/MS annotation is also intrinsically underdetermined: constitutional isomers, positional isomers, near-isobaric compounds, and structurally related analogs can yield similar fragmentation patterns. Thus, exact-match accuracy should be interpreted together with structural similarity, retrieval quality, formula/mass consistency, validity, and candidate-set calibration. MolSpecFlow should be used for candidate generation and prioritization in expert-assisted annotation, not as an autonomous source of definitive molecular identifications.

6 Conclusion

MolSpecFlow represents molecules, spectra, and fingerprints as components of a shared molecular–spectral state, enabling de novo annotation, retrieval, and spectrum simulation through observation masks. The results support modality-incomplete molecular–spectral learning as a practical formulation for MS/MS annotation under data scarcity, measurement heterogeneity, and chemical ambiguity.

Acknowledgments and Disclosure of Funding

This work was supported in part by the National Research Foundation, Singapore (Grant Nos. NRF-NRFF18-2026-0004 and NRF-AI4SCT_2025-0004); the Ministry of Education, Singapore (Grant No. T1 251RES2607); the Young Elite Scientists Sponsorship Program by the China Association for Science and Technology (CAST) (Grant No. 2023QNRC001); and the Research Project of the State Key Laboratory of Medical Proteomics (Grant No. SKLP-Y202401).

References

Felicity Allen, Russ Greiner, and David Wishart. Competitive fragmentation modeling of esi-ms/ms spectra for putative metabolite identification. *Metabolomics*, 11(1):98–110, 2015.

Saleh Alseekh, Asaph Aharoni, Yariv Brotman, Kévin Contrepois, John D'Auria, Jan Ewald, Jennifer C. Ewald, Paul D Fraser, Patrick Giavalisco, Robert D Hall, et al. Mass spectrometry-based metabolomics: a guide for annotation, quantification and best reporting practices. *Nature methods*, 18(7):747–756, 2021.

Allegra T Aron, Emily C Gentry, Kerry L McPhail, Louis-Félix Nothias, Mélissa Nothias-Esposito, Amina Bouslimani, Daniel Petras, Julia M Gauglitz, Nicole Sikora, Fernando Vargas, et al. Reproducible molecular networking of untargeted mass spectrometry data using GNPS. *Nature protocols*, 15(6):1954–1991, 2020.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

Wout Bittremieux, Robin Schmid, Florian Huber, Justin JJ van der Hooft, Mingxun Wang, and Pieter C Dorrestein. Comparison of cosine, modified cosine, and neutral loss based spectrum alignment for discovery of structurally related molecules. *Journal of the American Society for Mass Spectrometry*, 33(9):1733–1744, 2022.

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 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine*

Learning Research, pages 4737–4756. PMLR, 2025. URL https://proceedings.mlr.press/v267/bohde25a.html.

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. In *Advances in Neural Information Processing Systems*, volume 37, pages 110010–110027. Curran Associates, Inc., 2024.

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(4):630–640, 2026. doi: 10.1038/s41587-025-02663-3.

Thomas Butler, Abraham Frandsen, Rose Lightheart, Brian Bargh, Thomas Kerby, Kiana West, Joseph Davison, James Taylor, Christoph Krettler, TJ Bollerman, Gennady Voronov, Kevin Moon, Tobias Kind, Pieter Dorrestein, August Allen, Viswa Colluru, and David Healey. MS2Mol: A transformer model for illuminating dark chemical space from mass spectra, 2023. ChemRxiv preprint, version 4.

Ruiting Dai, Chenxi Li, Yandong Yan, Lisi Mo, Ke Qin, and Tao He. Unbiased missing-modality multimodal learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 24507–24517, 2025.

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.

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(4):299–302, 2019.

Kai Dührkop, Louis-Félix Nothias, Markus Fleischauer, Raphael Reher, Marcus Ludwig, Martin A Hoffmann, Daniel Petras, William H Gerwick, Juho Rousu, Pieter C Dorrestein, et al. Systematic classification of unknown metabolites using high-resolution fragmentation mass spectra. *Nature biotechnology*, 39(4):462–471, 2021.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky TQ Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. *Advances in Neural Information Processing Systems*, 37: 133345–133385, 2024.

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.

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. doi: 10.1021/acs.jcim.3c01082.

Florian Huber, Lars Ridder, Stefan Verhoeven, Jurriaan H Spaaks, Faruk Diblen, Simon Rogers, and Justin JJ Van Der Hooft. Spec2vec: Improved mass spectral similarity scoring through learning of structural relationships. *PLoS computational biology*, 17(2):e1008724, 2021a.

Florian Huber, Sven van der Burg, Justin JJ van der Hooft, and Lars Ridder. MS2DeepScore: a novel deep learning similarity measure to compare tandem mass spectra. *Journal of cheminformatics*, 13 (1):84, 2021b.

John J Irwin, Khanh G Tang, Jennifer Young, Chinzorig Dandarchuluun, Benjamin R Wong, Munkhzul Khurelbaatar, Yurii S Moroz, John Mayfield, and Roger A Sayle. ZINC20—a free ultralarge-scale chemical database for ligand discovery. *Journal of chemical information and modeling*, 60(12):6065–6073, 2020.

Guanzhou Ke, Bo Wang, Guoqing Chao, Weiming Hu, and Shengfeng He. How far are we from generating missing modalities with foundation models? *arXiv preprint arXiv:2506.03530*, 2025.

Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, et al. PubChem 2023 update. *Nucleic acids research*, 51(D1): D1373–D1380, 2023.

Mario Krenn, Florian Häse, AkshatKumar Nigam, Pascal Friederich, and Alan Aspuru-Guzik. Self-referencing embedded strings (SELFIES): A 100% robust molecular string representation. *Machine Learning: Science and Technology*, 1(4):045024, 2020.

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.

Yuanyue Li, Tobias Kind, Jacob Folz, Arpana Vaniya, Sajjan Singh Mehta, and Oliver Fiehn. Spectral entropy outperforms ms/ms dot product similarity for small-molecule compound identification. *Nature Methods*, 18(12):1524–1531, 2021.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Eleni E. Litsa, Vijil Chenthamarakshan, Payel Das, and Lydia E. Kavraki. An end-to-end deep learning framework for translating mass spectra to de-novo molecules. *Communications Chemistry*, 6:132, 2023. doi: 10.1038/s42004-023-00932-3.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Michael Murphy, Stefanie Jegelka, Ernest Fraenkel, Tobias Kind, David Healey, and Thomas Butler. Efficiently predicting high resolution mass spectra with graph neural networks. In *International Conference on Machine Learning*, pages 25549–25562. PMLR, 2023.

Richard E. Overstreet, Ethan King, Grady P. Clopton, Julia H. Nguyen, and Danielle K. Ciesielski. QC-GN^{2}oMS^2: a graph neural net for high resolution mass spectra prediction. *Journal of Chemical Information and Modeling*, 64(15):5806–5816, 2024. doi: 10.1021/acs.jcim.4c00446.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

Aditya Divyakant Shrivastava, Neil Swainston, Soumitra Samanta, Ivayla Roberts, Marina Wright Muelas, and Douglas B. Kell. MassGenie: A transformer-based deep learning method for identifying small molecules from their mass spectra. *Biomolecules*, 11(12):1793, 2021. doi: 10.3390/biom11121793.

Michael A Stravs, Kai Dührkop, Sebastian Böcker, and Nicola Zamboni. MSNovelist: de novo structure generation from mass spectra. *Nature Methods*, 19(7):865–870, 2022.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *arXiv preprint arXiv:2302.00482*, 2023.

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):11692–11700, 2021. doi: 10.1021/acs.analchem.1c01465.

Mingxun Wang, Jeremy J Carver, Vanessa V Phelan, Laura M Sanchez, Neha Garg, Yao Peng, Don Duy Nguyen, Jeramie Watrous, Clifford A Kapono, Tal Luzzatto-Knaan, et al. Sharing and community curation of mass spectrometry data with global natural products social molecular networking. *Nature biotechnology*, 34(8):828–837, 2016.

Yinkai Wang, Xiaohui Chen, Liping Liu, and Soha Hassoun. MADGEN: Mass-spec attends to de novo molecular generation. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=78tc3EiUrN`.

David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

Renjie Wu, Hu Wang, Hsiang-Ting Chen, and Gustavo Carneiro. Deep multimodal learning with missing modality: A survey. *arXiv preprint arXiv:2409.07825*, 2024.

Zhe Xu, Ruizhong Qiu, Yuzhong Chen, Huiyuan Chen, Xiran Fan, Menghai Pan, Zhichen Zeng, Mahashweta Das, and Hanghang Tong. Discrete-state continuous-time diffusion for graph generation. *Advances in Neural Information Processing Systems*, 37:79704–79740, 2024.

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

A Data, Licensing, and Leakage Control

A.1 Datasets and usage

MolSpecFlow uses three data types: paired molecule–spectrum examples, molecule-only structures, and spectrum-only MS/MS data. For paired supervision and evaluation, we use MassSpecGym [Bushuiev et al., 2024] and NPLIB1 [Dührkop et al., 2021, Bohde et al., 2025]. MassSpecGym is the primary benchmark because it defines standardized protocols for de novo annotation, molecular retrieval, and spectrum simulation under the MCES split. NPLIB1 is used only as an external de novo annotation benchmark. For molecule-only pretraining, we use large-scale molecular corpora such as ZINC. For spectrum-only pretraining, we use unlabeled public MS/MS spectra from GeMS/DreaMS-style spectral corpora [Bushuiev et al., 2026].

We do not train on license-restricted spectral libraries unless their license explicitly permits machine-learning training and redistribution-compatible use. This is important because several high-quality commercial libraries, including NIST-series resources, are not openly redistributable and may impose restrictions on machine-learning training. We therefore treat open and license-compatible benchmarks as the basis for reproducible evaluation.

A.2 Use of existing assets and release policy

This work uses existing public benchmarks, molecular databases, spectral resources, cheminformatics software, and published baseline results or implementations. We cite the corresponding original papers or resources where they are used. We do not redistribute third-party raw datasets with restricted terms of use. For resources with redistribution constraints, we describe the benchmark protocol and use the assets only under their documented terms. A complete asset-by-asset license, version, and access-date audit will be provided together with the public artifact release.

A.3 Benchmark-overlap filtering

Because MolSpecFlow uses molecule-only and spectrum-only corpora during pretraining, we apply benchmark-overlap filtering before modality-incomplete pretraining. For molecule-only corpora, validation and test molecules from MassSpecGym and NPLIB1 are canonicalized, and overlapping molecules are removed using canonical SMILES, InChIKey, or other available structure identifiers. For spectrum-only corpora, validation and test spectra are removed when they can be matched by spectrum identifiers, dataset identifiers, metadata, or annotated molecular structures. This filtering is applied before pretraining, so held-out benchmark molecules and spectra are not used as paired examples or as unpaired pretraining targets. Because public spectral resources differ in identifier quality, metadata completeness, preprocessing, and licensing constraints, we do not claim exhaustive removal of all possible near-duplicate spectra or structurally related analogs when reliable metadata are unavailable. We therefore interpret pretraining gains as evidence under the stated filtering protocol rather than as a proof of complete pretraining isolation.

A.4 Benchmark splits and evaluation scope

For MassSpecGym, we follow the official MCES split and evaluation protocol. The MCES split separates training, validation, and test molecules by molecular edit distance, reducing structural leakage from close analogs and near-duplicate molecules. All MassSpecGym de novo, retrieval, and spectrum-simulation results in the main paper use this split unless otherwise stated.

NPLIB1 is used only for external de novo annotation. We report Top-1/Top-10 exact-match accuracy, Top-1/Top-10 MCES, and Top-1/Top-10 Tanimoto similarity. We do not report retrieval or spectrum-simulation results on NPLIB1 because these are not part of the commonly used NPLIB1 de novo protocol.

A.5 Molecular and spectral preprocessing

A molecule is represented by a tokenized molecular string. When available, we use explicit-hydrogen molecular strings to make token-level mass and formula accounting less ambiguous. For each token,

we pre-compute a monoisotopic mass and an elemental composition vector used by token embeddings and expected-property losses.

To reduce sensitivity to molecular serialization, we optionally sample K randomized molecular paths for the same graph. The canonical path is always included, and the remaining paths are generated by randomized graph traversals. This strategy is used as representation augmentation and should not be interpreted as exact graph-invariant modeling. The empirical path-count trade-off is reported in Appendix G.

Each MS/MS spectrum is represented as a fixed-size peak array $\{(m_i, I_i, b_i)\}_{i=1}^{N}$, where m_i is the mass-to-charge ratio, I_i is the normalized intensity, and b_i indicates whether the entry is a real peak or padding. We retain the top N peaks by intensity and pad shorter spectra. The global condition vector includes available precursor m/z, adduct, charge, instrument-related metadata, and collision energy; missing metadata are represented by learned unknown embeddings.

B Architecture and Implementation Details

This section provides implementation details for the state-coordinate encoders and Mol-Spec Transformer described in Section 3.2 and Section 3.4.

B.1 Algorithmic summary of mask-conditioned heterogeneous flow matching

Algorithm 1 Mask-conditioned heterogeneous flow matching

1: Sample example $\mathbf{z} = (\mathbf{x}, \mathbf{s}, \mathbf{h})$ and availability mask $\mathbf{a}$.
2: Sample observation mask $\mathbf{o} \leq \mathbf{a}$ and set target mask $\mathbf{r} = \mathbf{a} - \mathbf{o}$.
3: Keep observed components $\mathbf{z_o}$ fixed as conditions.
4: Corrupt target molecular tokens with a discrete mask path.
5: Corrupt target spectra/fingerprints with continuous flow paths.
6: Predict molecular clean-token probabilities and continuous velocities with the shared Mol-Spec Transformer.
7: Compute losses only on available target components and add expected mass/formula regularization when supervision is valid.

B.2 Continuous scalar encoders

High-resolution scalar quantities such as peak m/z, intensity, precursor mass, and collision energy are encoded using learnable basis functions. For a scalar x, we use

$$\mathbf{g}(x) = \left[\exp\left(-\frac{(x-\mu_k)^2}{2\sigma_k^2}\right)\right]_{k=1}^{K_b}, \qquad \phi(x) = \mathbf{W}\mathbf{g}(x) + \mathbf{b},$$

where μ_k and σ_k are learnable centers and widths. This provides a smooth encoding of continuous values without coarse binning.

B.3 Token registry for mass and formula accounting

Let $\mathcal{V}$ be the molecular token vocabulary. For each token $v \in \mathcal{V}$, we pre-compute its monoisotopic mass $m(v)$ and elemental composition vector $\mathbf{c}(v)$. Special tokens and syntactic tokens without chemically meaningful composition are assigned zero mass/formula entries and are excluded by valid-token masks. The molecular token embedding is

$$\mathbf{e}_i^{\text{mol}} = \mathbf{E}_{\text{id}}(x_i) + \phi_{\text{mass}}(m(x_i)) + \phi_{\text{formula}}(\mathbf{c}(x_i)) + \mathbf{E}_{\text{pos}}(i).$$

These features provide local chemical cues but do not by themselves guarantee global syntactic or valency validity.

B.4 Relative m/z attention bias

To model neutral losses and peak-to-peak mass differences, the spectral stream adds a relative m/z bias to self-attention. We avoid per-spectrum min–max normalization because the same neutral

Algorithm 2 Token registry construction for expected mass and formula accounting

Input: tokenizer $\mathcal{T}$, chemistry toolkit
Output: token mass map M, token formula map C, valid-token mask V_{chem}
for each token $t \in \mathcal{T}$ **do**
  $idx \leftarrow \mathcal{T}.\text{token_to_id}(t)$
  if t is a special or purely syntactic token **then**
    $M[idx] \leftarrow 0,\ C[idx] \leftarrow \mathbf{0},\ V_{\text{chem}}[idx] \leftarrow 0$
  else
    Parse the token-level fragment when chemically valid.
    $M[idx] \leftarrow$ monoisotopic mass.
    $C[idx] \leftarrow$ elemental-count vector.
    $V_{\text{chem}}[idx] \leftarrow 1$
  end if
end for

loss should map to the same bin across spectra. For peaks i and j, we discretize the absolute mass difference

$$d_{ij} = |m_i - m_j|, \qquad b_{ij} = \mathbf{E}_{\text{bias}}\left(\text{clip}\left(\lfloor d_{ij}/\delta_m \rfloor, 0, B-1\right)\right),$$

where δ_m is a fixed mass-difference bin width and B is the number of relative-mass bins. The learned bias b_{ij} is added to the spectral self-attention logits. This retains high-resolution peak values while allowing attention heads to model recurring neutral-loss-like mass differences.

B.5 Mol-Spec Transformer

The backbone contains molecular, spectral, and fingerprint streams. Each fusion block applies intra-stream self-attention, cross-modal attention, and feed-forward updates. When multiple molecular paths are used, inter-path attention is applied within the molecular stream. A global condition vector containing flow time and available MS/MS metadata modulates the streams through adaptive normalization. The final hidden states feed modality-specific heads for molecular clean-token prediction, spectral velocity prediction, and fingerprint velocity prediction.

C Training, Fine-tuning, and Inference Details

C.1 Mask-conditioned training

For each example, we define an availability mask $\mathbf{a}$ and an observation mask $\mathbf{o} \leq \mathbf{a}$. Available but unobserved modalities are used as targets. The training loss is

$$\mathcal{L}_{\text{mask}} = \mathbb{E}_{(\mathbf{z},\mathbf{a}),\mathbf{o}}\left[\sum_{q\in\{x,s,h\}} (a_q - o_q)\mathcal{L}_q\right],$$

where $\mathcal{L}_x$, $\mathcal{L}_s$, and $\mathcal{L}_h$ denote molecular-token, spectrum, and fingerprint losses. Losses are computed only for available target modalities, so molecule-only, spectrum-only, and paired examples are all treated as instances of one conditional modeling procedure.

C.2 Task masks used for fine-tuning and inference

Table 4 summarizes the observation and target masks used for the three task interfaces. The same backbone is used across tasks; only the observed and generated components differ.

C.3 Model hyperparameters

C.4 Optimization and loss weights

We train with AdamW using $\beta_1 = 0.9$, $\beta_2 = 0.95$, and weight decay 0.01. The global batch size is 320. The learning rate follows cosine annealing with peak learning rate 2×10^{-4} during pretraining. We use exponential moving average of model parameters with decay 0.999.

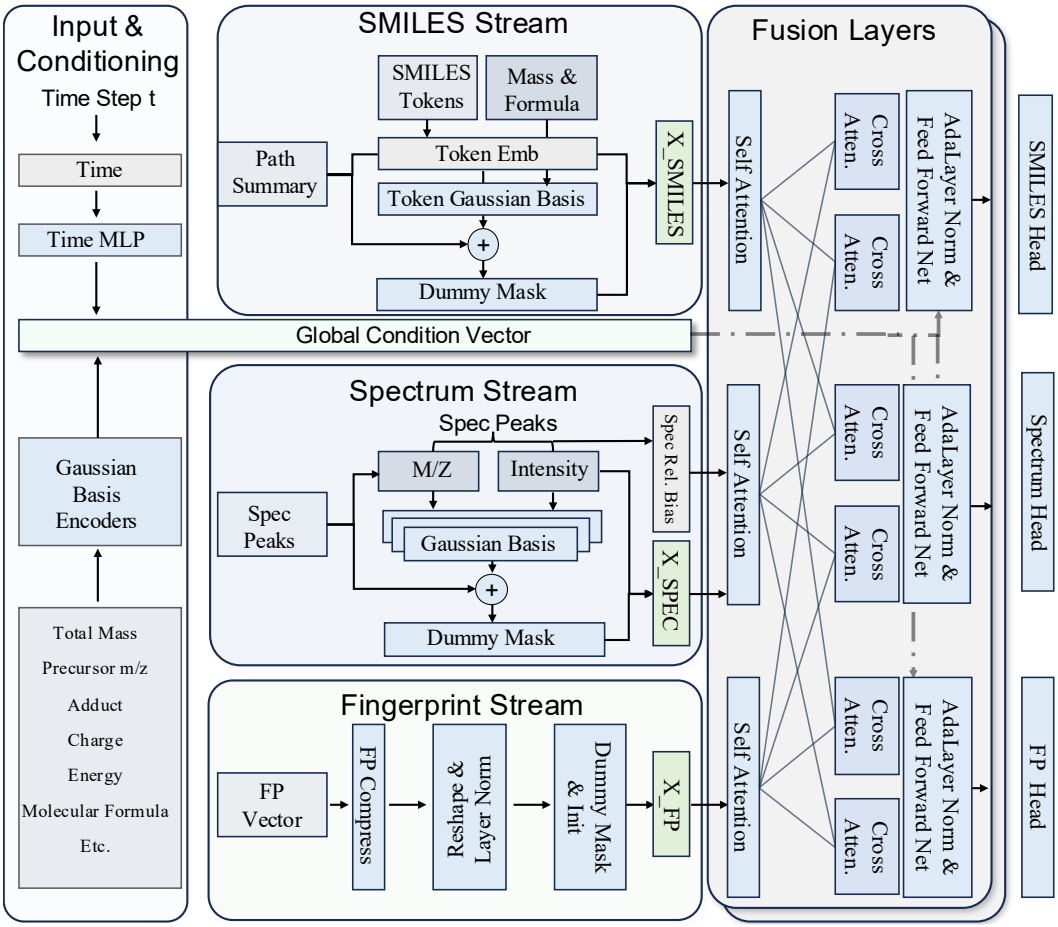


Figure 4: **Detailed MolSpecFlow architecture.** The model contains molecular-token, spectral-peak, and fingerprint streams. Continuous quantities are encoded with learnable basis functions, molecular tokens include mass and formula features, and spectral self-attention uses relative m/z bias. Cross-modal fusion conditions each generated component on the observed components specified by the modality masks.

Table 4: Task interfaces defined by observation and target masks.

Task	Observed component(s)	Target component(s)	Main output
De novo annotation	Spectrum $\mathbf{s}$	Molecule $\mathbf{x}$	Ranked molecular candidates
Molecular retrieval	Spectrum $\mathbf{s}$	Fingerprint $\mathbf{h}$	Ranked database candidates
Spectrum simulation	Molecule $\mathbf{x}$	Spectrum $\mathbf{s}$	Predicted MS/MS spectrum

Unless otherwise stated, the main loss weights are

$$\lambda_{\text{mol}} = 1.0, \quad \lambda_{\text{spec}} = 1.0, \quad \lambda_{\text{fp}} = 1.0, \quad \lambda_{\text{mass}} = 10^{-6}, \quad \lambda_{\text{formula}} = 1.0.$$

For spectral prediction, the m/z and intensity terms are balanced to prevent high-magnitude m/z errors from dominating intensity learning.

C.5 Fine-tuning protocol

For downstream fine-tuning, we use task-specific observation masks corresponding to de novo annotation, retrieval, or spectrum simulation. Unless otherwise stated, the pretrained backbone and task heads are fine-tuned end to end on paired examples from the corresponding training split. In

Table 5: Main model hyperparameters.

Hyperparameter	Value
Number of layers	8
Hidden dimension	512
Feed-forward dimension	2048
Attention heads	8
Dropout	0.1
Spectrum embedding dimension	256
Time embedding dimension	256
Condition vector dimension	256
Molecular vocabulary size	8000
Maximum molecular sequence length L	190
Retained spectral peaks N	128
Fingerprint dimension D	4096
Mass basis kernels	128
Intensity basis kernels	64
Relative m/z bins B	512
Integration steps T	50
Candidate samples per spectrum	100

low-data settings, we use a smaller learning rate for the pretrained backbone and a larger learning rate for task-specific heads. When needed, we apply L^2-SP regularization,

$$\Omega(\theta) = \frac{\lambda}{2}\|\theta - \theta_0\|_2^2,$$

where θ_0 denotes the pretrained parameters.

D Evaluation Protocols

D.1 De novo annotation

For de novo annotation, the input is an MS/MS spectrum and the output is a ranked list of molecular candidates. For $k \in \{1, 10\}$, we evaluate

$$\mathrm{Acc@}k = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \mathbb{I}\left[G_i \in \{\hat{G}_{i,1}, \ldots, \hat{G}_{i,k}\}\right],$$

$$\mathrm{MCES@}k = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \min_{j \le k} \mathrm{MCES}(G_i, \hat{G}_{i,j}), \qquad \mathrm{Tan@}k = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \max_{j \le k} \mathrm{Tanimoto}(G_i, \hat{G}_{i,j}).$$

Generated molecular strings are parsed and canonicalized with RDKit when possible. Invalid or unparsable outputs are retained for validity diagnostics. For benchmark ranking metrics, invalid outputs are excluded or assigned non-matching status according to the corresponding benchmark protocol.

D.2 Molecular retrieval

For retrieval, the model predicts a fingerprint from the query spectrum. Candidate molecules are ranked by similarity between predicted and candidate fingerprints. We report hit rate at k and MCES@1:

$$\mathrm{HR@}k = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \mathbb{I}\left[\mathrm{rank}(G_i) \le k\right].$$

D.3 Spectrum simulation

For spectrum simulation, the model predicts an MS/MS spectrum from a molecule. We evaluate cosine similarity, Jensen–Shannon similarity, and retrieval hit rates induced by simulated spectra,

following the MassSpecGym protocol. Predicted and reference spectra are preprocessed consistently before metric computation.

D.4 Point-estimate reporting

Several large-scale experiments are reported as point estimates under fixed training and evaluation protocols. We do not report repeated-run error bars for all pretraining and fine-tuning experiments because full repeated modality-incomplete pretraining is computationally expensive. When confidence intervals are available from official benchmark reports or directly reproduced evaluations, we report them; otherwise, reported values should be interpreted as point estimates under the stated protocol.

E Flow Derivations and Inference Algorithms

This section provides derivations and numerical update rules used by the implementation. We do not present these derivations as formal theoretical guarantees.

E.1 Discrete probability velocity

We use a mask-based discrete probability path

$$p_t(x_i \mid x_{1,i}) = \kappa(t)\delta_{x_{1,i}} + (1-\kappa(t))p_{\text{mask}}.$$

The generating velocity is parameterized by the predicted clean-token distribution $\hat{\mathbf{p}}_1$. For the convex path, the marginal probability velocity is

$$u_t^i(x^i, \mathbf{x}_t) = \frac{\dot{\kappa}(t)}{1-\kappa(t)}\left[p_{1|t}(x^i \mid \mathbf{x}_t) - \delta_{x_t^i}(x^i)\right].$$

Here $\mathbf{x}_t$ denotes the noisy molecular-token state; the symbol $\mathbf{z}$ is reserved for the full heterogeneous state in the main text.

E.2 Scheduler and numerical integration

We use a cubic scheduler

$$\kappa(t) = -2t^3 + 3t^2 + a(t^3 - 2t^2 + t) + b(t^3 - t^2),$$

with $a = 2.0$ and $b = 0.5$. Inference uses a discrete grid $t_k = k/T$, $k = 0, \ldots, T$, with $\Delta t = 1/T$. Continuous components are updated by Euler integration:

$$\mathbf{y}_{t_{k+1}} = \mathbf{y}_{t_k} + \Delta t\, v_\theta(\mathbf{z}_{t_k}, t_k, \cdot).$$

For molecular categorical probabilities, we maintain a simplex-valued distribution $\boldsymbol{\pi}_{k,i}$ for each position i. We use a clipped probability-velocity update

$$\tilde{\boldsymbol{\pi}}_{k+1,i} = \boldsymbol{\pi}_{k,i} + \eta_k u_{\theta,i}(\boldsymbol{\pi}_{k,i}, t_k, \cdot), \qquad \eta_k = \min\left(\Delta t, \rho\frac{1-\kappa(t_k)}{\dot{\kappa}(t_k)+\epsilon}\right),$$

with $\rho = 0.9$ and small $\epsilon > 0$. The updated vector is projected to the probability simplex and renormalized:

$$\boldsymbol{\pi}_{k+1,i} = \Pi_{\Delta_V}\left(\tilde{\boldsymbol{\pi}}_{k+1,i}\right).$$

This prevents negative probabilities and numerical overshoot near $t = 1$. At the final step, tokens are decoded by sampling or beam-style ranking from $\boldsymbol{\pi}_{T,i}$, followed by RDKit canonicalization and duplicate removal.

E.3 Synchronized hybrid inference

Algorithm 3 Synchronized hybrid inference

Input: observed components $\mathbf{z_o}$, task mask, metadata $\mathbf{c}$, steps T
Initialize molecular targets with mask-token distributions if $\mathbf{x}$ is generated.
Initialize spectral or fingerprint targets from Gaussian noise if $\mathbf{s}$ or $\mathbf{h}$ is generated.
for $k = 0, \ldots, T-1$ **do**
    Set $t_k = k/T$.
    Predict molecular clean-token probabilities and continuous velocities.
    Update continuous components by Euler integration.
    Update molecular categorical probabilities with clipped probability-velocity integration.
    Project molecular probabilities to the simplex and renormalize.
end for
Decode molecular tokens, convert generated peaks to spectra, and rank candidates when required.

F Additional Ablations and Diagnostics

F.1 Shared-backbone versus task-specific training

To isolate whether the unified formulation provides benefits beyond using the same architecture separately for each task, we compare three training settings. *Task-specific* trains separate MolSpecFlow models for de novo annotation, retrieval, and spectrum simulation using only the corresponding task mask. *Shared paired-only* trains one backbone with mixed task masks on paired molecule–spectrum data, without molecule-only or spectrum-only pretraining. *Full MolSpecFlow* uses the same shared backbone together with modality-incomplete pretraining.

Table 6: Shared-backbone versus task-specific training on MassSpecGym. Each setting uses the same MolSpecFlow architecture; the comparison isolates task-specific training, paired-only multi-interface sharing, and full modality-incomplete pretraining.

Setting	De novo		Retrieval		Simulation	
	Top-1 ↑	Top-10 ↑	HR@1 ↑	HR@20 ↑	Cos. ↑	HR@20 ↑
Task-specific models	1.23	2.37	8.36	40.94	0.45	70.89
Shared backbone, paired-only	2.08	4.38	12.57	43.15	0.54	75.74
Full MolSpecFlow	3.11	5.23	17.32	64.85	0.63	89.54

The shared paired-only model improves over task-specific training on all three interfaces, indicating that multi-interface sharing is beneficial even without unpaired pretraining. Full MolSpecFlow gives the best results across de novo annotation, retrieval, and spectrum simulation, supporting the role of modality-incomplete pretraining in strengthening the shared backbone.

F.2 Expected-property loss ablation

Table 7: Ablation of expected mass and formula losses on MassSpecGym de novo annotation.

Variant	Top-1 ↑	Top-10 ↑
Full model	**3.11**	**5.23**
w/o mass loss	2.88	4.82
w/o formula loss	2.48	4.18
w/o both	2.05	3.48

Table 8: Ablation of representation and spectral-encoding components on MassSpecGym de novo annotation.

Variant	Top-1 ↑	Top-10 ↑	MCES ↓	Tan. ↑
Full model	**3.11**	**5.23**	**17.32**	**0.34**
w/o mass emb.	2.15	3.92	21.45	0.28
w/o formula emb.	1.85	3.45	23.80	0.26
w/o relative m/z bias	2.68	4.75	18.92	0.31
w/o all above	1.25	2.10	28.15	0.19

Table 9: Retrieval interface ablation on MassSpecGym.

Interface	HR@1 ↑	HR@5 ↑	HR@20 ↑	MCES@1 ↓
MS→graph→FP	9.70	24.80	47.50	16.90
MS→FP	**17.32**	**39.93**	**64.85**	**14.05**

F.3 Representation-component ablation

F.4 Retrieval-interface ablation

F.5 Validity and consistency diagnostics

Table 10: Validity and consistency diagnostics in the predictive de novo setting.

Diagnostic	Value
Syntactic validity	0.9932
RDKit validity	0.9286
No AtomValenceException rate	0.9355
AtomValenceException / all samples	6.45%
AtomValenceException / invalid samples	90.42%
Formula-count error, mean	0.0522
Formula-count error, median	0.0000
Neutral mass–precursor m/z diff., mean	1.223 Da
Neutral mass–precursor m/z diff., median	1.007 Da

G Multi-Path Representation–Generation Trade-off

Increasing path diversity reduces dependence on a single molecular serialization and improves representation-oriented spectrum modeling, but it also increases the entropy of the sequence-generation target and degrades direct de novo molecular annotation in the current sequence-decoding setup. Accordingly, we use $K = 1$ for direct de novo annotation and $K = 10$ for the main representation-oriented spectrum-simulation setting reported in Table 2. Here K denotes the number of randomized molecular paths, not the molecular sequence length.

H Runtime and Compute Resources

Training compute. The reported experiments were trained on NVIDIA A100 GPUs. The modality-incomplete pretraining stage used approximately 8 A100 GPUs for about 20 days, corresponding to roughly 3,840 GPU-hours. Task-specific fine-tuning was substantially cheaper: de novo molecular annotation required approximately 18 hours on 8 A100 GPUs, molecular retrieval/fingerprint fine-tuning required approximately 16 hours on 8 A100 GPUs, and spectrum simulation fine-tuning required approximately 20 hours on 8 A100 GPUs. These values are approximate wall-clock estimates from the final training runs rather than a complete accounting of all pilot, debugging, and failed runs. GPU memory usage was not systematically logged, but all reported training runs fit within the available A100 GPU memory on our cluster.

Table 11: Representation–generation trade-off under single-path and multi-path molecular views. The $K = 10$ spectrum result corresponds to the main MolSpecFlow spectrum-simulation setting in Table 2.

Setting	De novo Top-1 Acc. (%) ↑	Spectrum HR@1 (%) ↑
Single path, $K = 1$	**3.11**	25.48
Main multi-path setting, $K = 10$	0.00	**55.32**

Inference runtime. All runtime measurements are conducted on a single NVIDIA V100 GPU using the same evaluation scripts and query-level protocol. For de novo annotation, MolSpecFlow with $K = 1$ requires 1042.3 ms/query, compared with 2884.5 ms/query for MADGEN and 7286.2 ms/query for DiffMS. For spectrum simulation, MolSpecFlow requires 1456.6 ms/query, compared with 1849.9 ms/query for FraGNNet. These measurements include model forward passes and decoding or sampling overhead and are intended as practical wall-clock comparisons rather than theoretical complexity claims.

Table 12: Inference cost on a single NVIDIA V100 GPU.

Task	Model	ms/query ↓	queries/s ↑
De novo	MolSpecFlow	**1042.3**	**0.959**
De novo	MADGEN	2884.5	0.347
De novo	DiffMS	7286.2	0.137
Simulation	MolSpecFlow	**1456.6**	**0.687**
Simulation	FraGNNet	1849.9	0.541

I Visualization of Generative Trajectories

In this section, we present representative token-level generation trajectories of MolSpecFlow. These examples illustrate how molecular tokens evolve from masked states to final decoded structures. They are intended as qualitative visualizations of inference dynamics rather than evidence of guaranteed synchronized reasoning or guaranteed chemical validity, and are not used for quantitative comparison.

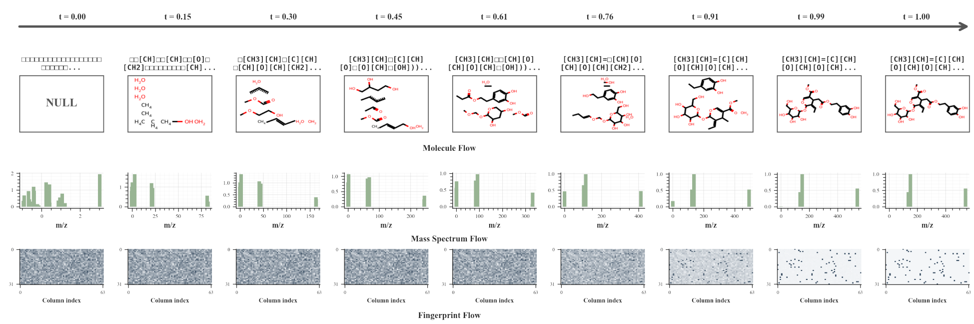
Sample 1: Joint Generation of Peptidomimetic Structure

Chemical Formula: $C_{33}H_{45}N_5O_5$

Total Mass (Calc): 590.3354 Da (**Precursor m/z:** 591.3427, **Charge:** 1)

Collision Energy: 30.0 eV

Generated SMILES: CCC(C)(C)C1C(=O)N2CCCC2C2(O)OC(NC(=O)C3CC4c5cccc6[nH]cc(c56)CC4N(C)C3)(C(C)C)C(=O)N12



Token Evolution Dynamics:

$t = 0.00$: [mask] . . . *(Pure Noise)*

$t = 0.15$: [<bos>] [[CH3]] [[CH2]] [mask] [mask] [)] [mask] [mask] [mask] [mask] [=] [mask] [)] [[N]] [mask] [mask] [[CH2]] [mask] [mask] [mask] [[OH]] [mask] [mask] [mask] [mask] [mask] [mask] [mask] [mask] [mask] [mask] . . . *(Early signals)*

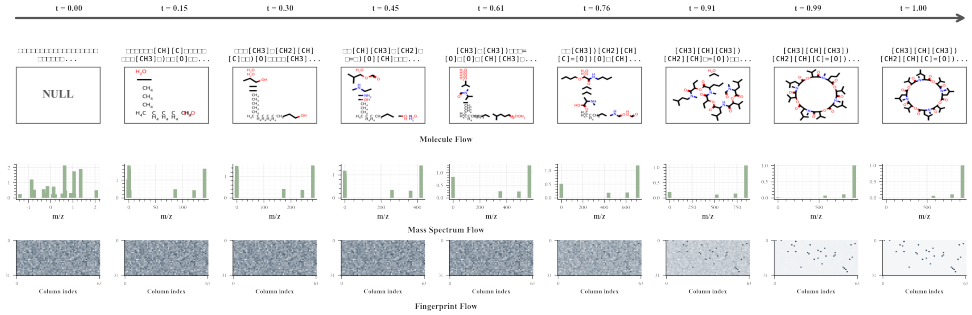
$t = 0.45$: [<bos>] [[CH3]] [mask] [[C]] [[CH3]] [)] [mask] [mask] [[CH]] [[C]] [mask] [[O]] [)] [[N]] [mask] [mask] [mask] [[CH]] [mask] [[C]] [[OH]] [mask] [[O]] [mask] [[NH]] [[C]] [=] . . . *(Scaffold emerging)*

$t = 0.76$: [<bos>] [[CH3]] [[CH2]] [[C]] [[CH3]] [)] [[CH3]] [)] [[CH]] [[C]] [=] [mask] [mask] [mask] [[CH2]] [[CH2]] [[CH2]] [[CH]] [5] [[C]] [mask] [)] [[O]] [[C]] [[NH]] [[C]] [=] [[O]] [)] . . . *(Refining)*

$t = 1.00$: [<bos>] [[CH3]] [[CH2]] [[C]] [[CH3]] [)] [[CH3]] [)] [[CH]] [[C]] [=] [[O]] [)] [[N]] [[CH2]] [[CH2]] [[CH2]] [[CH]] [5] [[C]] [[OH]] [)] [[O]] [[C]] [[NH]] [[C]] [=] [[O]] . . . *(Final)*

Sample 2: Joint Generation of Macrocyclic Peptide

Chemical Formula: $C_{40}H_{60}N_6O_8$
Total Mass (Calc): 752.4472 Da (**Precursor m/z:** 753.4545, **Charge:** 1)
Collision Energy: –
Generated SMILES: CC(C)C1C(=O)NC(C(=O)NC(C(=O)NC(C(=O)NC(C(=O)NC(C(=O)N1)C(C)C)C(C)C)C(C)C)C



Token Evolution Dynamics:

$t = 0.00$: [mask] ... *(Pure Noise)*

$t = 0.15$: [[CH]] [[CH]] [mask] [[C]] [mask] [[CH3]] [mask] [[O]] [mask] [[O]] [mask] [[NH]] [mask] [mask] [[CH]] [[CH3]] [mask] [[O]] [)] [mask] [[NH]] [mask] [[C]] [=] [[O]] ... *(Fragmented)*

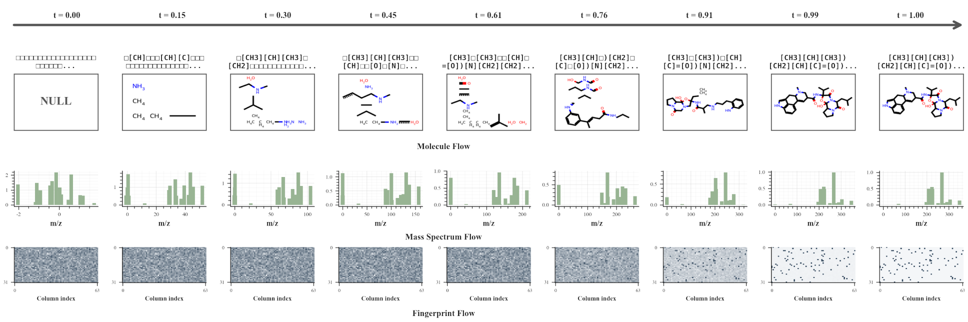
$t = 0.45$: [mask] [[CH3]] [[CH2]] [[CH]] [mask] [[CH3]] [[CH3]] [[CH2]] [[CH]] [=] [[O]] [[O]] [[CH]] [[CH]] [[CH3]] [=] [[O]] [)] [[NH]] [[CH]] [[C]] [=] [[O]] [)] ... *(Backbone forming)*

$t = 0.76$: [<bos>] [[CH3]] [)] [[CH2]] [[CH]] [[C]] [=] [[O]] [)] [[O]] [[CH]] [mask] [[CH]] [[CH3]] [[CH3]] [)] [[CH]] [[CH3]] [)] [[CH2]] [[CH]] [[C]] [=] [[O]] ... *(Ring closure)*

$t = 1.00$: [<bos>] [[CH3]] [[CH]] [[CH3]] [)] [[CH2]] [[CH]] [[C]] [=] [[O]] [)] [[N]] [[CH2]] [[CH]] [[C]] [=] [[O]] [)] [[NH]] [[CH]] [[C]] [=] [[O]] [)] [[NH]] ... *(Final)*

Sample 3: Joint Generation of Glycoside Derivative

Chemical Formula: $C_{25}H_{32}O_{13}$
Total Mass (Calc): 539.1755 Da (**Precursor m/z:** 540.1827, **Charge:** 1)
Collision Energy: –
Generated SMILES: CC=C1C(OC2OC(CO)C(O)C(O)C2O)OC=C(C(=O)OC)C1CC(=O)OCCc1ccc(O)c(O)c1



Token Evolution Dynamics:

$t = 0.00$: [mask] ...*(Pure Noise)*

$t = 0.15$: [mask] [mask] [[CH]] [mask] [mask] [[CH]] [mask] [mask] [[O]] [mask] [[CH2]] [mask] [mask] [mask] [mask] [mask] [mask] [mask] [mask] [mask] [[CH]] [mask] [[OH]] [mask] [)] [mask] [mask] [)] [mask] [mask] [mask] [mask] [mask] [=] ...*(High entropy)*

$t = 0.45$: [<bos>] [[CH3]] [[CH]] [mask] [[C]] [[CH]] [[O]] [mask] [[O]] [[CH]] [mask] [[OH]] [)] [)] [[CH]] [mask] [mask] [[CH]] [[OH]] [)] [[CH]] [6] [[OH]] [)] [)] [)] [mask] [)] [)] [)] ...*(Sugar moiety forming)*

$t = 0.76$: [<bos>] [[CH3]] [[CH]] [=] [mask] [[CH]] [[O]] [[CH]] [[O]] [[CH]] [[CH2]] [[OH]] [)] [)] [[CH]] [[OH]] [)] [[CH]] [[OH]] [)] [[CH]] [6] [[OH]] [)] [)] [)] [)] [)] [)] [)] [[O]] ...*(Stabilizing)*

$t = 1.00$: [<bos>] [[CH3]] [[CH]] [=] [[C]] [[CH]] [[O]] [[CH]] [[O]] [[CH]] [[CH2]] [[OH]] [)] [)] [[CH]] [[OH]] [)] [[CH]] [[OH]] [)] [[CH]] [6] [[OH]] [)] [)] [)] [)] [)] [)] [)] [[O]] ...*(Final)*

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction describe MolSpecFlow as a modality-incomplete molecular–spectral learning framework, state the supported task interfaces, and avoid claiming that MS/MS structure elucidation is solved. The reported claims are tied to MassSpecGym and NPLIB1 results and are further qualified in the Discussion and Limitations section.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The Discussion and Limitations section discusses data scarcity and licensing constraints, measurement heterogeneity, the non-identifiability of MS/MS annotation, benchmark coverage, and the need to treat MolSpecFlow as a candidate-generation and prioritization tool rather than an autonomous source of definitive molecular identification.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper does not present formal theoretical theorems or claims. The appendix provides derivations and algorithms for the discrete probability path, scheduler, and synchronized hybrid inference used by the method.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material.
- Any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The main paper and appendix describe the datasets, splits, task protocols, model architecture, mask-conditioned training objective, optimization settings, metrics, inference procedures, runtime setting, and ablation protocols. Public benchmarks are used for the main evaluation, and baseline numbers are reported only when the protocol is directly comparable.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: The paper uses public benchmarks and describes the datasets, preprocessing, model architecture, training objectives, evaluation metrics, and inference protocols in the main text and appendix. We do not provide a public repository at initial submission while the codebase is being cleaned and anonymized; we plan to release code, configuration files, and reproduction instructions upon acceptance, subject to third-party dataset licenses and redistribution constraints.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer.
   - The instructions should contain the exact command and environment needed to run to reproduce the results.
   - The authors should provide instructions on data access and preparation.
   - The authors should provide scripts to reproduce all experimental results for the proposed method and baselines.
   - At submission time, to preserve anonymity, the authors should release anonymized versions if applicable.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: The main paper states the evaluation benchmarks, task protocols, and metrics, while the appendix provides dataset preprocessing, architecture hyperparameters, optimizer settings, loss weights, fine-tuning protocol, inference details, and ablation settings.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: The paper reports standardized benchmark metrics and ablations under fixed evaluation protocols, but does not report repeated-run error bars or statistical significance tests for all main experiments. Large-scale modality-incomplete pretraining and multiple task-specific fine-tuning runs are computationally expensive; we therefore focus on matched benchmark protocols, consistent metric reporting, and conservative interpretation of small gains. Where confidence intervals are available from official benchmark reports or reproduced evaluations, they are reported; otherwise, results should be interpreted as point estimates under the stated training and evaluation settings.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated.
- The method for calculating the error bars should be explained.
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Appendix H reports the GPU type, number of GPUs, approximate wall-clock time, and estimated GPU-hours for the reported final pretraining and fine-tuning runs. The modality-incomplete pretraining stage used approximately 8 NVIDIA A100 GPUs for about 20 days, and task-specific fine-tuning required approximately 16–20 hours on 8 NVIDIA A100 GPUs depending on the task. GPU memory usage was not systematically logged, and the reported accounting excludes additional pilot, debugging, and failed runs.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper.

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The work uses existing molecular and MS/MS datasets and does not involve human-subject experiments, personally identifiable data, or deployment in safety-critical settings. The paper discusses data licensing, benchmark reproducibility, limitations of MS/MS-based structure annotation, and the need for expert validation of generated molecular candidates.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity.

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The motivation discusses potential positive impact for metabolomics, environmental analysis, clinical analysis, and chemical characterization. The Discussion and Limitations section discusses risks and limitations related to ambiguous annotations, benchmark/data licensing, overconfident use of predicted molecular structures, and the need for uncertainty-aware candidate generation rather than deterministic point predictions.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses, fairness considerations, privacy considerations, and security considerations.
- The authors should consider possible harms that could arise when the technology is used as intended but gives incorrect results, and harms following from misuse.

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The work does not release a high-risk model for human-facing media generation, surveillance, personally identifiable data, or autonomous molecular optimization. MolSpecFlow is intended for MS/MS candidate annotation and prioritization; the main risks are scientific misannotation, overconfident use, and data-licensing constraints, which are discussed in the limitations and responsible-use discussion rather than requiring controlled-release safeguards.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards.
- Datasets that have been scraped from the Internet could pose safety risks.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [No]

Justification: The paper cites the datasets, benchmarks, software packages, and baselines used in the experiments and does not redistribute restricted third-party raw datasets. However, the current submission does not yet include a complete asset-by-asset license, version, and access-date audit for all external resources; we will provide this table together with the public artifact release.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license should be included for each asset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset should be provided.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The initial submission does not release a new public dataset, checkpoint, or code artifact. If code, checkpoints, or processed artifacts are released after acceptance, they will be documented with usage instructions and license information.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates.
- At submission time, remember to anonymize your assets if applicable.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The work does not involve crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution involves human subjects, as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The work does not involve crowdsourcing experiments or research with human subjects, so IRB or equivalent approval is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval or equivalent may be required for human subjects research.
- For initial submissions, do not include any information that would break anonymity.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method does not use an LLM as an important, original, or non-standard component. Any use of language models for writing, editing, or formatting does not affect the methodology, experiments, or scientific claims and therefore does not require declaration under this checklist item.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to the LLM policy in the NeurIPS handbook for what should or should not be described.