FlexMS: A Unified Public Benchmark for Molecule Tandem Mass Spectrum Prediction

Yunhua Zhong[1,3]*, **Yixuan Tang**[1], **Yifan Li**[1], **Pan Liu**[1],
Zhiwen Yang[1,4] **Zanyi Wang**[6], **Jie Yang**[5], **Jun Xia**[1,2]†
[1]The Hong Kong University of Science and Technology (Guangzhou).
[2]The Hong Kong University of Science and Technology.
[3]The University of Hong Kong. [4]Yangzhou University. [5]Fudan University.
[6] University of California, San Diego
Corresponding author(s). E-mail(s): junxia@hkust-gz.edu.cn;
Contributing authors: yunhuazhong@connect.hku.hk;tangyixuan@stu2022.jnu.edu.cn
yli994@connect.hkust-gz.edu.cn; panliu@hkust-gz.edu.cn;
zhiwenyang2004@gmail.com; yangjie23@m.fudan.edu.cn;

Abstract

Tandem mass spectrometry (MS/MS) is central to small molecule identification, but current deep learning systems for spectrum prediction still remain difficult to evaluate and deploy in practice. While novel architectures constantly claim state-of-the-art performance, inconsistent metadata conditioning and entangled preprocessing pipelines hinder fair architectural comparisons. Besides, existing evaluations are often restricted to curated datasets, failing to capture the heterogeneity and cross-domain shifts of real-world metabolomics. Furthermore, current benchmarks lack difficulty-aware diagnostics and leave blind to how models behave under specific compute or data constraints. To address this, we present FlexMS, a modular public-data benchmark framework that standardizes MS/MS prediction across public resources while keeping molecular encoders, metadata conditioning, predictor heads, and downstream retrieval under one protocol. FlexMS establishes a fair evaluation playground which significantly lowers the barrier for integrating new predictive tools. Rather than solely optimizing for average scores, FlexMS augments aggregate accuracy with difficulty-aware diagnostics, providing actionable guidance on model selection across different compute constraints, data scales, and downstream retrieval objectives. Ultimately, FlexMS provides the community with a reproducible standard to identify which algorithmic conclusions are stable and which operating points are most viable in practice.

1 Introduction

Tandem mass spectrometry (MS/MS) is central to molecule identification, drug discovery, and metabolism analysis because fragmentation patterns can provide rich clues about molecular structure [1, 2]. In practical analytical workflows, interpreting an unannotated spectrum frequently requires evaluating multiple plausible structural hypotheses, such as proposed drug metabolites, reaction byproducts or novel natural product derivatives. Computational spectrum prediction and retrieval serve as crucial diagnostic tools in this context because researchers can perform *in silico* retrieval to rank candidates against the experimental signal [3, 4, 5]. Formally, the computational task of *in silico* MS/MS spectrum prediction maps a molecular structure (e.g., SMILES or 2D graph) and

*This work was done during an internship at HKUST-GZ.
†Corresponding author.

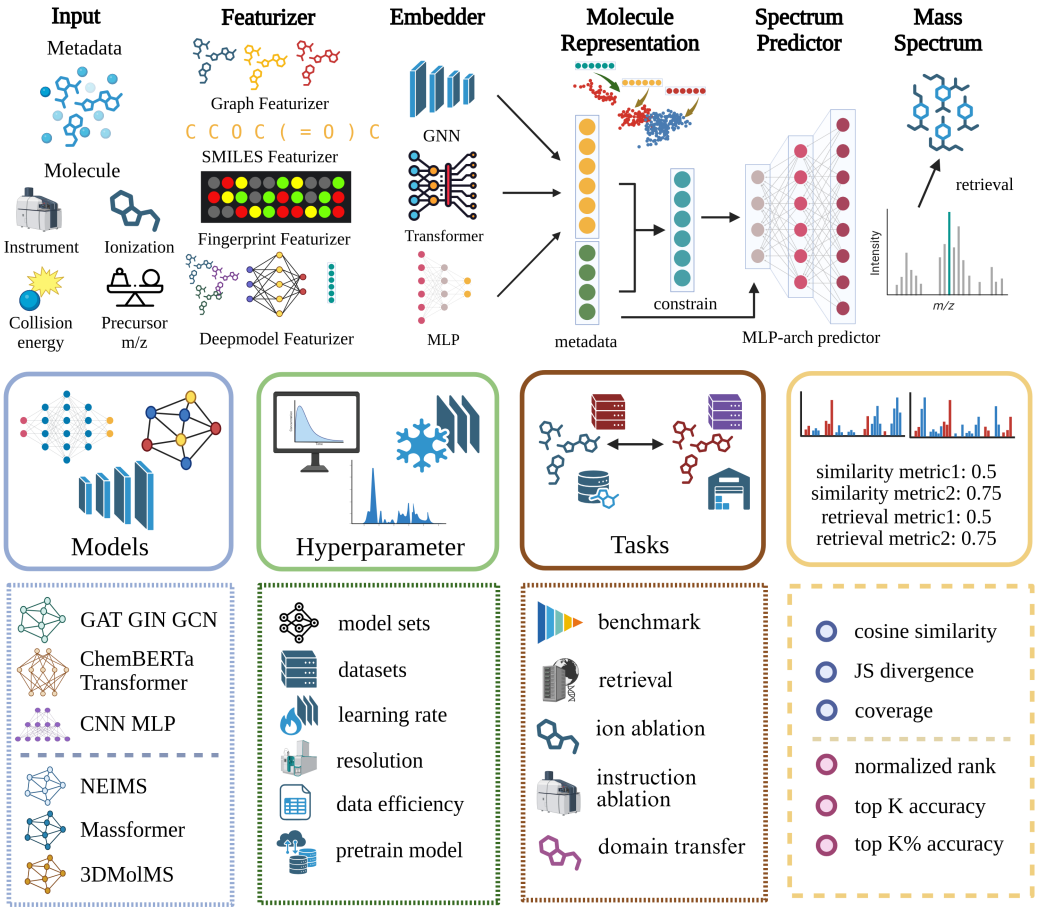

Figure 1: Main components of FlexMS. FlexMS framework takes molecules and associated metadata as inputs, employs various featurizers and embedders to generate molecular representations, and utilizes different multi-layer perceptron (MLP) architectures for spectrum prediction at specified bin resolutions. We assess the performance of various models, investigate the impact of different hyperparameters, and evaluate outcomes across diverse task scenarios. Comprehensive metrics are employed to quantify performance.

essential experimental metadata (e.g., collision energy, precursor ion and charge) to a simulated fragmentation spectrum. To make the output space tractable for neural networks, the continuous m/z (mass-to-charge ratio) axis is typically discretized into fixed bins, which transforms the task into a high-dimensional regression problem. The model predicts the relative intensity of fragment ions within each bin.

To solve this high-dimensional regression problem effectively, the field has increasingly shifted away from traditional rule-based empirical methods toward deep learning architectures[6, 7]. Because these DL models excel at learning complex fragmentation heuristics directly from large-scale mass spectral databases, the model space for spectrum prediction has expanded dramatically. Recent methods now span descriptor-based MLPs, sequence models over SMILES, graph neural networks and pretrained chemical foundation models [8, 9, 10, 11, 12]. In other words, the field no longer lacks modeling ideas.

This lack of a shared protocol is one of the main bottlenecks in modern spectrum prediction. A typical pipeline combines a molecular representation choice, a metadata-conditioning scheme, a molecule-to-spectrum prediction head, a target representation rule, and a downstream optimization and evaluation objective[12]. New methods are often assembled *de novo* rather than within a standardized framework, which makes benchmarking fragile[13]. In practice, model family, metadata

assumptions, split semantics, and evaluation targets often vary together, making reported gains difficult to attribute to model quality alone.

Current evaluation practice exhibits three concrete limitations. First, many studies provide incomplete protocol coverage, reporting only few datasets, a single task and a limited set of prediction metrics, even though different evaluation settings can support very different conclusions about generalization[13]. Second, comparisons are often entangled as reported gains may be confounded by preprocessing pipelines, metadata availability, or inaccessible assets, and some reported results depend on private commercial data or expensive substructure-annotation pipelines that limit fair comparison and efficient reuse [9, 12, 14, 15, 16, 17]. Finally, many models provide limited or hard practical guidance, as they typically optimize for the best average score but offer less insight into which model family remains preferable when data are limited, compute is constrained, or the downstream objective shifts from spectrum reconstruction to molecule retrieval.

Similar issues in chemical machine learning have motivated benchmark-centered resources where progress depends not only on stronger models, but also on stronger evaluation protocols [18, 19]. While recent works like MassSpecGym [20] have made significant strides in standardizing MS/MS datasets and defining prediction tasks, FlexMS complements these efforts by providing a more modular and diagnostic-oriented infrastructure. Specifically, while MassSpecGym focuses on curated tasks and splits, FlexMS adds modular encoder-predictor construction, multi-dataset diagnostics, compute-aware analysis, and retrieval evaluation. Since researchers often implement deeply coupled, custom scripts to conduct experiments, FlexMS addresses this critical gap by treating MS/MS prediction not merely as a leaderboard competition, but as a flexible and modular infrastructure problem over shared public resources. Rather than narrowly asking which model achieves the highest nominal score, FlexMS provides an aligned evaluation playground that yields actionable insights. By systematically mapping these aligned metrics to specific split difficulties, data regimes and compute budgets, FlexMS elevates difficulty-aware evaluation to serve as both a rigorous tool and a practical guide, helping practitioners to decide which architecture to deploy.

Contributions

- We introduce FlexMS as a modular public-data benchmark framework for MS/MS prediction, designed so that new datasets, models and evaluations can be inserted into one reproducible comparison space.
- We define a unified cross-dataset protocol, including official or scaffold-aware splits, metadata-aware conditioning, and shared similarity metrics.
- We contribute a difficulty-aware evaluation view for tandem MS prediction by coupling aggregate benchmark scores with molecular-similarity and spectral-entropy diagnostics, so that changes in model ranking can be interpreted rather than merely observed.
- We distill practical, constraint-aware insights to help practitioners identify optimal model choices and operating points based on specific data regimes, computational budgets and evaluating settings.

2 Related Work

Spectrum prediction methods. Deep learning has turned MS/MS prediction into a supervised mapping from molecular structure to spectral outputs. Existing approaches now span descriptor-based predictors such as NEIMS [11], graph-transformer systems such as MassFormer [12], geometry-aware models such as 3DMolMS [17], and autoregressive fragmentation predictors [14, 9]. At the representation level, researchers also draw from a broad encoder space including classical graph neural networks, attentive molecular encoders, and pretrained chemical foundation models [8, 21, 22, 23, 24, 25, 26]. This diversity is scientifically useful but makes fair comparison difficult: when datasets, split semantics, metadata assumptions, and preprocessing pipelines differ across papers, improvements in model design do not translate into stable conclusions about model choice.

Public spectral libraries. GNPS [27], MassBank [28], and NIST [15] are the most widely used MS/MS resources, but they differ substantially in scale, curation quality and licensing. GNPS offers the largest crowd-sourced collection with substantial heterogeneity with official preprocess version; MassBank provides cleaner but source-diverse reference spectra at smaller scale; NIST remains difficult to use in fully reproducible public benchmarking because of commercial restrictions. As a

result, many prediction papers rely on overlapping raw resources but adopt different filtering rules, metadata handling and train–test splits, so even nominally similar evaluations can correspond to materially different benchmark conditions.

Benchmarks and modular frameworks. Benchmark-centered resources in cheminformatics such as MoleculeNet [19], Therapeutics Data Commons [29], and ProteinGym [30] have shown that community progress depends not only on stronger models, but also on shared datasets, standardized splits and protocols that make conclusions comparable across studies. FlexMol [18] further showed that modular frameworks enabling systematic encoder–predictor combinations can make comparisons broader and fairer. Within MS/MS, MassSpecGym [20] unifies multiple annotation challenges around one curated resource and a demanding split design. These cross-dataset, protocol-centered works emphasize recommendations and difficulty-aware diagnostics rather than only reporting leaderboard.

3 FlexMS Framework

3.1 Mass Spectrum Prediction

Inputs for mass spectrum prediction consist of molecular entities and their associated experimental metadata [20]. Molecules are typically represented as SMILES strings or 2D/3D molecular graphs. The experimental metadata includes conditions such as collision energy (CE), instrument type, and precursor ion mode. The primary objective is to develop a mapping function $f : (X, M) \rightarrow Y$ that takes a molecular representation X and metadata embedding M as inputs, and predicts the interaction fragmentation pattern Y. In our binned prediction approach, the continuous mass-to-charge (m/z) domain is discretized into fixed-width bins (e.g., 1 Da), transforming the task into a multi-label regression problem where the model outputs intensity values for each predefined bin.

We provide a detailed ablation on how varying this bin width from 1 Da to 10 Da non-linearly affects different model architectures in *Appendix* B.

3.2 Framework

FlexMS provides a modular and user-friendly API for the dynamic construction of mass spectrum prediction models. The workflow consists of three primary stages: First, users select a specific task and load the corresponding dataset with designated split strategies (random or scaffold). Second, users customize their models by declaring and combining FlexMS components, which strictly decouple the `Embedder` for molecular representation, the `Predictor` for spectrum generation along with a `Metadata Encoder`. Third, FlexMS automatically handles data featurization, constructs the computational graph, and manages training and multi-metric evaluations.

3.2.1 Components

Featurizer This module handles the transformation of raw chemical inputs into specific mathematical formats required by downstream models. It generates diverse featurizations, including fixed-length physicochemical descriptors and binary fingerprints (e.g., Morgan, MACCS), tokenized sequences for language models (e.g., via byte-pair encoding), and topological graph objects with detailed atomic node and bond edge features for message-passing networks.

Metadata Encoder The Metadata Encoder standardizes continuous experimental variables such as Absolute Collision Energy (ACE) or Normalized Collision Energy (NCE), Discretized precursor m/z, and one-hot encodes categorical features such as instrument type and ionization mode. These processed attributes are then concatenated into a unified, dense context vector to capture the specific conditions influencing fragmentation patterns..

Embedder Distinct from static featurization, this module serves as the primary neural architecture that learns to map processed chemical inputs into fixed-dimensional latent representations. It acts as the core model backbone rather than merely fetching pre-computed embeddings, it introduces critical inductive biases into the representation learning process. FlexMS implements a diverse suite of embedders—ranging from traditional MLP-based architectures to sequence-based models and graph neural networks (GNNs).

Predictor Integrating the molecular representation and metadata context, this module acts as a decoder to generate the final mass spectrum. It is uniquely challenging because it requires mapping dense latent states into highly sparse m/z peak intensity distributions constrained by complex physical fragmentation rules. FlexMS adapts three distinct predictor paradigms: MassFormer, NEIMS, and MolMS, representing varying levels of model capacity and architectural complexity.

Detailed explanations of the featurization processes, network architecture detail, and the metadata embedding mechanisms are provided in the *Appendix* C.7.

3.3 Evaluation Metrics

To comprehensively evaluate model performance, FlexMS supports a wide array of metrics tailored for different downstream objectives. For generative spectrum prediction, it includes **Cosine Similarity**, **Jensen-Shannon Similarity** and **Spectral Coverage** to assess peak pattern resemblance and probability distribution alignment. For practical compound identification (retrieval tasks), FlexMS implements ranking-based metrics, including **Absolute / Normalized Rank**, and **Top-K / Top-K% Accuracy**. Mathematical formulations for all metrics are detailed in the *Appendix* C.

3.4 Supporting Datasets

FlexMS standardizes data loading and processing for major public mass spectrometry resources. Supported datasets include **GNPS** (a large-scale, crowdsourced library) [27], **MassBank** (high-resolution reference spectra) [28], **MassSpecGym** (a standardized ML benchmark) [20] and **NPLIB1** (a curated natural products collection) [31]. Furthermore, FlexMS supports challenge-based datasets like **CASMI** 2016 and 2022 [32] to evaluate real-world compound retrieval. Detailed dataset statistics and split evaluations are available in the *Appendix* C.1.

4 Results

4.1 Dataset Characterization in Mass Spectrum Prediction

Structural Diversity quantifies the distributional shifts of splits. In the rapidly evolving field of mass spectrometry, the choice of datasets plays a pivotal role in ensuring robust generalization. A key aspect of molecular datasets is structural diversity, which directly impacts the difficulty of prediction tasks [33]. We evaluated structural similarity using Tanimoto coefficients based on Morgan fingerprints [34], comparing scaffold splits against random splits (methods are in *Appendix* C.3.2.

As shown in Table 1, despite similar mean Tanimoto similarities across splits, two-sample Kolmogorov-Smirnov (KS) tests reveal fundamental differences in data leakage. Scaffold splits consistently produce elevated KS statistics with highly significant p-values, confirming substantial distributional shifts that rigorously challenge model generalization [35] (with $\log(\text{pval}) < -50$). To provide independent validation from a spectral perspective, we also analyzed the distribution of spectral entropy, which corroborates these structural findings (details are in *Appendix* A.1).

Table 1: Comparison of structural similarity metrics across different datasets and splits. MB denotes MassBank, and MSG denotes MassSpecGym. The suffixes "-R" and "-S" indicate random and scaffold splits. NPLIB1-0, 1, and 2 correspond to predefined splits of the NPLIB1 dataset. Tanimoto is the mean cross-set Tanimoto similarity. KS-stat is the Kolmogorov-Smirnov statistic comparing within-set vs cross-set similarity distributions. Log KS-pval is $\log(\text{p-value})$ from the KS test.

Train-Val	**GNPS-S**	**GNPS-R**	**MB-S**	**MB-R**	**MSG**	**NPLIB1-0**	**NPLIB1-1**	**NPLIB1-2**
Tanimoto	0.1059	0.1056	0.0795	0.0875	0.1002	0.1032	0.1052	0.1055
KS-stat	0.0528	0.0024	0.0831	0.0044	0.0345	0.0188	0.0051	0.0039
Log KS-pval	**-120.64**	-0.03	**-299.87**	-0.53	**-51.32**	**-14.97**	-0.84	-0.36
Train-Test	**GNPS-S**	**GNPS-R**	**MB-S**	**MB-R**	**MSG**	**NPLIB1-0**	**NPLIB1-1**	**NPLIB1-2**
Tanimoto	0.1045	0.1051	0.0788	0.0871	0.1005	0.1053	0.1052	0.1040
KS-stat	0.0326	0.0050	0.0794	0.0094	0.1002	0.0033	0.0020	0.0232
Log KS-pval	**-45.74**	-0.079	**-274.18**	-3.51	**-61.88**	-0.07	-0.01	**-6.23**

Table 2: Performance of different embedder-predictor combinations on the MassSpecGym dataset. Best, second best, and third best results in each row are denoted by **bold**, doubleline, and underline, respectively. "Trans", "AttnFP", "DL", "AllDes", "CBTa" denotes Transformers, AttentiveFP, Daylight, All-Descriptors MLP and ChemBERTa. In predictor, "Mass" denotes MassFormers.

Pred	**CNN**	**GCN**	**GIN**	**GAT**	**Trans**	**ESPF**	**DL**	**Attn**	**AllDes**	**CBTa**	**GFv2**
Cosine Similarity (Cos Sim)											
Mass	0.316	0.306	0.329	0.317	0.310	0.310	0.324	0.323	0.332	0.317	**0.357**
NEIMS	0.303	0.302	0.324	0.304	0.305	0.297	0.308	0.314	0.319	0.303	**0.354**
MolMS	0.226	0.240	0.247	0.240	0.224	0.212	0.229	0.248	0.241	0.233	**0.271**
Jensen-Shannon Similarity (JS Sim)											
Mass	0.477	0.484	0.494	0.483	0.471	0.475	0.481	0.489	0.491	0.480	**0.516**
NEIMS	0.455	0.468	0.461	0.458	0.456	0.454	0.462	0.461	0.470	0.460	**0.488**
MolMS	0.433	0.440	0.442	0.439	0.430	0.430	0.436	0.442	0.446	0.436	**0.457**

4.2 Empirical Study on Embedder and Prediction Models

GFv2 and Massformer has the highest performance. We systematically evaluated diverse combinations of embedder and predictor architectures. Table 2 shows the result. Among the tested embedders, GraphormerV2 (GFv2) [36] consistently achieved the highest performance across all predictor methods and resolutions, demonstrating superior capability in molecular representation. Graph-based methods showed competitive performance, while transformer-based and CNN methods exhibited moderate results. Among the predictors, MassFormer [12] achieved the highest overall metrics, followed by NEIMS [11] and MolMS [17]. These consistent performance hierarchies are not limited to MassSpecGym (*Appendix* D and Figure S2).

Critically, our results highlight strong synergistic relationships between specific molecular representation methods and prediction architectures. For instance, the optimal combination was the GFv2 embedder paired with the MassFormer predictor. MolMS predictors demonstrate superior performance when paired with GNN-based embedders, as both components operate within a graph-theoretic framework that naturally captures spatial relationships [21], and NEIMS shows enhanced effectiveness when combined with MLP-based embedders.

High cost of GFv2 makes efficient alternatives more practical under constraints. As detailed in Table 3, GFv2 requires substantially more time and more GPU memory. Training GFv2 on large datasets like GNPS can require several days even on A100 GPU. In resource-constrained environments, models like GIN or simple MLPs provide a much better trade-off. For memory-constrained scenarios or applications requiring high-throughput inference, we highly recommend the GIN or All-descriptors MLP.

Table 3: Computational efficiency comparison across model architectures. Time (seconds) and peak GPU memory (MB) per forward pass (in training) are measured during training with a batch size of 512. Asterisks (*) indicate pretrained models.

Time	**CNN**	**GCN**	**GIN**	**GAT**	**Trans**	**ESPF**	**DL**	**Attn**	**AllDes**	**CBTa***	**GFv2***
Mass	0.532	0.656	0.658	0.823	0.270	0.395	0.394	0.690	0.351	3.493	123.5
NEIMS	0.471	0.581	0.564	0.594	0.212	0.324	0.355	0.527	0.346	3.432	121.9
MolMS	0.473	0.467	0.498	0.538	0.202	0.318	0.312	0.523	0.344	3.415	122.6
Mem	**CNN**	**GCN**	**GIN**	**GAT**	**Trans**	**ESPF**	**DL**	**Attn**	**AllDes**	**CBTa***	**GFv2***
Mass	228.1	341.4	428.7	296.4	2882	538.1	672.9	1201	781.5	982.2	24403
NEIMS	122.8	112.1	144.2	122.0	2554	135.2	165.2	746.7	209.7	476.7	23105
MolMS	271.9	264.0	299.7	272.5	2717	290.7	323.7	906.7	375.8	643.9	23587

Table 4: Impact of data sparsity on embedder performance, evaluated on Cosine Similarity (↑).

Data	CNN	GCN	GIN	GAT	Trans	ESPF	DL	Attn	AllDes	CBTa	GFv2
25%	**0.254**	0.185	0.229	0.236	0.218	0.251	0.244	0.225	0.245	0.244	0.241
50%	0.293	0.236	0.281	0.288	0.266	0.280	0.292	0.285	0.286	0.290	**0.306**
100%	0.316	0.306	0.329	0.317	0.310	0.310	0.324	0.323	0.332	0.317	**0.357**

4.3 Impact of Hyperparameters, Data Sparsity and Pretrain

To provide actionable guidance for model deployment in resource-constrained environments, we investigated the impact of training data sparsity and learning rate variations on prediction performance.

Data Sparsity will influence the best model. We subsampled the MassSpecGym dataset into 25%, 50%, and 100% scaffold fractions. As shown in Table 4 and S2, models with complex inductive biases experienced severe performance degradation under the 25% regime, indicating a high propensity for overfitting. In contrast, fingerprint-based MLPs and models leveraging pre-trained backbones maintained robust performance even at low data volumes, though CNNs eventually plateaued as data scaled to 100% where GNNs could fully leverage their structural inductive biases.

Optimal learning rates depend strongly on dataset noise and curation quality. Table 5 shows the result. For heterogeneous datasets such as MassBank, lower learning rates (10^{-4}) are preferable for navigating complex optimization landscapes. In contrast, highly curated datasets such as NPLIB1 offer smoother loss surfaces, enabling faster convergence with higher learning rates (10^{-3}). *Appendix* Figure S4 provides the broader multi-metric comparison across datasets, predictors, and resolutions, while *Appendix* Figure S5 shows the corresponding retrieval-side sensitivity.

Molecular Pretraining matters. The lack of experimentally annotated MS/MS spectra makes self-supervised pretraining a highly attractive strategy. We assessed the performance of randomly initialized embedders against their pretrained counterparts across different datasets and split methods, and we choose MoleBERT because of its tiny but calibratedly designed technique [24].

Our results in Table 6 indicate that pretraining provides only marginal improvements on random splits, but more consistent and often larger gains on scaffold splits. **This suggests that self-supervised pretraining on large, unlabeled molecular corpora helps the embedder capture general chemical regularities, thereby improving generalization to unseen molecular scaffolds.** Furthermore, pretrained models converged faster during fine-tuning on the target mass spectrum datasets. A broader cross-dataset and multi-metric comparison is provided in *Appendix* Figure S6.

4.4 Cross-Domain Transfer Learning Analysis

Real-world applications often require applying spectrum-prediction models to spectra acquired under different experimental conditions from those seen during training. To evaluate robustness under such domain shifts, cross-domain transfer experiments were conducted on MassSpecGym across two axes: ionization mode ($[M+H]^+$ vs. $[M+Na]^+$) and instrument type (Orbitrap vs. QTOF).

The results in Table 7 reveal asymmetric behavior that cannot be explained by sample size alone. MassSpecGym is strongly imbalanced across these domains: approximately 85% of spectra are $[M+H]^+$ and 15% are $[M+Na]^+$. **Despite much smaller source domain, models trained on $[M+Na]^+$ spectra often transfer better to $[M+H]^+$ spectra than the reverse direction. Previous papers demonstrated that sodium-adducted ions exhibit markedly different fragmentation behavior [37]**, with a low similarity to [M+H]+ due to charge-remote fragmentation and coordination-based stabilization mechanisms. We hypothesize that training on these highly complex, remote-cleavage patterns forces the model to learn deeper and more robust topological representations of the molecular graph, which acts as a strong regularization mechanism,

A similar asymmetry appears across instrument, where roughly 75% are on Orbitrap instruments and 25% on QTOF. **Models trained on high-resolution Orbitrap spectra generally transfer better to QTOF spectra than QTOF-trained models transfer to Orbitrap spectra, which is consistent with the higher mass accuracy and more spectral detail of Orbitrap measurements [38].**

Table 5: Impact of learning rate on embedder performance across three datasets (1 Da resolution). Performance is evaluated via Cosine Similarity.

Method	LR	Embedder									
		CNN	GCN	GIN	GAT	Trans	ESPF	DL	Attn	AllDes	CBTa
MB											
MF	10^{-3}	0.534	0.504	0.516	0.536	0.478	**0.460**	0.445	**0.522**	**0.468**	0.521
	10^{-4}	**0.546**	**0.531**	**0.538**	**0.544**	**0.497**	0.455	**0.461**	0.521	0.467	**0.537**
NEIMS	10^{-3}	0.454	**0.506**	0.498	**0.511**	0.445	0.408	0.442	0.398	0.405	0.461
	10^{-4}	**0.488**	0.503	**0.519**	0.508	**0.477**	**0.424**	**0.456**	**0.469**	**0.451**	**0.492**
MolMS	10^{-3}	0.433	0.445	0.451	**0.491**	0.385	0.348	0.362	0.400	0.371	0.420
	10^{-4}	**0.449**	**0.451**	**0.474**	0.465	**0.410**	**0.350**	**0.371**	**0.434**	**0.371**	**0.451**
MSG											
MF	10^{-3}	0.301	0.312	0.328	0.316	**0.319**	**0.311**	0.308	**0.324**	**0.316**	0.326
	10^{-4}	**0.306**	**0.324**	**0.332**	**0.323**	0.317	0.310	**0.310**	0.317	0.316	**0.329**
NEIMS	10^{-3}	0.292	0.298	0.316	0.311	0.296	**0.297**	**0.306**	0.296	0.300	0.320
	10^{-4}	**0.302**	**0.308**	**0.319**	**0.314**	**0.303**	0.297	0.305	**0.304**	**0.303**	**0.324**
MolMS	10^{-3}	0.233	0.221	0.236	0.239	0.232	**0.214**	0.222	0.239	**0.227**	0.233
	10^{-4}	**0.240**	**0.229**	**0.241**	**0.248**	**0.233**	0.212	**0.224**	**0.240**	0.226	**0.247**
NPLIB1											
MF	10^{-3}	0.557	0.529	0.561	0.549	**0.539**	**0.526**	0.530	**0.561**	0.527	**0.560**
	10^{-4}	**0.560**	**0.545**	**0.570**	**0.557**	0.534	0.524	**0.538**	0.558	**0.533**	0.550
NEIMS	10^{-3}	**0.552**	**0.543**	**0.559**	**0.552**	**0.545**	**0.517**	**0.538**	**0.551**	**0.510**	**0.555**
	10^{-4}	0.512	0.526	0.526	0.525	0.508	0.485	0.513	0.491	0.486	0.503
MolMS	10^{-3}	**0.512**	**0.487**	0.500	**0.517**	**0.501**	**0.481**	0.452	**0.510**	**0.475**	0.489
	10^{-4}	0.498	0.485	**0.526**	0.513	0.497	0.474	**0.503**	0.476	0.473	**0.490**

These results indicate that domain coverage should be evaluated both on amount of spectra, the information content and transferability of the source domain. Extended transfer diagnostics, including direction-wise visual comparisons and critical-difference analysis across embedders, are provided in the *Appendix* D.3. Same-domain simulations are also detailed in *Appendix* D.4.

Table 6: Performance comparison at 1 Bin Resolution with or without pretrained MoleBERT model.

Method	**MB (Random)**		**MB (Scaffold)**		**MassSpecGym**		**NPLIB1**	
	Pretrain	w/o Pre	Pretrain	w/o Pre	Pretrain	w/o Pre	Pretrain	w/o Pre
Mass	**0.5363**	0.5202	**0.4124**	0.3844	**0.3406**	0.3138	**0.5939**	0.5337
NEIMS	**0.5201**	0.5164	**0.3950**	0.3792	**0.3469**	0.3127	**0.5603**	0.5374
MolMS	**0.4731**	0.4508	**0.3389**	0.3162	**0.2467**	0.2215	**0.5265**	0.4693

4.5 Retrieval Benchmarks for Practical Identification

While generative metrics such as cosine similarity and Jensen–Shannon similarity assess the fidelity of predicted spectra, they do not directly measure utility in compound identification. FlexMS therefore includes a retrieval benchmark designed to evaluate whether predicted spectra are sufficiently discriminative to rank the correct molecular structure near the top of a realistic candidate list.

The retrieval benchmark follows CASMI-style identification settings on both CASMI 2016 and 2022. Candidate sets are generated from mass-based searches and then filtered for structural validity, after which each candidate molecule is scored by similarity between its predicted spectrum under the query

Table 7: Cross-domain transfer learning performance. Transfer directions are denoted using: Q (QTOF) and O (Orbitrap) for instrument types; H ($[M+H]^+$) and Na ($[M+Na]^+$) for ionization modes.

Pred	Dir.	CNN	GCN	GIN	GAT	Trans	ESPF	DL	Attn	AllDes	CBTa	GFv2
Instrument Types												
Mass	Q→O	0.255	0.228	0.268	0.274	0.262	0.200	0.244	0.267	0.273	0.216	**0.285**
	O→Q	0.303	0.294	0.323	0.332	0.314	0.184	0.324	0.315	0.311	0.303	**0.345**
NEIMS	Q→O	0.260	0.242	**0.276**	0.243	0.259	0.252	0.245	0.263	0.252	0.251	0.269
	O→Q	0.318	0.289	0.326	0.316	0.316	0.312	0.317	0.324	0.328	0.320	**0.351**
MolMS	Q→O	0.175	0.190	0.179	0.192	0.179	0.164	0.156	0.198	0.180	0.152	**0.242**
	O→Q	0.254	0.266	0.240	0.251	0.263	0.194	0.131	0.267	0.211	0.196	**0.302**
Ionization Modes												
Mass	H→Na	0.131	0.176	0.203	0.209	**0.212**	0.180	0.183	0.204	0.166	0.179	0.171
	Na→H	0.250	0.196	**0.273**	0.258	0.210	0.252	0.245	0.269	0.260	0.250	0.226
NEIMS	H→Na	0.219	0.176	0.227	0.212	0.228	0.221	0.222	**0.234**	0.211	0.216	0.227
	Na→H	0.241	0.209	0.239	0.235	**0.242**	0.224	0.233	0.231	0.226	0.236	0.233
MolMS	H→Na	0.094	0.103	0.106	0.106	**0.117**	0.090	0.081	0.110	0.085	0.093	0.107
	Na→H	0.206	0.205	0.205	0.211	0.211	0.204	0.212	**0.213**	0.208	0.199	0.212

metadata and the observed query spectrum. Full protocol details and candidate-pool statistics and metric definitions are provided in the *Appendix* C.2.

Models with strong reconstruction metrics generally remain competitive in retrieval, but the relationship is not perfectly linear. Still, GNN-based embedders such as GFv2 achieve the strongest normalized-rank performance across predictors in CASMI 2016 (Table 8), indicating better discrimination ability. Specifically, we discuss GCN in *Appendix* B. By contrast, simpler descriptor-based baselines can remain competitive on average reconstruction scores while producing less selective rankings. Extended evaluations on the CASMI 2022 dataset and respective analysis further corroborate these trends (see *Appendix* C.2 and Table S4).

Table 8: Retrieval benchmark performance evaluated on the CASMI 2016 dataset. Metrics include Recall@1 (R@1), Recall@10 (R@10), and Normalized Rank (NormR ↓) across candidate pools. Best, second best, and third best results are denoted by **bold**, doubleline, underline, respectively.

	MassFormer			NEIMS			MolMS		
	R@1	**R@10**	**NormR↓**	**R@1**	**R@10**	**NormR↓**	**R@1**	**R@10**	**NormR↓**
CNN	0.0354	0.1869	0.3283	0.0202	0.1313	0.3385	0.0152	0.0758	0.3957
GCN	0.1667	0.5455	0.1470	0.0808	0.5354	0.1312	0.1465	0.5303	0.1969
GIN	0.1061	0.4949	0.1381	0.0808	0.4798	0.1457	0.0960	0.4394	0.1838
GAT	0.0859	0.4848	0.1522	0.0707	0.3889	0.1756	0.0354	0.3030	0.2077
Trans	0.0354	0.1212	0.3699	0.0253	0.1364	0.3523	0.0051	0.0707	0.4284
ESPF	0.0354	0.1212	0.3809	0.0101	0.1010	0.3636	0.0051	0.0909	0.3860
DL	0.1515	0.4495	0.1988	0.1818	0.5505	0.1487	0.1465	0.5051	0.1611
Attn	0.1313	0.6111	0.1286	0.0808	0.5253	0.1267	0.1263	0.5909	**0.1566**
AllDes	**0.1970**	0.5808	0.1696	**0.2071**	0.5505	0.1324	0.1364	0.4394	0.1852
CBTa	0.0354	0.1162	0.3394	0.0303	0.1465	0.3135	0.0101	0.1010	0.4141
GFv2	0.1515	**0.6667**	**0.1022**	**0.2071**	**0.6414**	**0.1133**	**0.1717**	**0.6061**	0.1574

5 Conclusion and Limitations

In this work, we introduced FlexMS, a modular and comprehensive benchmarking framework designed to standardize the evaluation of deep learning-based mass spectrum prediction tools. By decoupling molecular prediction pipeline and systematically evaluating them across diverse public metabolomics resources, we demonstrated the model accuracy is highly contingent on the underlying evaluation protocol. Our empirical results establish that while GraphormerV2 paired with MassFormer

consistently achieves the strongest performance under our unified protocol, this performance requires substantial computational overhead. By rigorously evaluating cross-domain robustness and validating distribution shifts through structural and spectral metrics, we demonstrate that realistic problem settings, such as scaffold splits and instrument transfer, are vital for guiding generalizable and practical model selection.

While our framework systematically evaluates a wide range of representative components, it does not currently cover all the latest state-of-the-art methods or advanced training tricks in this field. However, the contribution of FlexMS is not merely identifying a single winning architecture but rather providing a reproducible ecosystem. This framework empowers researchers to move beyond entangled comparisons, enabling them to rigorously benchmark future innovations and identify the most practical operating points.

References

[1] Ruedi Aebersold and Matthias Mann. Mass-spectrometric exploration of proteome structure and function. *Nature*, 537(7620):347–355, 2016.

[2] Edmond De Hoffmann and Vincent Stroobant. *Mass spectrometry: principles and applications*. John Wiley & Sons, Chichester, England, 2007.

[3] 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.

[4] Tobias Kind, Hiroshi Tsugawa, Tomas Cajka, Yan Ma, Zijuan Lai, Sajjan S Mehta, Gert Wohlgemuth, Dinesh Kumar Barupal, Megan R Showalter, Masanori Arita, et al. Identification of small molecules using accurate mass ms/ms search. *Mass spectrometry reviews*, 37(4):513–532, 2018.

[5] David S Wishart, Yannick Djoumbou Feunang, Ana Marcu, An Chi Guo, Kevin Liang, Rosa Vázquez-Fresno, Tanvir Sajed, Daniel Johnson, Carin Li, Naama Karu, et al. Hmdb 4.0: the human metabolome database for 2018. *Nucleic acids research*, 46(D1):D608–D617, 2018.

[6] Fei Wang, Dana Allen, Siyang Tian, Eponine Oler, Vasuk Gautam, Russell Greiner, Thomas O Metz, 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.

[7] Christoph Ruttkies, Emma L Schymanski, Sebastian Wolf, Juliane Hollender, and Steffen Neumann. Metfrag relaunched: incorporating strategies beyond in silico fragmentation. *Journal of cheminformatics*, 8(1):3, 2016.

[8] Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. Chemberta: large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*, 2020.

[9] Samuel Goldman, Janet Li, and Connor W Coley. Generating molecular fragmentation graphs with autoregressive neural networks. *Analytical Chemistry*, 96(8):3419–3428, 2024.

[10] Maya Hirohara, Yutaka Saito, Yuki Koda, Kengo Sato, and Yasubumi Sakakibara. Convolutional neural network based on smiles representation of compounds for detecting chemical motif. *BMC bioinformatics*, 19(Suppl 19):526, 2018.

[11] Jennifer N Wei, David Belanger, Ryan P Adams, and D Sculley. Rapid prediction of electron–ionization mass spectrometry using neural networks. *ACS central science*, 5(4):700–708, 2019.

[12] Adamo Young, Hannes Röst, and Bo Wang. Tandem mass spectrum prediction for small molecules using graph transformers. *Nature Machine Intelligence*, 6(4):404–416, 2024.

[13] Michael Murphy et al. Efficiently predicting high resolution mass spectra with graph neural networks. *arXiv preprint arXiv:2301.11419*, 2023.

[14] Samuel Goldman, John Bradshaw, Jiayi Xin, and Connor Coley. Prefix-tree decoding for predicting mass spectra from molecules. *Advances in Neural Information Processing Systems*, 36:48548–48572, 2023.

[15] National Institute of Standards and Technology. Nist 20 mass spectral library (nist/epa/nih), 2020. NIST/EPA/NIH Mass Spectral Library and NIST Tandem Mass Spectral Library.

[16] Lars Ridder, Justin J J van der Hooft, and Stefan Verhoeven. Automatic compound annotation from mass spectrometry data using magma. *Mass Spectrometry*, 3(Special Issue 2):S0033–S0033, 2014.

[17] Yuhui Hong, Sujun Li, Christopher J Welch, Shane Tichy, Yuzhen Ye, and Haixu Tang. 3dmolms: prediction of tandem mass spectra from 3d molecular conformations. *Bioinformatics*, 39(6):btad354, 2023.

[18] Sizhe Liu, Jun Xia, Lecheng Zhang, Yuchen Liu, Yue Liu, Wenjie Du, Zhangyang Gao, Bozhen Hu, Cheng Tan, Hongxin Xiang, and Stan Z. Li. Flexmol: A flexible toolkit for benchmarking molecular relational learning. *Advances in Neural Information Processing Systems*, 37, 2024.

[19] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

[20] Roman Bushuiev, Anton Bushuiev, Niek de Jonge, Adamo Young, Fleming Kretschmer, Raman Samusevich, Janne Heirman, Fei Wang, Luke Zhang, Kai Dührkop, et al. Massspecgym: A benchmark for the discovery and identification of molecules. *Advances in Neural Information Processing Systems*, 37:110010–110027, 2024.

[21] TN Kipf. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[22] Yu Rong, Yatao Bian, Tingyang Xu, Weiyang Xie, Ying Wei, Wenbing Huang, and Junzhou Huang. Self-supervised graph transformer on large-scale molecular data. *Advances in neural information processing systems*, 33:12559–12571, 2020.

[23] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[24] Jun Xia, Chengshuai Zhao, Bozhen Hu, Zhangyang Gao, Cheng Tan, Yue Liu, Siyuan Li, and Stan Z Li. Mole-bert: Rethinking pre-training graph neural networks for molecules. 2023.

[25] Zhaoping Xiong, Dingyan Wang, Xiaohong Liu, Feisheng Zhong, Xiaozhe Wan, Xutong Li, Zhaojun Li, Xiaomin Luo, Kaixian Chen, Hualiang Jiang, et al. Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. *Journal of medicinal chemistry*, 63(16):8749–8760, 2019.

[26] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[27] 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.

[28] Hisayuki Horai, Masanori Arita, Shigehiko Kanaya, Yoshito Nihei, Tasuku Ikeda, Kazuhiro Suwa, Yuya Ojima, Kenichi Tanaka, Satoshi Tanaka, Ken Aoshima, et al. Massbank: a public repository for sharing mass spectral data for life sciences. *Journal of mass spectrometry*, 45(7):703–714, 2010.

[29] Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W. Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. *arXiv preprint arXiv:2102.09548*, 2021.

[30] Pascal Notin, Aaron W. Kollasch, Daniel Ritter, Lood van Niekerk, Steffanie Paul, Helen Spinner, Nicholas J. Rollins, Ada Shaw, Roded Orenbuch, Ruth Weitzman, et al. Proteingym: Large-scale benchmarks for protein fitness prediction and design. *Advances in Neural Information Processing Systems*, 36, 2023.

[31] 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.

[32] Emma L Schymanski, Christoph Ruttkies, Martin Krauss, Céline Brouard, Tobias Kind, Kai Dührkop, Felicity Allen, Arpana Vaniya, Dries Verdegem, Sebastian Böcker, et al. Critical assessment of small molecule identification 2016: automated methods. *Journal of cheminformatics*, 9(1):22, 2017.

[33] Jonathan S Mason and Mark A Hermsmeier. Diversity assessment. *Current opinion in chemical biology*, 3(3):342–349, 1999.

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

[35] Ansgar Schuffenhauer, Peter Ertl, Silvio Roggo, Stefan Wetzel, Marcus A Koch, and Herbert Waldmann. The scaffold tree- visualization of the scaffold universe by hierarchical scaffold classification. *Journal of chemical information and modeling*, 47(1):47–58, 2007.

[36] Junhan Yang, Zheng Liu, Shitao Xiao, Chaozhuo Li, Defu Lian, Sanjay Agrawal, Amit Singh, Guangzhong Sun, and Xing Xie. Graphformers: Gnn-nested transformers for representation learning on textual graph. *Advances in Neural Information Processing Systems*, 34:28798–28810, 2021.

[37] Botao Liu, Zhifeng Tang, and Tao Huan. Adduct-induced variability in tandem mass spectrometry. *Analytical Chemistry*, 97(31):17058–17066, 2025.

[38] Estelle Deschamps, Valentina Calabrese, Isabelle Schmitz, Marie Hubert-Roux, Denis Castagnos, and Carlos Afonso. Advances in ultra-high-resolution mass spectrometry for pharmaceutical analysis. *Molecules*, 28(5):2061, 2023.

[39] 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.

[40] Timnit Gebru, Jamie Morgenstern, Brenda Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92, 2021.

[41] A Patrícia Bento, Anne Hersey, Eloy Félix, Greg Landrum, Anna Gaulton, Francis Atkinson, Louisa J Bellis, Marleen De Veij, and Andrew R Leach. An open source chemical structure curation pipeline using rdkit. *Journal of Cheminformatics*, 12(1):51, 2020.

[42] Harry L Morgan. The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. *Journal of chemical documentation*, 5(2):107–113, 1965.

[43] Joseph L Durant, Burton A Leland, Douglas R Henry, and James G Nourse. Reoptimization of mdl keys for use in drug discovery. *Journal of chemical information and computer sciences*, 42(6):1273–1280, 2002.

[44] Nikolaus Stiefl, Ian A Watson, Knut Baumann, and Andrea Zaliani. Erg: 2d pharmacophore descriptions for scaffold hopping. *Journal of chemical information and modeling*, 46(1):208–220, 2006.

[45] Craig A James. Daylight theory manual. *http://www. daylight. com/dayhtml/doc/theory/theory. toc. html*, 2004.

[46] Kazi Yasin Helal, Mateusz Maciejewski, Elisabet Gregori-Puigjane, Meir Glick, and Anne Mai Wassermann. Public domain hts fingerprints: design and evaluation of compound bioactivity profiles from pubchem's bioassay repository. *Journal of chemical information and modeling*, 56(2):390–398, 2016.

[47] Kexin Huang, Cao Xiao, Lucas Glass, and Jimeng Sun. Explainable substructure partition fingerprint for protein, drug, and more. In *NeurIPS learning meaningful representation of life workshop*, 2019.

[48] Minjie Wang, Da Zheng, Zihao Ye, Quan Gan, Mufei Li, Xiang Song, Jinjing Zhou, Chao Ma, Lingfan Yu, Yu Gai, et al. Deep graph library: A graph-centric, highly-performant package for graph neural networks. *arXiv preprint arXiv:1909.01315*, 2019.

[49] Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.

[50] Ulf W Liebal, An NT Phan, Malvika Sudhakar, Karthik Raman, and Lars M Blank. Machine learning applications for mass spectrometry-based metabolomics. *Metabolites*, 10(6):243, 2020.

[51] Kexin Huang, Tianfan Fu, Lucas M Glass, Marinka Zitnik, Cao Xiao, and Jimeng Sun. Deeppurpose: a deep learning library for drug–target interaction prediction. *Bioinformatics*, 36(22-23):5545–5547, 2020.

A Supplementary Material

A.1 Complementary Validation via Spectral Entropy Analysis

To provide independent validation from the spectral perspective, we analyzed the distribution of spectral entropy across dataset splits. Spectral entropy [39], defined as $H = -\sum_i p_i \ln(p_i)$ where p_i represents the normalized peak intensity, quantifies the complexity of fragmentation patterns in mass spectra. While the Tanimoto-based analysis examines structural similarity at the molecular input level, spectral entropy captures differences in fragmentation behavior at the output level.

As shown in Table S1, the mean spectral entropy varies across datasets. GNPS, MassBank, and MassSpecGym exhibit similar entropy levels ($H \approx 1.6$–1.7), whereas NPLIB1 shows notably higher entropy ($H \approx 2.8$), reflecting the complex fragmentation patterns characteristic of natural products [39]. More importantly, scaffold splits exhibit substantially larger distributional shifts in spectral entropy compared to random splits. For GNPS, the KS statistic under scaffold split is 4.3 times higher than random split (0.011) with significant p-values, and MassBank shows a 7.1-fold increase in KS statistics for scaffold versus random splits similarly.

Table S1: Spectral entropy distribution analysis across dataset splits. Mean Entropy denotes the average Shannon entropy (in nats) of training set spectra. KS-stat and Log KS-pval report the Kolmogorov-Smirnov test statistics and $\log_{10}$-transformed p-values comparing entropy distributions between splits. Bold values indicate highly significant distributional shifts.

Train-Val	**GNPS-S**	**GNPS-R**	**MB-S**	**MB-R**	**MSG**	**NPLIB1-0**	**NPLIB1-1**	**NPLIB1-2**
Mean Entropy	1.7275	1.7169	1.6374	1.7027	1.7396	2.8494	2.8486	2.8326
KS-stat	0.0595	0.0054	0.1245	0.0163	0.0247	0.0340	0.0716	0.0586
Log KS-pval	**-262.11**	-1.66	**-171.78**	-2.13	-9.02	-0.44	-2.67	-1.76
Train-Test	**GNPS-S**	**GNPS-R**	**MB-S**	**MB-R**	**MSG**	**NPLIB1-0**	**NPLIB1-1**	**NPLIB1-2**
Mean Entropy	1.7275	1.7169	1.6374	1.7027	1.7396	2.8494	2.8486	2.8326
KS-stat	**0.0492**	0.0115	**0.0995**	0.0140	0.0137	0.0386	0.0332	0.0507
Log KS-pval	**-172.84**	-7.94	**-88.19**	-1.48	-2.33	-1.44	-1.01	-2.91

B Extended Conclusions and Discussion

Beyond the primary findings presented in the main text, our extensive ablation studies within the FlexMS framework provide several nuanced insights into model behaviors under specific operational constraints.

Data Sparsity and Architecture Choice Our data ablation experiments highlight a critical trade-off between model capacity and data availability. We observed that high-capacity architectures with complex inductive biases, such as deep Graph Neural Networks (GNNs) and standard Transformers, are highly susceptible to overfitting in data-scarce cases. In contrast, fingerprint-based Multi-Layer Perceptrons (MLPs) leverage strong, predefined chemical priors to maintain robust performance, making them the preferred choice when annotated MS/MS data is severely limited. **Interestingly, 1D-CNNs emerged as highly efficient feature extractors in this extreme low-data regime, even outperforming several fingerprint methods.** By treating SMILES strings as sequences, CNNs efficiently capture local substructure patterns like functional groups, without requiring massive datasets to learn global topological rules.

Table S2: Impact of data sparsity on embedder performance, evaluated via Jensen-Shannon Similarity.

Data	**CNN**	**GCN**	**GIN**	**GAT**	**Trans**	**ESPF**	**DL**	**Attn**	**AllDes**	**CBTa**	**GFv2**
25%	0.436	0.412	0.430	0.431	0.425	0.435	0.435	0.427	**0.438**	0.435	0.431
50%	0.458	0.436	0.459	0.457	0.445	0.452	0.457	0.455	0.461	0.460	**0.475**
100%	0.477	0.484	0.494	0.483	0.471	0.475	0.481	0.489	0.491	0.480	**0.516**

Dataset Noise and the Optimization Landscape **Our empirical study highlights that hyperparameter sensitivity in binned spectrum prediction is deeply intertwined with the specific characteristics of the training corpus.** The robust preference for a lower learning rate (10^{-4}) on MassSpecGym can be attributed to the high variability in spectral quality, unpredictable noise, and diverse fragmentation patterns inherent to crowdsourced repositories or synthetic augmentations. In a multi-label regression context with thousands of m/z bins, this heterogeneity creates a rugged, highly non-convex optimization landscape where aggressive gradient steps easily lead to overshooting. In stark contrast, the NPLIB1 dataset consists of curated, high-quality spectra from natural products with predictable and consistent fragmentation rules. This distinct curation process yields a comparatively smoother optimization landscape, permitting models to utilize a higher learning rate (10^{-3}) to achieve faster convergence and leverage dataset regularities more aggressively without destabilizing the training process. *Appendix* D.5 shows the experiment.

The GCN Bottleneck **Our multi-task evaluation identifies GCN as an outlier because it remains competitive in retrieval despite poor performance in absolute simulation.** This gap is theoretically rooted in GCN's aggregation operator ($D^{-1/2}AD^{-1/2}$) performing signal smoothing. While this smoothing builds robust latent representations for discriminative retrieval by capturing global structural similarities, it acts as a low-pass filter that erases the local heterogeneity required for generative simulation. However, because retrieval evaluates relative ranking rather than absolute peak matching, this oversmoothing paradoxically provides a stable envelope for predictions.

Even though the generated spectra lack fine-grained fidelity, these structurally distinctive envelopes are sufficient to reliably discriminate the true compound from decoys in the candidate pool. In contrast, other backbones like GIN (using sum aggregation) and GAT (via adaptive attention) preserve the atom-specific variance necessary to map chemical graphs to sparse, high-dimensional spectral outputs.

C Detailed Methodology

C.1 Datasets Details

To comprehensively evaluate the robustness and generalizability of our framework, we have curated a diverse suite of datasets representing various scenarios encountered in metabolic research . An overview of their scale and benchmark role is provided in Table according to NIPS recommendation S3 [40].

MassSpecGym Dataset MassSpecGym [20] serves as a foundational platform for evaluating model performance in a controlled environment. It comprises 231,000 spectra from 29,000 unique molecules. The dataset was specifically designed for machine learning applications, featuring highly standardized metadata. A key advantage is its rigorous data-splitting procedure based on molecular edit distance, designed to prevent data leakage and robustly assess a model's generalization to novel chemical structures.

GNPS Dataset To test model performance under realistic, large-scale conditions, we incorporate the Global Natural Products Social Molecular Networking (GNPS) library [27]. Containing over 322,000 spectra from more than 16,000 molecules, it is characterized by significant heterogeneity from diverse instruments, laboratories, and experimental protocols.

MassBank Dataset MassBank [28] provides a rich, open-source library of over 62,000 spectra from more than 4,000 small chemical compounds, many of which are high-resolution. Its strict record validation ensures high reliability for metabolomics identification.

NPLIB1 Dataset For more targeted analyses where data quality and balance are critical, we employ the NPLIB1 benchmark dataset [31]. It consists of approximately 8,000 high-quality spectra from 7,000 unique natural products, with a careful curation process that ensures an even distribution across chemical classes.

CASMI Contest Challenges from the Critical Assessment of Small Molecule Identification (CASMI) contest [32] are used to simulate real-world compound identification. The core task

involves correctly identifying the true molecular structure for a given query spectrum from a provided list of candidates.

Dataset	Data Size(Approx.)	Key Characteristic
MassSpecGym	231k spectra 29k molecules	Standardized ML benchmark with clean metadata and generalization-aware data splits. It serves as a modern baseline for evaluating deep learning models across diverse chemical spaces.
GNPS	>322k spectra >16k molecules	Large-scale, crowdsourced library with significant real-world noise and heterogeneity.
MassBank	>62k spectra >4k molecules	The first public repository, offering a rich source of high-resolution spectra. It enforces strict record validation and open data standards to ensure high reliability for metabolomics identification.
NPLIB1 (MIST/Canopus)	∼8k spectra ∼7k molecules	Curated natural products collection with balanced chemical classes and high-quality spectra. It serves as a critical testbed for evaluating model performance on biologically relevant chemical spaces.
CASMI16+22	312/208 cases (16) 500 cases (22)	Real-world compound identification task, requiring ranking of candidate structures.

Table S3: Overview of Datasets Used for Benchmarking

C.2 Retrieval Benchmark Setting

Task Background Direct spectrum-reconstruction metrics quantify fidelity at the bin level, but practical compound identification is fundamentally a ranking problem. The retrieval benchmark therefore evaluates whether a model can place the correct molecular structure near the top of a candidate list for a given experimental spectrum. This setting follows the identification-oriented formulation used in the Critical Assessment of Small Molecule Identification (CASMI) contests [32], where each query is paired with a finite candidate pool and success depends on the rank of the true compound rather than only on reconstruction quality. Figure S1 shows some details.

CASMI 2016 Protocol The CASMI 2016 retrieval benchmark uses Orbitrap spectra for 124 compounds after restricting the evaluation to [M+H]+ adducts and excluding charged species. For each query compound, spectra acquired at normalized collision energies (NCEs) of 20, 35, and 50 are combined into a single query representation. Candidate lists are derived from ChemSpider searches using precursor-mass similarity, yielding an average of 1,251 candidates per query after preprocessing. The candidate-count distribution is long-tailed: most compounds have candidate pools on the order of 10^3, while only a small subset exceeds 4,000 candidates.

CASMI 2022 Protocol The CASMI 2022 retrieval benchmark uses Orbitrap spectra for 228 compounds after restricting the evaluation to [M+H]+ and [M+Na]+ adducts and removing compounds with unsupported elements. This restriction keeps the retrieval task aligned with the ionization conditions most consistently represented across the public training corpora. Candidate molecules are generated from PubChem within a 10 ppm precursor-mass tolerance, following the MassFormer retrieval protocol, with at most 10,000 candidates retained per query. After filtering unsupported elements, removing multimolecular compounds, and deduplicating stereoisomers, the resulting candidate pools contain 4,849 molecules per spectrum on average. In contrast to CASMI 2016, the candidate-count distribution is strongly skewed toward the upper cap, indicating a substantially harder low-prior retrieval regime.

Candidate Preparation and Inference For each query spectrum q, candidate molecules in C_q are matched by precursor m/z and annotated with the query's experimental metadata (e.g., collision energy, instrument type, adduct). After filtering invalid structures, FlexMS predicts a theoretical

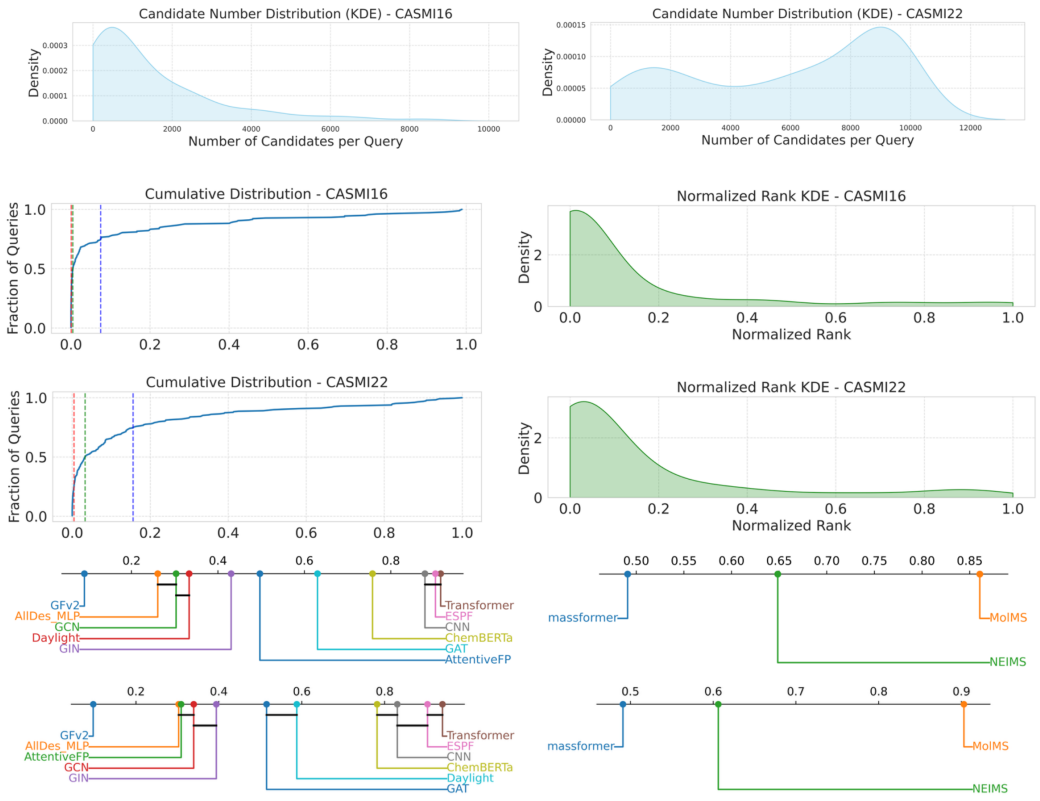

Figure S1: **Upper:** The KDE plot of CASMI compound's candidate number distribution on both CASMI16 and CASMI22 contest. **Mid:** The cumulative distribution and KDE plot of the normalized-rank performance on GFv2-MassFormerMLP combinations. **Bottom:** The Critical Difference Diagram of different embedders and predictors on CASMI16 and CASMI22 contest. We compared the performance on normalized rank, so lower value means better rank.

spectrum for each candidate under these query-specific conditions. The candidates are then ranked by the cosine similarity between their predicted spectra and the observed query spectrum.

C.3 Data Split Evaluation and Statistical Analysis

C.3.1 Molecule Split Setting

Benchmark conclusions depend strongly on the molecular partition protocol because split design controls the degree of chemical overlap between training and evaluation sets. Two split regimes are considered in this benchmark. A *random split* assigns molecules to training, validation, and test partitions stochastically, which may place structurally related compounds across different subsets. A *scaffold split* instead groups molecules by their Bemis–Murcko scaffolds and assigns all molecules sharing the same scaffold to the same partition, thereby enforcing a stricter structural separation [19].

For GNPS and MassBank, both random and scaffold-based partitions are constructed using an 80%/10%/10% train/validation/test ratio. This paired design enables a direct comparison between easier in-distribution evaluation and harder scaffold-aware evaluation under otherwise matched preprocessing and model settings. For MassSpecGym and NPLIB1, the official benchmark splits are retained in order to preserve the intended evaluation semantics of those resources and to maintain comparability with prior work.

This benchmark design intentionally combines official splits and newly constructed scaffold-aware splits across datasets. The objective is not to force every resource into a single partition scheme, but to expose how model rankings and practical conclusions change when the degree of structural separation varies. Retrieval benchmarks on CASMI 2016 and CASMI 2022 are treated separately

from direct prediction splits, because they are organized around query-specific candidate pools and are evaluated as test sets.

C.3.2 Kolmogorov-Smirnov Test and Murcko Metric on Morgan fingerprint

Let each molecule x be represented by a binary Morgan fingerprint $\mathbf{f}(x) \in \{0,1\}^d$ on ECFP radius r and bit-length d. For two molecules x, y, with $a = \|\mathbf{f}(x)\|_1$, $b = \|\mathbf{f}(y)\|_1$, and $c = \mathbf{f}(x)^\top \mathbf{f}(y)$, the Tanimoto similarity is $T(x,y) = \frac{c}{a+b-c} \in [0,1]$.

Given splits Train, Val, and Test, we form similarity pairs. On datasets with large unique molecules, uniform random sampling is taken:

$$Pair_{Train-Train} = \{T(x_i, x_j) : x_i, x_j \in Train,\, i \neq j\} \tag{1}$$

$$Pair_{Train-Val} = \{T(x_i, y_j) : x_i \in Train,\, y_j \in Val\} \tag{2}$$

$$Pair_{Train-Test} = \{T(x_i, x_j) : x_i \in Train,\, y_j \in Test\} \tag{3}$$

To test whether Train-Val distributions and Train-Test distributions differ, the two-sample Kolmogorov–Smirnov (KS) statistic is defined as:

$$D_{n,m} \;=\; \sup_{t \in \mathbb{R}} \big|\widehat{F}_n(t) - \widehat{G}_m(t)\big|, \tag{4}$$

where $\widehat{F}_n$ and $\widehat{G}_m$ are the empirical CDFs. Under the null hypothesis H_0, the corresponding p-value has the Kolmogorov tail with the large-Z asymptotic approximation ($Z = D_{n,m}\sqrt{n_{\text{eff}}}$, where $n_{\text{eff}} = \frac{nm}{n+m}$):

$$p \approx 2e^{-2Z^2} \quad \text{(for large } Z\text{)}. \tag{5}$$

For scaffold overlap using Murcko scaffolds $\sigma(\cdot)$, let $\Sigma(\mathcal{A}) = \{\sigma(x) : x \in \mathcal{A}\}$. The overlaps are defined as:

$$\text{overlap}_{\text{test in train}} = \frac{|\Sigma(\mathcal{A}) \cap \Sigma(\mathcal{B})|}{|\Sigma(\mathcal{B})|}, \quad \text{Jaccard overlap} = \frac{|\Sigma(\mathcal{A}) \cap \Sigma(\mathcal{B})|}{|\Sigma(\mathcal{A}) \cup \Sigma(\mathcal{B})|}. \tag{6}$$

C.3.3 Spectral Entropy Analysis

To assess distributional shifts from the spectral perspective, we employ spectral entropy [39]. For a spectrum with n peaks of intensities $\{I_i\}_{i=1}^n$, the spectral entropy is defined as:

$$H = -\sum_{i=1}^{n} p_i \ln(p_i), \quad \text{where} \quad p_i = \frac{I_i}{\sum_{j=1}^n I_j} \tag{7}$$

We compute spectral entropy for all spectra and apply the two-sample KS test to compare distributions.

C.3.4 Critical Difference Diagram with Wilcoxon-Holm Method

To evaluate the relative performance of k models across N datasets, the average rank $\bar{R}_j$ for model C_j is calculated as:

$$\bar{R}_j = \frac{1}{N} \sum_{i=1}^{N} r_{i,j} \tag{8}$$

Statistical significance is determined using a Friedman test followed by a post-hoc Wilcoxon signed-rank test. To control the Family-Wise Error Rate across m pairwise comparisons, we apply the Holm-Bonferroni correction. For the ordered p-values, a hypothesis H_i is rejected when

$$p_i \leq \frac{\alpha}{m - i + 1} \quad (\alpha = 0.05). \tag{9}$$

C.4 Detailed Featurizer Architectures

Traditional Featurizers We implemented a suite of traditional methods that convert molecular structures into fixed-length vector representations, primarily leveraging RDKit [41] to encode structural and property-based features. We utilize **Morgan fingerprints** (generating bit vectors through

circular hashing of atomic environments with specified radii and bit lengths) [42], **MACCS keys** (166 predefined structural fragments for binary encoding) [43], and **Physicochemical descriptors** (computing molecular properties such as exact molecular weight, logP, hydrogen bond donors and acceptors, rotatable bonds, topological polar surface area, formal charge, and ring counts). We also compute **Atom pair fingerprints** (encoding topological distances between atom pairs), **Extended-reduced Graph (ErG) fingerprints** (representing simplified molecular graphs) [44], **Daylight fingerprints** (using path-based hashing to produce 2048-bit vectors) [45], and **PubChem fingerprints** (881-bit vectors based on predefined substructures) [46]. Finally, a concatenated descriptor featurizer combines Morgan, Morgan counts, MACCS, physicochemical, atom pair, and ErG features into a unified, high-dimensional representation.

Sequence-Based Featurizers Inspired by sequence processing in natural language models, we introduced featurizers that treat SMILES strings as sequential data. **One-hot encoding** transforms SMILES characters into categorical vectors, padding or truncating to a maximum sequence length with a predefined alphabet of 58 symbols (including atoms, bonds, and special tokens). **Explainable Substructure Partition fingerprint (ESPF)** [47] employs byte-pair encoding (BPE) on SMILES strings using a ChEMBL-derived vocabulary, producing tokenized sequences with optional masking and length constraints for transformer compatibility. **ChemBERTa** [8] utilizes the output embeddings generated by a pretrained masked language model applied to these tokenized sequences, obtaining fixed-dimensional molecular representations by post-processing the token embeddings.

Graph-Based Featurizers To exploit the inherent topological structure of molecules, we incorporated graph neural network-compatible featurizers using DGL and RDKit [48, 41]. **Canonical featurizers** employ atom and bond encoders (atomic numbers, chirality, hybridization, bond types) with self-loops and optional virtual nodes for padding to a fixed node count. **AttentiveFP** [25] extends this framework with attention-based atom and bond features, capturing local environments through multi-head mechanisms. **3D featurizers** generate complete graphs from single conformers, embedding 3D coordinates and spatial distances using Alchemy-inspired node and edge features after conformer optimization via UFF force fields. Additionally, simplified molecular graphs convert molecules into PyTorch Geometric data objects encoded with sparse adjacency formats [49].

C.5 MetaData Embedding

In our model architecture, metadata associated with mass spectra includes absolute collision energy (ACE), normalized collision energy (NCE), instrument type, precursor type, ion mode, and precursor $\mathbf{m/z}$.

For ACE and NCE, binary indicators are created to flag missing values. The non-missing values are standardized by subtracting the mean and dividing by the standard deviation with missing values filled as -1 post-normalization. Categorical features are converted to one-hot encoding. If the number of unique categories in the current dataset differs from a predefined count, the one-hot vectors are zeroed out to maintain dimensionality consistency. Precursor $\mathbf{m/z}$ is binned into integer values and treated as a discrete feature. The final metadata embedding is formed by concatenating these processed components.

C.6 Model Architectures

FlexMS adopts a modular design philosophy, decoupling the mass spectrum prediction pipeline into two distinct components: the Molecule Embedding Model (Embedder) and the Spectrum Predictor (Predictor). This modularity allows for the systematic benchmarking of arbitrary combinations of molecular encoders and prediction heads [50].

C.6.1 Embedders

The role of the embedder is to map the raw molecular features into a fixed-dimensional latent vector (h_{mol}), encapsulating essential structural and chemical information. FlexMS implements a robust suite of embedders covering different inductive biases:

- **MLP-based Embedders:** Multi-Layer Perceptrons (MLPs) with residual connections map high-dimensional sparse vectors (e.g., Morgan, MACCS) into dense latent representations.

Daylight, ESPF, and AttentiveFP fingerprints are trained via these networks, alongside an "All-Descriptor" concatenation.

- **Sequence-based Embedders:** We utilize 1D-CNNs to capture local sequence patterns from one-hot encodings [10]. Furthermore, we incorporate Transformer-based encoders (including DeepPurpose [51], ChemBERTa [8], and MoleBERT [24]) to capture long-range chemical dependencies through self-attention mechanisms.
- **Graph Neural Networks (GNNs):** We implement standard message-passing architectures for graph inputs, including GCN [21], GAT [23], and GIN [26]. To evaluate advanced spatial biases, we also include AttentiveFP [25] and GraphormerV2 (GFv2) [36], utilizing sophisticated spatial encodings.

C.6.2 Predictors

The predictor serves as the decoding module, taking the concatenated vector of the molecular embedding (h_{mol}) and the metadata embedding (h_{meta}) as input to generate the final binned mass spectrum. We adapt three distinct predictor architectures representing different modeling paradigms:

- **MassFormer Predictor:** A high-capacity MLP design featuring a bidirectional prediction mechanism [12]. It explicitly models two simultaneous processes: the forward generation of fragments and the reverse neutral loss from the precursor ion. A learnable gating mechanism dynamically weighs the contribution of these two streams, optimizing peak capture across both low and high m/z regions.
- **NEIMS Predictor:** A streamlined, efficiency-focused MLP architecture that prioritizes inference speed [11]. While mathematically simpler than MassFormer, it retains the bidirectional logic required to handle the physical constraints of fragmentation, providing a highly efficient and robust baseline.
- **MolMS Predictor:** An advanced encoder-decoder structure emphasizing deep residual learning [17]. The decoding block utilizes multiple fully connected layers integrated with Residual Blocks. This prevents vanishing gradient issues in deeper networks, theoretically enabling the model to capture highly complex, non-linear relationships between molecular structures and spectral intensity outputs.

C.7 Evaluation Metrics Formulation

C.7.1 Generative Prediction Metrics

- **Cosine Similarity:** Let $\mathbf{y}_{\text{pred}}, \mathbf{y}_{\text{true}} \in \mathbb{R}^B_{\geq 0}$ denote the predicted and observed binned spectra over B mass bins. Cosine similarity is defined as

$$\text{CosSim}(\mathbf{y}_{\text{pred}}, \mathbf{y}_{\text{true}}) = \frac{\mathbf{y}_{\text{pred}}^\top \mathbf{y}_{\text{true}}}{\|\mathbf{y}_{\text{pred}}\|_2 \, \|\mathbf{y}_{\text{true}}\|_2}. \tag{10}$$

- **Jensen–Shannon Similarity:** For distribution-level comparison, the normalized spectra are first defined as

$$\tilde{\mathbf{y}}_{\text{pred}} = \frac{\mathbf{y}_{\text{pred}}}{\sum_{b=1}^B y_{\text{pred},b}}, \qquad \tilde{\mathbf{y}}_{\text{true}} = \frac{\mathbf{y}_{\text{true}}}{\sum_{b=1}^B y_{\text{true},b}}, \tag{11}$$

with mixture distribution

$$\mathbf{m} = \frac{1}{2}\left(\tilde{\mathbf{y}}_{\text{pred}} + \tilde{\mathbf{y}}_{\text{true}}\right). \tag{12}$$

The Jensen–Shannon similarity is then computed as

$$\text{JSSim}(\mathbf{y}_{\text{pred}}, \mathbf{y}_{\text{true}}) = 1 - \frac{1}{2}\left[D_{\text{KL}}(\tilde{\mathbf{y}}_{\text{pred}} \,\|\, \mathbf{m}) + D_{\text{KL}}(\tilde{\mathbf{y}}_{\text{true}} \,\|\, \mathbf{m})\right]. \tag{13}$$

- **Spectral Coverage:** Spectral coverage measures peak-level recall after thresholding with τ. Let

$$\mathbf{1}_\tau(y_b) = \mathbf{1}[y_b > \tau]. \tag{14}$$

Coverage is defined as

$$\text{Coverage}(\mathbf{y}_{\text{pred}}, \mathbf{y}_{\text{true}}) = \frac{\sum_{b=1}^B \mathbf{1}_\tau(y_{\text{pred},b})\, \mathbf{1}_\tau(y_{\text{true},b})}{\sum_{b=1}^B \mathbf{1}_\tau(y_{\text{true},b})}. \tag{15}$$

C.7.2 Retrieval Metrics

- **Raw Rank:** Let $\mathcal{Q}$ denote the set of query spectra. For each query $q \in \mathcal{Q}$, let C_q be the associated candidate set, let $c_q^\star \in C_q$ denote the ground-truth molecule, and let

$$s_q(c) = \mathrm{CosSim}(\hat{\mathbf{y}}_{q,c}, \mathbf{y}_q) \tag{16}$$

be the cosine similarity between the predicted spectrum for candidate c under the metadata of query q and the observed query spectrum $\mathbf{y}_q$. The raw rank of the ground-truth molecule is then defined as

$$\mathrm{Rank}(q) = 1 + \sum_{c \in C_q \setminus \{c_q^\star\}} \mathbf{1}\left[s_q(c) > s_q(c_q^\star)\right]. \tag{17}$$

Lower values indicate better retrieval performance, and $\mathrm{Rank}(q) = 1$ corresponds to a correct top-ranked identification.

- **Normalized Rank:** To account for varying candidate-pool sizes, the normalized rank is defined as

$$\mathrm{NormRank}(q) = \frac{\mathrm{Rank}(q)}{|C_q|}. \tag{18}$$

Aggregate retrieval performance is summarized by the query-averaged rank and normalized rank:

$$\overline{\mathrm{Rank}} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathrm{Rank}(q), \qquad \overline{\mathrm{NormRank}} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathrm{NormRank}(q). \tag{19}$$

- **Top-K Accuracy:** Top-K accuracy measures whether the ground-truth molecule appears within the first K ranked candidates:

$$\text{Top-}K = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathbf{1}[\mathrm{Rank}(q) \le K]. \tag{20}$$

In the experiments, $K \in \{1, 5, 10\}$ is reported.

- **Top-$p\%$ Accuracy:** Top-$p\%$ accuracy measures whether the ground-truth molecule lies within the top $p\%$ of the candidate pool:

$$\text{Top-}p\% = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathbf{1}[\mathrm{NormRank}(q) \le p/100]. \tag{21}$$

Results are reported for $p \in \{1, 5, 10\}$.

C.8 Hyperparameters and Training Details

We fixed specific hyperparameters across the benchmark:

- **CNN:** 3 1D-Conv blocks with channels 32, 64, 96, kernel size 4, and an output dimension of 256.
- **GAT:** 3 DGL GAT blocks with ReLU, hidden dimension 64, and output dimension 64.
- **GCN:** 1 DGL GCN block with ReLU, 3 hidden layers of dimension 64, and output dimension 256.
- **GIN:** 4 DGL GINConv layers with batch normalization and ReLU. Each layer contains a 2-layer MLP (hidden dimension 64). Output dimension is 64.
- **Transformer:** 8 transformer layers, 8 attention heads with hidden size 512, and output embedding dimension of 128.
- **AttentiveFP:** 2 AttentiveFPGNN layers with node feature size 39, edge feature size 11, and output dimension 64.
- **MLPs (ESPF, Daylight, All-descriptors):** Architecture features 3 hidden layers of sizes 1024, 256, and 64 with ReLU activations.

Experimental and Training Configurations To ensure reproducibility and fair evaluation, most critical training details were locked across dataset variants unless specified otherwise:

- **Optimizer and Scheduler:** All models are optimized using the **AdamW** optimizer at a base learning rate of 10^{-4} (unless mentioned) and weight_decay $= 0.0$.
- **Loss Function:** Training explicitly optimizes a modified Cosine Loss defined primarily as $(1 - \text{mean}(\text{cosine\_sim}))$.
- **Epochs and Early Stopping:** We train all regressors for a maximum of **100 epochs**. Early stopping evaluates the validation cosine score, applying a strict early-stopping, reverting to the checkpoint with the highest validation metric.
- **Batch Size:** The standard training batch size enforced for main experimental suite is **512**.
- **Random Seeds:** Datasets are seeded deterministically (seed 42) to isolate architectural impact reliably.
- **GFv2 / Foundation Details:** Models leveraging massive-scale pretrianing (such as GraphormerV2) were **fully finetuned** end-to-end to aggressively adapt their chemical spatial awareness directly toward the MS prediction manifold rather than keeping attention weights frozen.

D Additional Experimental Results and Figures

This section provides additional empirical results and comprehensive visualizations that complement the core findings presented in the main manuscript, including detailed data ablation studies, learning rate sensitivity analyses, and the extended effects of molecular pretraining.

D.1 Comprehensive analysis on predictor and embedder

Figure S2 shows the CDF about predictor and embedder. It illustrates the statistical performance hierarchy of the evaluated embedders and predictors, where positions further to the right signify superior predictive accuracy. In the MSG dataset, graph-based embedders such as GFv2, AttentiveFP, and GIN demonstrate significantly higher efficacy compared to sequence-based models. Furthermore, massformer consistently emerges as the top-ranking predictor across all three datasets—MSG, GNPS, and MassBank—maintaining a statistically significant lead over NEIMS and MolMS in both random and scaffold split scenarios.

D.2 Retrieval benchmark performance on CASMI22

The comparison between CASMI 2016 and CASMI 2022 reveals that performance degradation is primarily driven by the substantially larger candidate pool in the 2022. This harder retrieval regime amplifies the difficulty of placing the correct structure near the top of the ranking: a model that achieves Recall@10 of 60% on CASMI 2016 may drop to under 19% on CASMI 2022, not necessarily because its underlying representations have deteriorated, but because the prior probability of correct identification decreases proportionally with pool size. The candidate-count distributions further confirm this asymmetry—CASMI 2016 exhibits a long-tailed distribution with many small pools, while CASMI 2022 is concentrated near the upper cap, indicating that most queries face a similarly challenging high-prior regime.

Despite the absolute performance drop, the relative ordering of embedders remains broadly consistent across both benchmarks. GFv2 maintains the strongest embedder, and graph-based methods continue to outperform. This pattern suggests that pretrained graph representations capture structural features that are both necessary for spectrum reconstruction and robust to variations in retrieval difficulty. However, the magnitude of GFv2's advantage narrows considerably in CASMI 2022, particularly for top-ranked candidates (Recall@1), indicating that even the best available molecular representations face fundamental limits when discriminating among chemically plausible candidates in extremely large search spaces.

D.3 Cross-Domain Transfer Learning Analysis (Complete)

Figure S3 shows the detailed results about retrieval analysis.

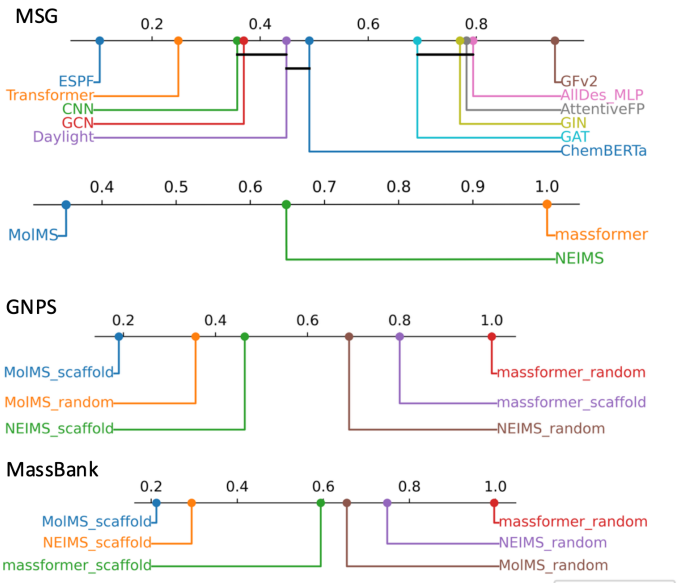

Figure S2: Critical difference diagrams of different embedders and predictors on the MassSpecGym, GNPS and MassBank datasets, obtained by combining results across embedder-predictor pairs, resolutions and datasets, and applying the Wilcoxon-Holm test to detect pairwise significance.

Table S4: Retrieval benchmark performance evaluated on the CASMI 2022 dataset using different predictors. Metrics include Recall@1 (R@1), Recall@10 (R@10), and Normalized Rank (NormR ↓) across candidate pools. Best, second best, and third best results in each column are denoted by **bold**, doubleline, underline, respectively.

	MassFormer			**NEIMS**			**MolMS**		
	R@1	**R@10**	**NormR↓**	**R@1**	**R@10**	**NormR↓**	**R@1**	**R@10**	**NormR↓**
CNN	0.0044	0.0306	0.3332	0.0015	0.0175	0.3576	0.0000	0.0087	0.3906
GCN	0.0044	0.0830	0.2043	0.0131	0.0859	0.2184	0.0087	0.0611	0.2624
GIN	0.0131	0.0786	0.2109	0.0073	0.0830	0.2215	0.0044	0.0699	0.2472
GAT	0.0087	0.0480	0.2473	0.0044	0.0393	0.2378	0.0044	0.0262	0.2510
Trans	0.0087	0.0262	0.3863	0.0029	0.0116	0.3821	0.0000	0.0000	0.3967
ESPF	0.0044	0.0131	0.3844	0.0029	0.0116	0.3689	0.0044	0.0131	0.3557
DL	0.0087	0.0917	0.2046	0.0058	0.0902	0.2128	0.0000	0.0786	0.2348
AttnFP	0.0044	0.0786	0.1918	0.0044	0.0655	0.2225	0.0000	0.0393	0.2852
AllDes	0.0087	0.0917	0.2046	0.0058	0.0902	0.2128	0.0000	0.0786	0.2348
GFv2	0.0218	**0.1878**	**0.1761**	**0.0277**	**0.1572**	**0.1767**	0.0087	**0.1004**	**0.2017**

D.4 Same-Domain Simulation Analysis

To complement the cross-domain transfer learning analysis, we also evaluated within-domain predictive performance. Table S5 demonstrates the spectrum reconstruction fidelity (measured by cosine similarity) when models are trained and tested on the same instrument type (Orbitrap, QTOF) or the same ionization mode ($[M+H]^+$, $[M+Na]^+$).

Consistently, the pretrained graph model GFv2 outperforms almost all other representations across different predictors and sub-domains, confirming that the structural priors captured during pretraining are highly beneficial for capturing within-domain fragmentation rules. Furthermore, prediction accuracy on QTOF spectra generally surpasses Orbitrap spectra across most architectures. In terms of ionization modes, $[M+H]^+$ adducts systematically yield higher reconstruction fidelity than $[M+Na]^+$ adducts. The notable drop in performance for $[M+Na]^+$ (especially with the MolMS predictor) suggests that sodium-adduct fragmentation pathways are either underrepresented in the training data or inherently more difficult to simulate due to their distinct rearrangement mechanisms.

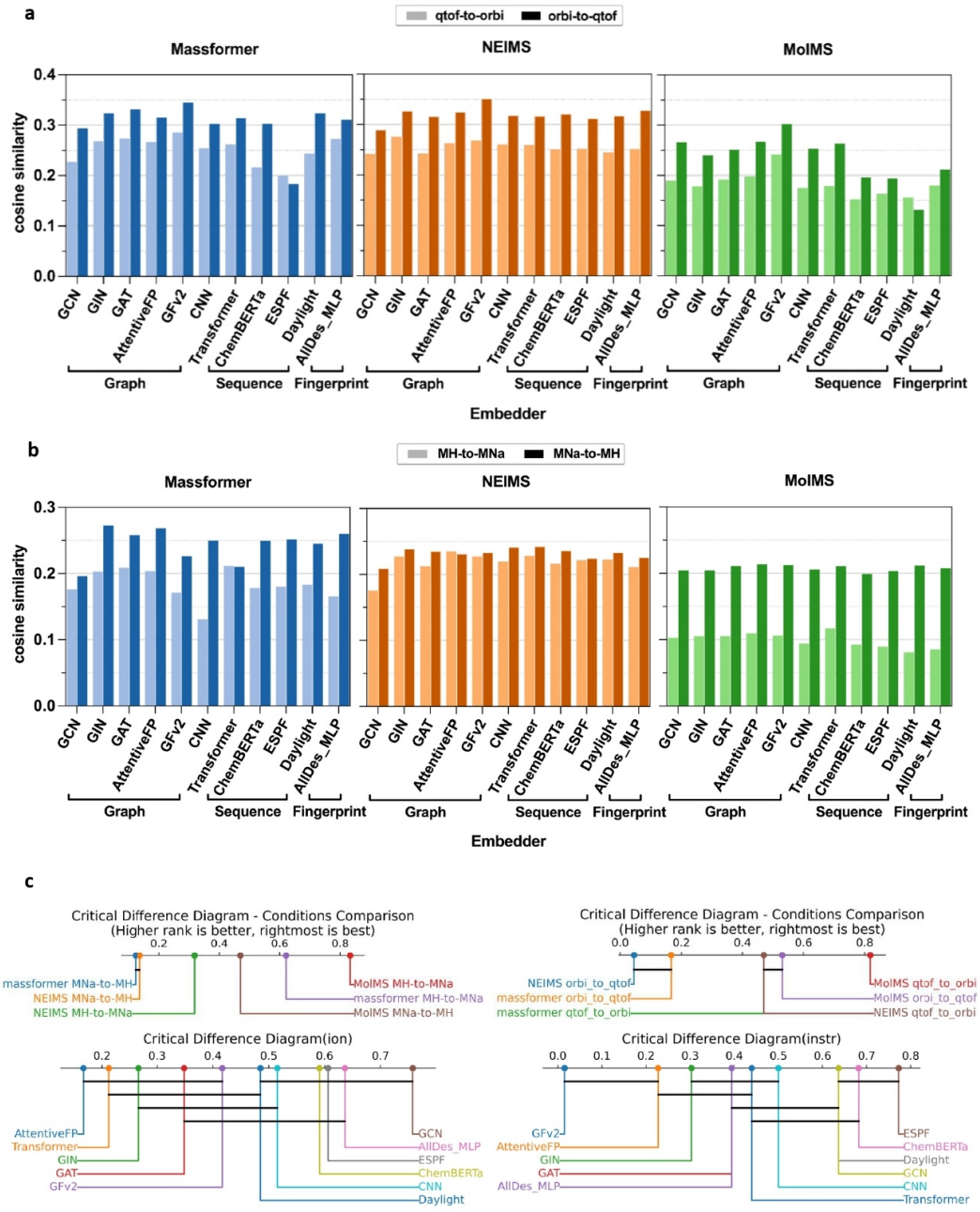

Figure S3: **(a)(b)** Detailed performance comparison of domain transfer, ionization mode transfer and instrument type transfer. **(c)**The critical difference diagram of different transfer case and embedders. Left is ion-transfer case and right is instrument-transfer case.

Table S5: Within-domain transfer learning performance (cosine similarity, number=10). Best results per row are in bold. Transfer directions (Train → Test) are denoted using: Q (QTOF) and O (Orbitrap) for instrument types; H ($[M+H]^+$) and Na ($[M+Na]^+$) for ionization modes.

Task	Pred	Subset	CNN	GCN	GIN	GAT	Trans	ESPF	DL	AttnFP	AllDes	CBT	GFv2
Inst.	Mass	Orbi	0.2862	0.2963	0.2972	0.2898	0.2836	0.2807	0.2915	0.2975	0.3121	0.2975	**0.3392**
		QTOF	0.3233	0.3220	0.3493	0.3357	0.3232	0.3254	0.3583	0.3512	0.3670	0.3223	**0.3883**
	NEIMS	Orbi	0.2797	0.2909	0.3096	0.2885	0.2618	0.2843	0.2953	0.3047	0.3073	0.2880	**0.3377**
		QTOF	0.3071	0.3061	0.3486	0.3264	0.3263	0.3229	0.3390	0.3457	0.3497	0.3348	**0.3690**
	MolMS	Orbi	0.2083	0.2190	0.2270	0.2233	0.2030	0.1949	0.2088	0.2233	0.2092	0.2162	**0.2514**
		QTOF	0.2714	0.2811	0.2992	0.2910	0.2726	0.2606	0.2924	0.3043	0.2999	0.2953	**0.3148**
Ion.	Mass	MH	0.3129	0.3282	0.3271	0.3270	0.3139	0.3011	0.3084	0.3371	0.3345	0.3278	**0.3796**
		Na	0.2434	0.2180	0.2594	0.2366	0.2483	0.2533	0.2288	0.2402	0.2343	0.2408	**0.2604**
	NEIMS	MH	0.2934	0.2974	0.3321	0.3057	0.2997	0.2995	0.3159	0.3213	0.3274	0.3090	**0.3718**
		Na	0.2760	0.2548	0.2607	0.2667	0.2427	0.2651	0.2438	0.2605	0.2614	0.2752	**0.2824**
	MolMS	MH	0.2447	0.2710	0.2791	0.2638	0.2489	0.2307	0.2561	0.2722	0.2681	0.2560	**0.3099**
		Na	0.1229	0.1252	0.1295	0.1305	0.1302	0.1277	**0.1410**	0.1324	0.1246	0.1341	0.1321

D.5 Learning Rate Sensitivity on Mass Prediction

Figure S4 indicates that learning-rate effects are generally smaller than architecture effects, but they remain dataset-dependent. Most points lie close to the diagonal, which means that switching from 10^{-3} to 10^{-4} rarely changes the overall ranking of methods in a dramatic way. However, MassBank and MassSpecGym more often benefit from the smaller learning rate, consistent with their noisier and more heterogeneous optimization landscapes. This observation reinforces the main-text argument that hyperparameter selection in binned spectrum prediction should follow dataset characteristics rather than a single universal default.

D.6 Learning Rate Sensitivity on Molecule Retrieval

Figure S5 extends the learning-rate comparison (10^{-3} vs. 10^{-4}) from direct spectrum reconstruction to downstream retrieval on the CASMI16 dataset. Based on our mass spectrum prediction evaluations, we established two key conclusions: (1) learning rate effects in generative modeling are generally secondary to architectural differences, and (2) optimal learning rates are typically dataset-dependent, with smoother optimization landscapes permitting 10^{-3} and noisier, heterogeneous spectra strictly requiring 10^{-4}.

However, evaluating these models on downstream candidate retrieval reveals a markedly different sensitivity profile. While the performance gap between 10^{-3} and 10^{-4} remains relatively modest for classical fingerprint-based models (e.g., Daylight, ESPF) and 1D-CNNs, deep graph-based architectures—particularly pretrained models like GFv2—exhibit profound optimization instability. Specifically, across Massformer and MolMS predictors, GFv2 experiences a catastrophic collapse in Top-10 accuracy when fine-tuned at a higher learning rate of 10^{-3} (dropping to ~ 0.06), whereas reducing the learning rate to 10^{-4} restores performance to competitive levels ($\sim 0.60 - 0.66$). This stark contrast demonstrates that complex, high-capacity structural encoders are deeply susceptible to optimization divergence or catastrophic forgetting at higher learning rates. Consequently, downstream retrieval tasks heavily amplify structural instabilities that may appear benign under average generative metrics, emphasizing that retrieval-oriented model and hyperparameter selection cannot rely solely on reconstruction robustness.

D.7 Effects of Molecular Pretraining (Complete)

Figure S6 shows that most pretrained-versus-random comparisons lie above the identity line, confirming that self-supervised molecular pretraining usually improves downstream spectrum prediction. The gains are more consistent for cosine similarity and Jensen-Shannon similarity than for spectral coverage, suggesting that pretraining mainly sharpens global spectral fidelity rather than merely recovering more peaks. The improvement is also observed across multiple datasets and predictor heads, which indicates that the benefit is not tied to a single benchmark configuration. Overall,

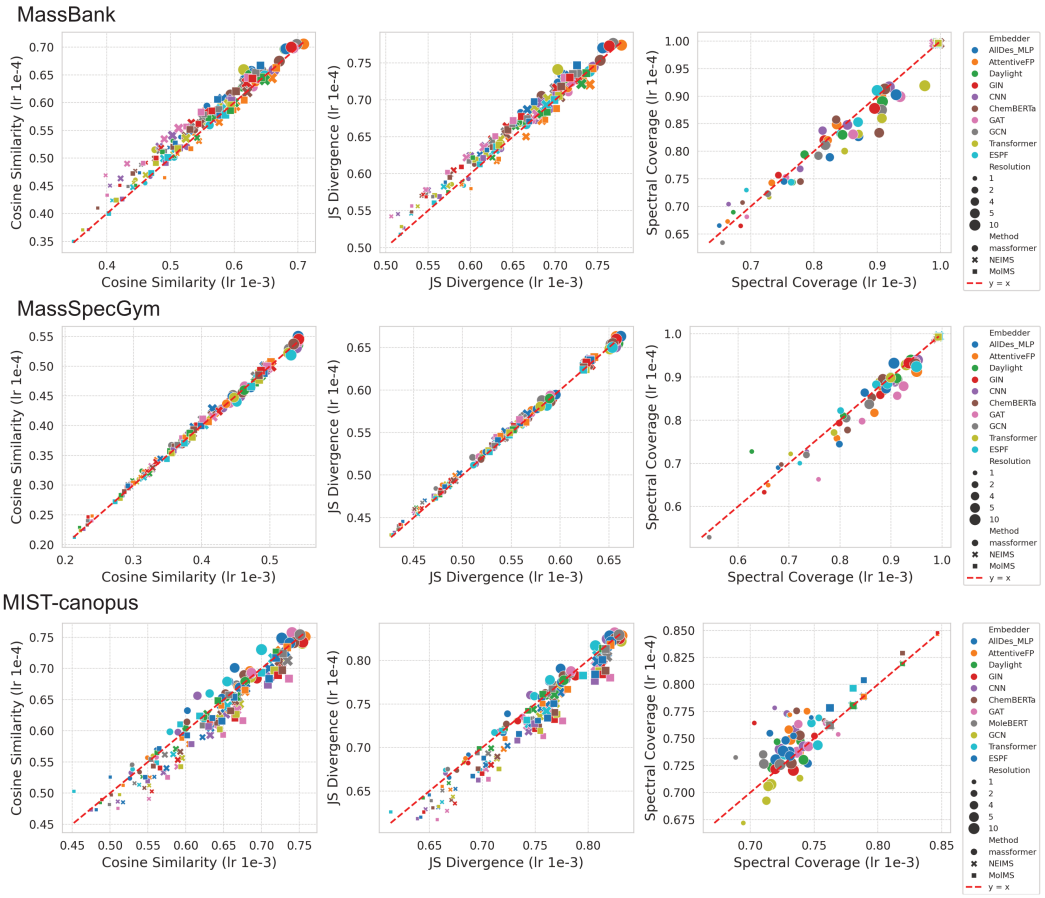

Figure S4: Cosine similarity, Jensen-Shannon similarity, and coverage across different learning rates and conditions. The analysis evaluates all combinations of embedders, predictors and resolutions across 3 datasets. Notably, the GNPS dataset is excluded from this specific ablation study due to its massive scale, which poses significant computational challenges and excessive training time.

these results support the interpretation that pretrained molecular representations transfer chemically meaningful regularities that remain useful in binned MS/MS prediction.

D.8 Resolution Effects and Architectural Robustness in Mass Spectrum Simulation

Our systematic evaluation in Figure S7 of spectral resolution (bin widths ranging from 1 Da to 10 Da) reveals a non-linear performance trajectory in the benchmark. This trend is primarily a mathematical effect of reduced output dimensionality rather than a true accuracy gain, it serves as a valuable structural test.

We identified a distinct architectural nuance in how models respond to resolution changes: graph-based embedders exhibit stable and monotonic improvements as resolution coarsens, indicating robust structural feature extraction. In contrast, sequence-based and pre-trained language representations exhibit more erratic performance trajectories, highlighting differential robustness to variations in spectral granularity.

D.9 Resolution Effects and Architectural Robustness in Retrieval

Consistent with the simulation findings, retrieval benchmarks on CASMI 2016 and CASMI 2022 exhibit a non-linear performance trajectory across resolution settings. Coarser resolutions universally yield improved retrieval accuracy, with the most substantial gains occurring between 1 Da and 2 Da, after which further aggregation provides diminishing returns. Graph-based embedders (GCN, GIN,

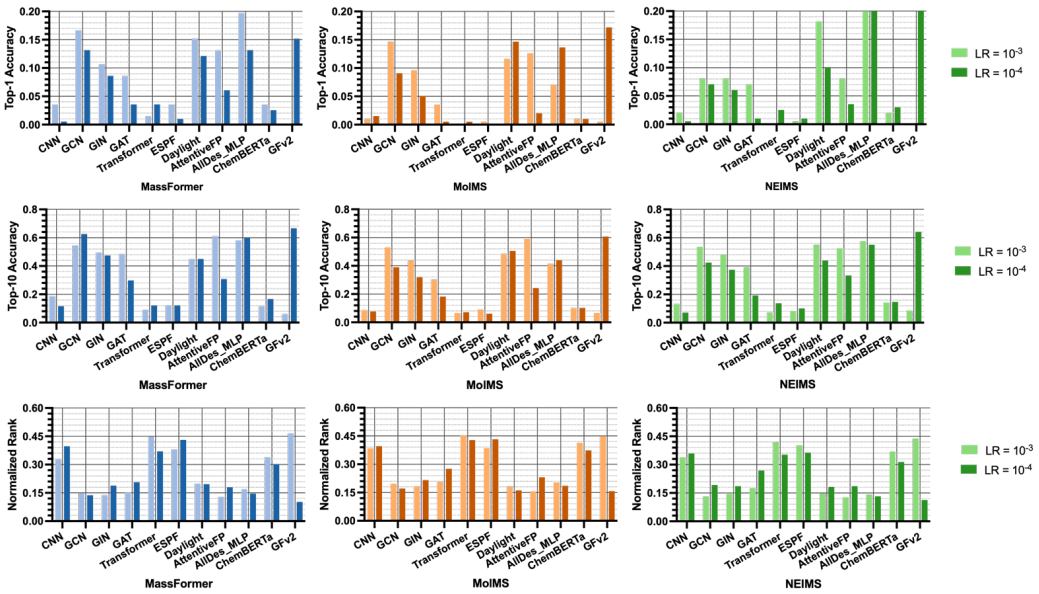

Figure S5: Retrieval performance metrics on the CASMI16 dataset across different learning rates. Various metrics were evaluated for this task to determine optimal convergence behaviors.

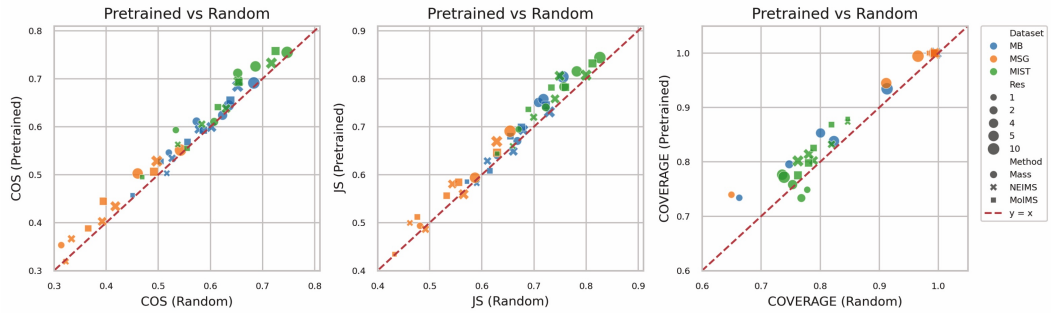

Figure S6: Performance metrics comparison between pretrained MoleBERT and random-initialized MoleBERT. Various metrics and datasets (MassBank, NPLIB1(MIST), and MassSpecGym) are evaluated to demonstrate the structural extrapolation capabilities acquired through self-supervised pretraining.

GAT, GFv2) demonstrate stable, monotonic improvements in normalized rank as resolution coarsens, reflecting robust structural feature extraction invariant to spectral granularity.

Sequence-based and pre-trained representations (Transformer, ESPF, ChemBERTa) display erratic trajectories across both benchmarks, with non-monotonic fluctuations between resolution settings. This architectural divide is particularly evident in Top-5% and Top-10% accuracy trends: graph-based models maintain consistent ranking quality across resolutions, while sequence-based models exhibit sensitivity that can cause degradation at certain coarsening thresholds. The pattern suggests molecular graph representations remain stable under spectral aggregation, whereas SMILES-derived embeddings lose critical substructural information when peaks are binned, leading to inconsistent candidate ranking on real-world identification tasks.

E Future Directions

Several directions remain open for extending the benchmark scope of FlexMS. A first priority is to move beyond the current binned formulation toward evaluation settings that better reflect the structured nature of tandem mass spectra, including peak-list prediction, formula-aware decoding,

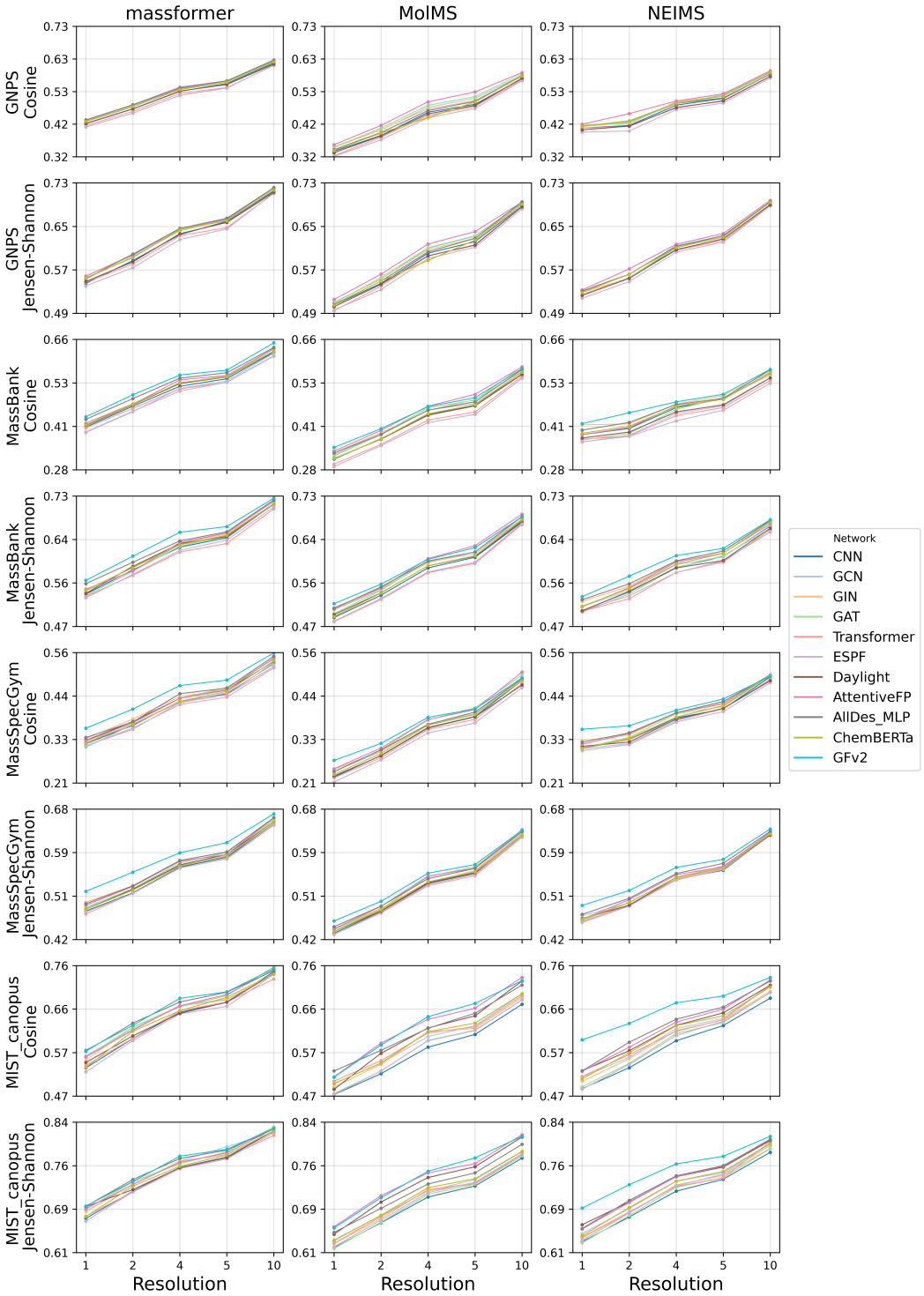

Figure S7: Performance of different embedder-predictor combinations with different resolutions on GNPS dataset (random split), MassBank dataset (random split), MassSpecGym dataset and NPLIB1(MIST) dataset.

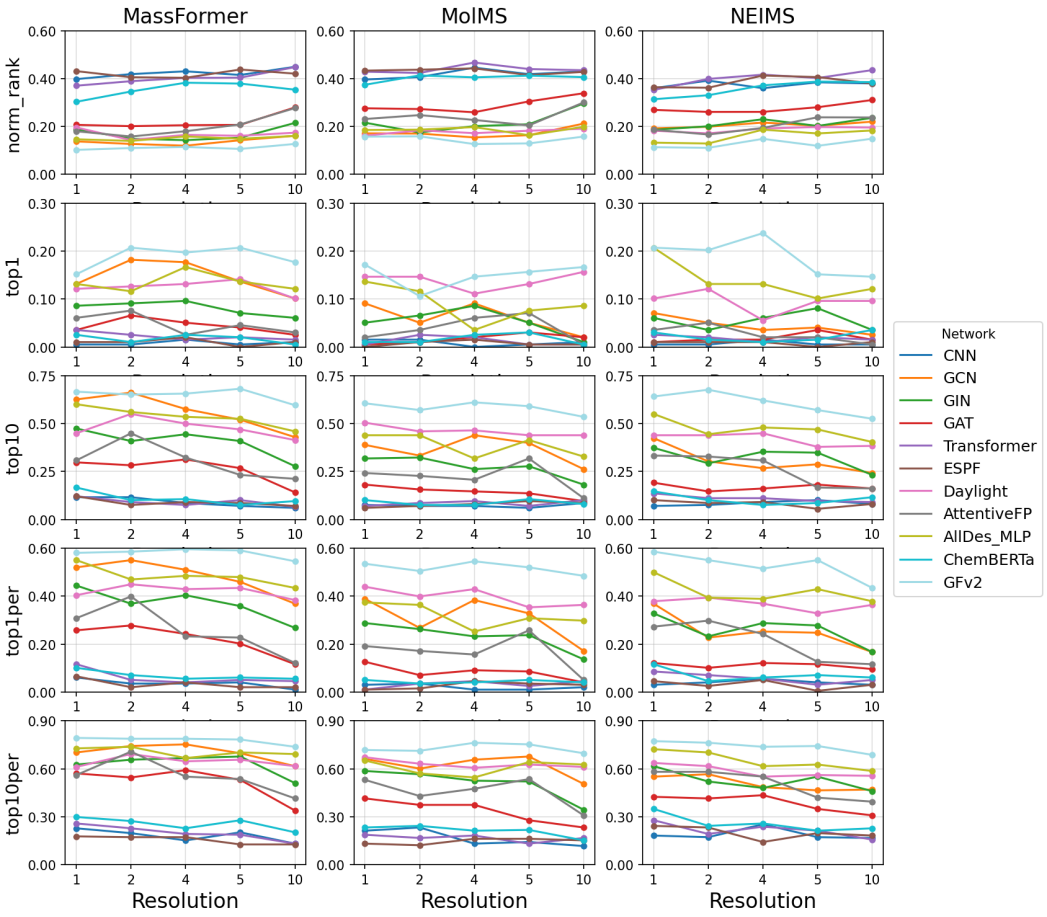

Figure S8: Performance of different embedder-predictor combinations with different resolutions on the CASMI 2016 retrieval benchmark.

and other variable-length output formulations. Such extensions would test whether conclusions that hold for binned regression remain stable under more chemically explicit prediction targets.

A second direction concerns benchmark coverage under domain shift. Although the present study examines transfer across ion types and instrument classes, important sources of variation remain underexplored, including collision-energy differences and other conditions. More systematic protocols for these cases would strengthen the benchmark as a testbed for robust generalization rather than only within-protocol comparison.

A third direction is to broaden efficiency-aware evaluation. Current benchmark tables emphasize predictive quality, but practical adoption also depends on training cost, inference latency, and memory footprint. Standardized budget-aware reporting would make it easier to compare architectures under realistic deployment constraints and to distinguish accuracy gains from gains that remain usable at scale.

Finally, future benchmark iterations would benefit from wider dataset coverage and more diverse downstream tasks. Public MS/MS resources remain uneven in chemical space, adduct composition, acquisition settings, and annotation quality. As additional curated resources become available, the benchmark could be extended to cover broader molecule classes and more realistic identification to do reproducible comparison.

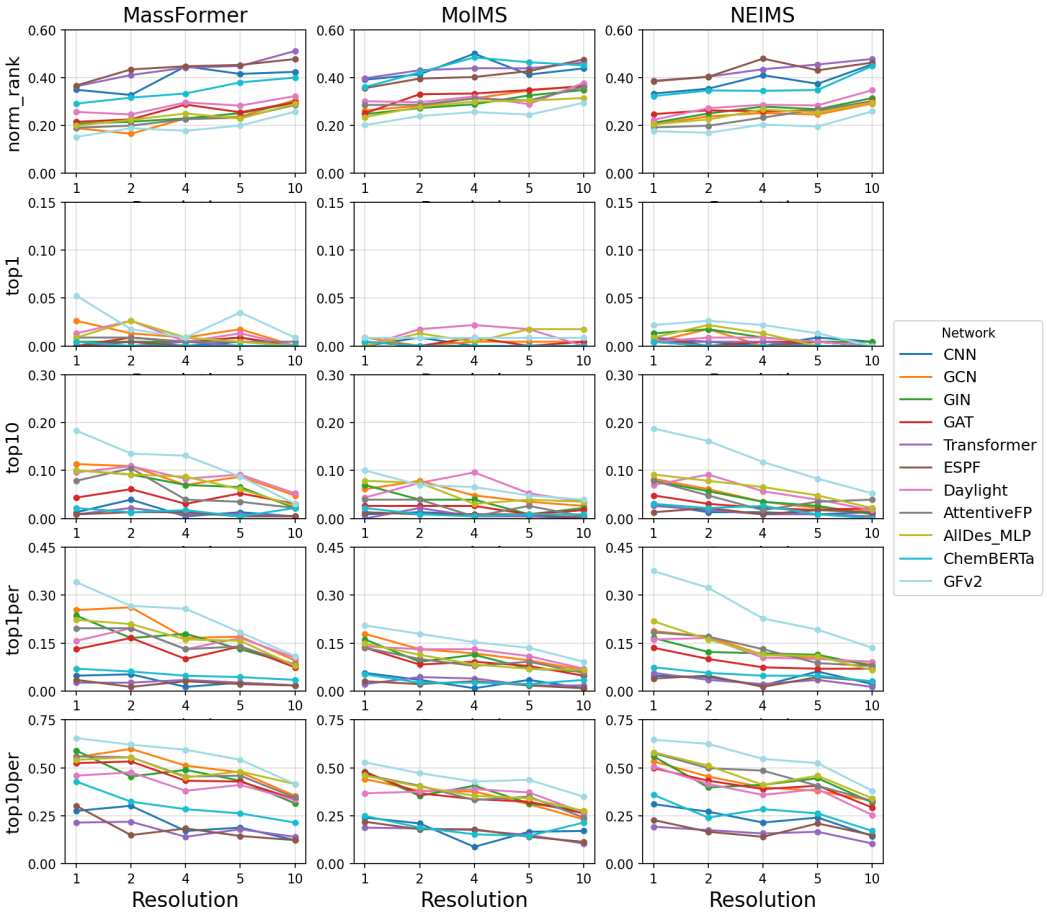

Figure S9: Performance of different embedder-predictor combinations with different resolutions on the CASMI 2022 retrieval benchmark.