FREEDIA: SEARCH-FREE PRETRAINING FOR DIA DE NOVO PEPTIDE SEQUENCING

Anonymous authors
Paper under double-blind review

ABSTRACT

De novo peptide sequencing reconstructs peptide sequences directly from tandem mass spectra without relying on a reference database search. However, sequencing from data-independent acquisition (DIA) spectra is challenging because peaks from multiple co-isolated precursors and background signals are mixed within the same spectrum. Existing DIA de novo methods often use search-dependent supervision based on database-identified peptides, limiting the use of unidentified DIA data for representation learning. We introduce **FREEDIA**, a search-free pretraining framework that learns precursor–fragment attribution from sequence-independent chromatographic supervision. Instead of using rule-based assignments as hard filters, FREEDIA uses them as supervision while retaining the complete observed spectrum, jointly learning denoising and deconvolution and transferring precursor-specific fragment evidence to peptide decoding. Across two datasets, FREEDIA achieves strong performance under both unseen-peptide and sample-holdout evaluations, with state-of-the-art generalization to unseen peptide sequences. Further analyses show that the learned attribution captures precursor-specific and graded spectral evidence beyond binary assignments, while ablation studies confirm the complementary roles of pretraining and reranking. These results highlight search-free spectral pretraining as an effective way to exploit DIA data without requiring peptide identifications during pretraining. Code for reproducing our experiments is provided in the supplementary material.

1 INTRODUCTION

Tandem mass spectrometry (MS/MS) is widely used to identify peptides in complex biological samples, supporting applications such as protein characterization, quantitative proteomics, and biomarker discovery (Aebersold & Mann, 2016). Conventional peptide identification typically relies on database search, matching observed spectra against candidates from a predefined sequence space. In contrast, de novo peptide sequencing directly reconstructs peptide sequences from MS/MS spectra, enabling the identification of previously unseen peptides without a reference database (Tran et al., 2017; Karunratanakul et al., 2019; Eloff et al., 2025). Recent deep learning approaches have substantially improved de novo sequencing, and their performance is closely associated with the extent to which the observed spectrum reflects the fragmentation pattern of the target peptide.

Proteomic MS2 spectra are generated by isolating precursor ions, i.e., ionized peptides selected for fragmentation, from an MS1 spectrum and fragmenting them into sequence-informative product ions. Two common acquisition strategies are data-dependent acquisition (DDA) and data-independent acquisition (DIA), as illustrated in Figure 1. In DDA, only a subset of detected precursors is selected, producing relatively clean MS2 spectra dominated by fragments from a single precursor. However, this selective acquisition can miss low-abundance peptides and vary across runs (Koopmans et al., 2018). DIA instead partitions the precursor mass range into predefined isolation windows and fragments all precursors within each window (Gillet et al., 2012; Tsou et al., 2015). This enables more systematic sampling, including low-abundance peptides missed by DDA, but at the cost of multiplexed MS2 spectra, where target fragments from the target precursor are mixed with co-isolated fragments and background or noise peaks. This mixture obscures the target fragmentation pattern and makes fragment attribution ambiguous. This is particularly challenging for de novo sequencing, which reconstructs peptide sequences directly from spectral evidence without database candidates.

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

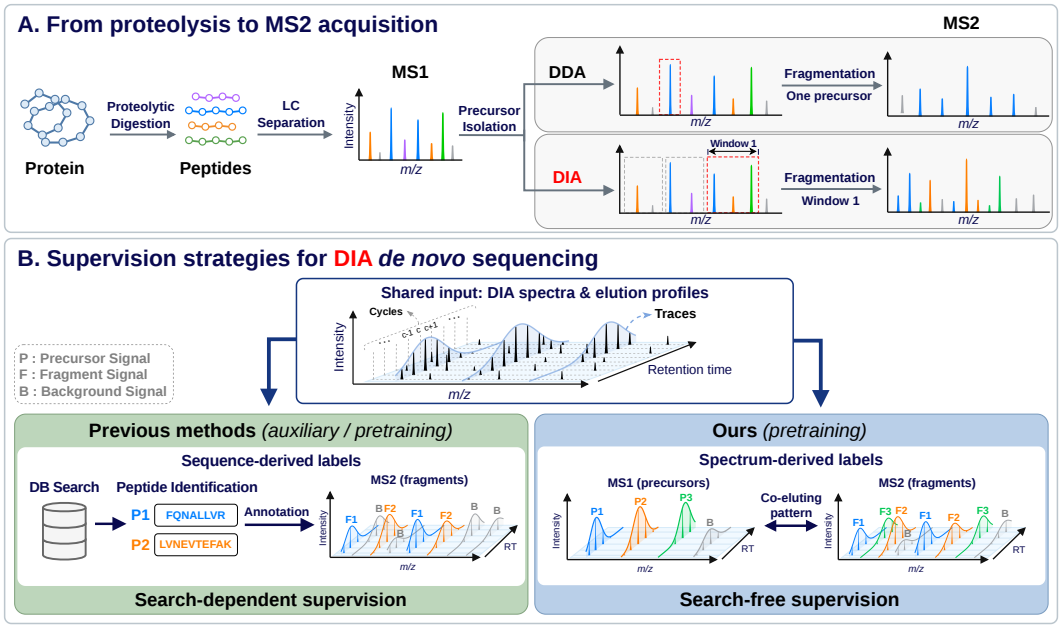


Figure 1: Comparison of DDA and DIA acquisition and pretraining strategies.

Existing DIA de novo sequencing methods address this complexity differently. Earlier approaches primarily rely on peptide sequence supervision (Tran et al., 2019), while recent methods also incorporate peak- or ion-level supervision. Cascadia derives theoretical ion labels from database-identified target peptides, whereas DIANovo pretrains with theoretical ion types from database-identified co-eluting peptides (Ma et al., 2024; Sanders et al., 2025). Although these signals provide richer supervision, they remain restricted to peptides identifiable by database search, preventing unidentified DIA spectra from providing such labels. Thus, although de novo predictions are not restricted to database candidates, their supervision remains limited by database-search identification.

A natural source of supervision without peptide identification is provided by traditional rule-based DIA methods. DIA-Umpire, for example, uses chromatographic evidence, including precursor–fragment co-elution and correlations across neighboring spectra, to infer precursor-specific fragment associations without peptide sequences (Tsou et al., 2015). These relationships provide a precursor-conditioned view of the spectrum, distinguishing fragments associated with the target precursor from co-isolated precursor fragments and background signals. However, rule-based preprocessing applies predefined thresholds to multiple chromatographic features and discards associations that do not satisfy them. This hard filtering can remove borderline but potentially informative evidence, while the retained assignments remain fixed during downstream sequencing (Li et al., 2025).

To address these limitations, we introduce FREEDIA , a search-free pretraining framework for DIA de novo peptide sequencing. Instead of using rule-based assignments to discard unmatched evidence, FREEDIA uses them as supervision while retaining the spectrum. The model learns continuous precursor-conditioned attribution scores distinguishing target fragments, fragments from other precursors, and background signals, jointly capturing denoising and deconvolution. The resulting precursor-conditioned representation guides the decoder toward target fragment evidence, enabling cleaner sequence generation from multiplexed spectra. However, better generation does not guarantee correct candidate selection, so we introduce a spectrum-evidence reranker adapted from RankNovo (Qiu et al., 2025), using DIA-tailored precursor-conditioned evidence. Across diverse DIA datasets, FREEDIA improves sequencing accuracy and unseen-peptide generalization by transferring representations learned from large-scale unidentified DIA data to identified datasets. Our main contributions are: (1) a search-free pretraining task that learns precursor–fragment attribution without database-identified peptides; (2) a precursor-conditioned representation that jointly captures denoising and deconvolution and directly guides peptide decoding; and (3) a candidate reranker that leverages spectrum-evidence to improve candidate selection.

2 RELATED WORK

Deep learning has been increasingly applied to DIA de novo peptide sequencing. DeepNovo-DIA models multiplexed DIA spectra using precursor and fragment information across m/z, retention time, and intensity (Tran et al., 2019). Transformer-DIA adopts a Transformer encoder–decoder architecture (Ebrahimi & Guo, 2023), while DiffNovo incorporates diffusion modeling for peptide generation (Ebrahimi et al., 2025). Despite these architectural advances, these methods rely primarily on peptide sequence supervision from annotated DIA spectra, without explicitly learning precursor–fragment relationships before decoding.

Recent methods introduce additional spectral supervision. Cascadia predicts whether observed peaks match theoretical b/y ions of an identified target peptide (Sanders et al., 2025), while DIANovo uses BERT-style pretraining to predict theoretical ion types from database-identified co-eluting peptides (Ma et al., 2024). Although these objectives improve spectral representation learning, their labels depend on database-identified peptides and therefore cannot directly exploit unidentified DIA.

FREEDIA instead derives supervision from precursor–fragment relationships obtained through rule-based co-elution analysis, enabling search-free pretraining on unidentified DIA data. The learned precursor-conditioned attribution is transferred to peptide decoding to exploit precursor-specific spectral evidence.

3 METHODOLOGY

3.1 PROBLEM FORMULATION

A DIA run consists of repeated cycles, each with a full MS1 survey spectrum followed by MS2 spectra over predefined isolation windows $W_r = [\mu_r^{\text{lo}}, \mu_r^{\text{hi}}]$, $r = 1, \ldots, R$. For cycle c, let S_c^1 denote the MS1 spectrum and $S_{c,r}^2$ the MS2 spectrum for window W_r, obtained by fragmenting all precursors within the window. Each spectrum is a set of peaks (μ, I) denoting m/z and intensity. We define the window-restricted MS1 spectrum as $S_{c,r}^1 = \{(\mu, I) \in S_c^1 : \mu \in W_r\}$.

Over C neighboring cycles, we construct a chromatographic trace group $G = (\mathcal{T}^1, \mathcal{T}^2)$, with $N_1 = |\mathcal{T}^1|$ MS1 traces and $N_2 = |\mathcal{T}^2|$ MS2 traces. The traces in G are then tokenized into $E \in \mathbb{R}^{(N_1+N_2)\times d}$, as described in Section 3.2.

The encoder produces $H = f_\theta(E) \in \mathbb{R}^{(N_1+N_2)\times d}$. For each precursor trace k, pretraining yields attribution scores $\hat{\mathbf{s}}_k \in \mathbb{R}^{N_2}$ over the fragment traces, as detailed in Section 3.3.

For de novo sequencing, let μ_k and z_k denote the precursor m/z and charge. Given the pretrained encoder representations H and (μ_k, z_k), the decoder generates a variable-length peptide $\hat{P}_k = (a_1, \ldots, a_L)$, as detailed in Section 3.4.

3.2 CHROMATOGRAPHIC TRACES AS TOKENS

Precursor and fragment ions from the same peptide tend to vary together in intensity over neighboring cycles. DIA-Umpire uses this shared temporal behavior to associate precursor and fragment signals (Tsou et al., 2015). Following this principle, we represent spectral signals as chromatographic traces rather than isolated peaks, allowing the model to directly capture precursor–fragment co-elution over time.

Specifically, we link signals with consistent m/z across a window of neighboring cycles, with implementation details provided in Appendix B. Each trace i is represented as

$$\tau_i = (\mu_i, \mathbf{x}_i, \rho_i, \ell_i), \qquad \mathbf{x}_i = (I_{i,1}, \ldots, I_{i,C}) \in \mathbb{R}_{\geq 0}^C, \qquad \rho_i = t_{c^*}, \;\; c^* = \arg\max_c I_{i,c}. \tag{1}$$

where μ_i is the trace m/z and $\ell_i \in \{1, 2\}$ denotes the MS level. The elution profile $\mathbf{x}_i$ contains the trace intensity $I_{i,c}$ across cycles, and ρ_i is the apex retention time, defined as the retention time t_{c^*} of the cycle with maximum intensity.

For each isolation window, the resulting traces form

$$G = (\mathcal{T}^1, \mathcal{T}^2), \qquad \mathcal{T}^1 = \{\tau_i^1\}_{i=1}^{N_1}, \qquad \mathcal{T}^2 = \{\tau_i^2\}_{i=1}^{N_2}. \tag{2}$$

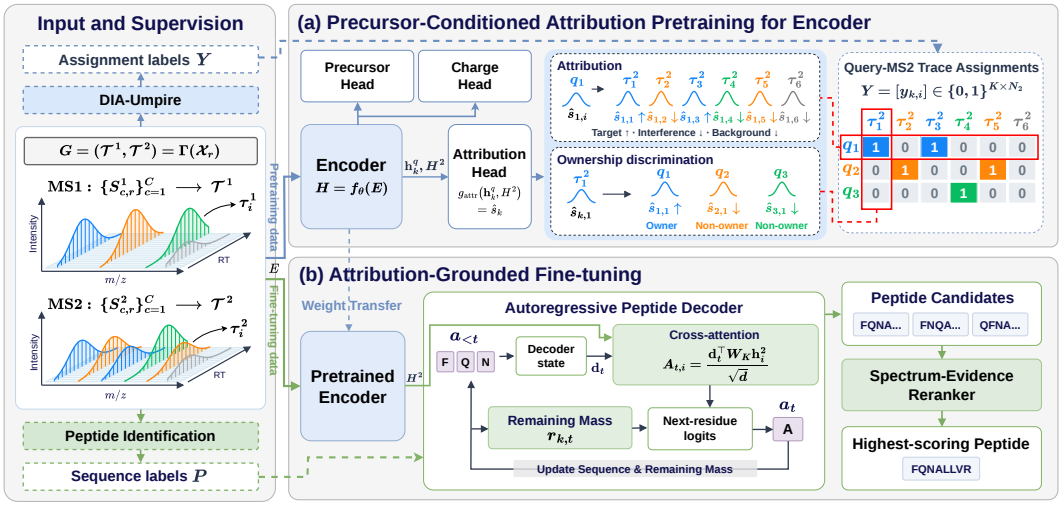


Figure 2: **Overview of FREEDIA .** FREEDIA learns precursor–fragment attribution from search-free chromatographic supervision and transfers it to de novo sequencing.

Each trace τ_i is embedded as $\mathbf{e}_i = g_{\text{emb}}(\tau_i) \in \mathbb{R}^d$, where g_{emb} combines sinusoidal m/z encoding with learned projections of chromatographic and trace-level features (Yilmaz et al., 2024). The token matrix is $E = [\mathbf{e}_1, \ldots, \mathbf{e}_{N_1+N_2}]^\top \in \mathbb{R}^{(N_1+N_2)\times d}$. Details are in Appendix A.

3.3 PRECURSOR-CONDITIONED ATTRIBUTION PRETRAINING FOR ENCODER

Given the trace tokens E constructed in Section 3.2, the encoder jointly contextualizes all MS1 and MS2 traces through full self-attention:

$$H = \begin{bmatrix} H^1 \\ H^2 \end{bmatrix} = f_\theta(E) \in \mathbb{R}^{(N_1+N_2)\times d}, \qquad H^1 \in \mathbb{R}^{N_1 \times d}, \quad H^2 \in \mathbb{R}^{N_2 \times d}. \tag{3}$$

We pretrain the encoder with four objectives: precursor-conditioned attribution, ownership discrimination, precursor detection, and charge prediction. The primary objective is precursor-conditioned attribution, which estimates how strongly each MS2 trace is associated with a given target precursor.

Precursor-conditioned attribution. To construct supervision, we first run DIA-Umpire to detect precursors and their associated fragment traces from chromatographic evidence (Tsou et al., 2015). The detected precursors are matched to the corresponding MS1 traces $\mathcal{T}^1$, which are used as precursor queries, $\mathcal{Q} = \{q_k\}_{k=1}^{K} \subseteq \mathcal{T}^1$, where $K = |\mathcal{Q}|$.

The DIA-Umpire precursor–fragment assignments provide binary attribution targets:

$$y_{k,i} = \mathcal{R}\big(\text{corr}(\mathbf{x}^q_k, \mathbf{x}^2_i), |\rho^q_k - \rho^2_i|, \text{ov}(q_k, \tau^2_i)\big) \in \{0,1\}, \qquad Y = [y_{k,i}] \in \{0,1\}^{K \times N_2}. \tag{4}$$

where $\mathcal{R}$ denotes the DIA-Umpire assignment rule and ov measures overlap of the elution ranges. Rather than retaining separate precursor-specific pseudo-MS2 spectra, Y re-indexes all precursor–fragment assignments over the shared set of MS2 traces.

Relative to query q_k, an MS2 trace is TARGET if assigned to q_k, INTERFERENCE if assigned to another co-isolated precursor, and BACKGROUND otherwise. This query-conditioned view captures denoising and deconvolution within a single attribution task.

To predict these relationships, let $\mathbf{h}^q_k \in \mathbb{R}^d$ denote the encoded representation of query q_k in H^1, and $\mathbf{h}^2_i \in \mathbb{R}^d$ that of MS2 trace i in H^2. The attribution head scores each query–MS2 trace pair as

$$\hat{s}_{k,i} = \frac{\langle W_q \mathbf{h}^q_k, W_v \mathbf{h}^2_i \rangle}{\sqrt{d_s}} + \phi\big(U_q \mathbf{h}^q_k \odot U_v \mathbf{h}^2_i\big) + b. \tag{5}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product, W_q, W_v, U_q, and U_v are learned projections, d_s is the projection dimension, ϕ is a two-layer MLP, and $\odot$ denotes element-wise multiplication. A larger $\hat{s}_{k,i}$

indicates stronger evidence that trace i belongs to query q_k. We optimize these scores with binary cross-entropy:

$$\mathcal{L}_{\text{attr}} = \frac{1}{KN_2} \sum_{k=1}^{K} \sum_{i=1}^{N_2} \ell_{\text{BCE}}(\hat{s}_{k,i}, y_{k,i}) \,. \tag{6}$$

Ownership discrimination. Although $\mathcal{L}_{\text{attr}}$ treats each query–MS2 trace pair independently, it does not compare precursor ownership across queries. For each MS2 trace, let $O_i = \{k : y_{k,i} = 1\}$ denote its assigned queries and $\mathcal{C} = \{i : 1 \le |O_i| < K\}$ the traces used for ownership discrimination. The ownership loss is

$$\mathcal{L}_{\text{own}} = \frac{1}{|\mathcal{C}|} \sum_{i \in \mathcal{C}} \left[\log\left(1 + \sum_{k \notin O_i} e^{\hat{s}_{k,i}}\right) + \log\left(1 + \sum_{k \in O_i} e^{-\hat{s}_{k,i}}\right) \right]. \tag{7}$$

The first term suppresses non-owner scores and the second promotes owner scores, contrasting TARGET and INTERFERENCE. A trace may belong to multiple precursors, so we use multi-label learning.

Precursor and charge prediction. We further apply two auxiliary objectives to the MS1 representations. The precursor head predicts whether each MS1 trace corresponds to a precursor detected by DIA-Umpire, while the charge head predicts the charge of precursor-positive traces:

$$\mathcal{L}_{\text{prec}} = \frac{1}{N_1} \sum_{i=1}^{N_1} \text{BCE}\big(g_{\text{prec}}(\mathbf{h}_i^1), y_i^{\text{prec}}\big) \,, \qquad \mathcal{L}_{\text{chg}} = \frac{1}{|\mathcal{P}|} \sum_{i \in \mathcal{P}} \text{CE}\big(g_{\text{chg}}(\mathbf{h}_i^1), z_i\big) \,. \tag{8}$$

Here, $y_i^{\text{prec}} \in \{0, 1\}$ indicates whether MS1 trace i matches a DIA-Umpire-detected precursor, $\mathcal{P} = \{i : y_i^{\text{prec}} = 1\}$ denotes precursor-positive traces, and z_i is the charge label.

Pretraining objective. The overall pretraining objective combines the four losses:

$$\mathcal{L}_{\text{pre}} = \mathcal{L}_{\text{attr}} + \lambda_{\text{own}}\mathcal{L}_{\text{own}} + \lambda_{\text{prec}}\mathcal{L}_{\text{prec}} + \lambda_{\text{chg}}\mathcal{L}_{\text{chg}}. \tag{9}$$

DIA-Umpire assignments are used only as supervision and never filter the spectral input. Since none of these objectives requires peptide identification, the encoder can be pretrained directly on unidentified DIA data while retaining the full spectral evidence.

3.4 ATTRIBUTION-GROUNDED FINE-TUNING

After attribution pretraining, we jointly fine-tune the encoder and an autoregressive peptide decoder for sequence generation. Given the encoded traces $H \in \mathbb{R}^{(N_1+N_2)\times d}$ and precursor mass $M_k \in \mathbb{R}$, computed from the observed μ_k and z_k of precursor k, the decoder generates the peptide sequence.

Peptide decoding. For a peptide $P_k = (a_1, \ldots, a_L) \in \mathcal{V}^L$, the decoder p_ϕ predicts each residue autoregressively. We guide decoding with cross-attention over the encoded MS2 spectral representations and the remaining precursor mass:

$$A_{t,i} = \frac{\mathbf{d}_t^\top W_K \mathbf{h}_i^2}{\sqrt{d}}, \qquad r_{k,t} = M_k - \sum_{u<t} m(a_u). \tag{10}$$

Here, $\mathbf{d}_t \in \mathbb{R}^d$ is the decoder state at step t, W_K is a learned projection, and $m(a)$ denotes the residue mass. The cross-attention term allows the decoder to attend to MS2 spectral representations relevant to the current decoding step, while $r_{k,t}$ provides the remaining mass for subsequent residues.

Thus, residue prediction is conditioned on the sequence prefix, spectral representation, and remaining mass:

$$p_\phi(P \mid H, M_k) = \prod_{t=1}^{L} p_\phi(a_t \mid a_{<t}, H, r_{k,t}), \qquad \mathcal{L}_{\text{pep}} = -\frac{1}{L} \log p_\phi(P \mid H, M_k). \tag{11}$$

Mass matching. Although the remaining mass guides each decoding step, the complete generated sequence should also match the precursor mass. Let $p_{k,t}(a) = p_\phi(a_t = a \mid a_{<t}, H, r_{k,t})$ denote the probability of residue $a \in \mathcal{V}$ at step t. We compute the implied peptide mass as

$$\hat{M}_k = \sum_{t=1}^{L} \sum_{a \in \mathcal{V}} p_{k,t}(a)\, m(a), \qquad \mathcal{L}_{\text{mass}} = \left(\frac{\hat{M}_k - M_k}{M_k} \right)^2. \tag{12}$$

This promotes mass-consistent decoding, with precursor mass enforced at inference.

Fine-tuning objective. The overall fine-tuning objective is

$$\mathcal{L}_{\text{ft}} = \mathcal{L}_{\text{pep}} + \lambda_{\text{mass}} \mathcal{L}_{\text{mass}}, \tag{13}$$

where λ_{mass} controls the mass-matching contribution. The encoder and decoder are jointly optimized, allowing the pretrained precursor-conditioned representation to adapt to peptide sequencing.

3.5 Spectrum-Evidence Candidate Reranking

Beam search improves candidate coverage, but decoder scores may not select the candidate best supported by the spectrum (Qiu et al., 2025). We therefore introduce a spectrum-evidence reranker to improve candidate selection using precursor-specific fragment evidence.

For query q_k, beam search produces a candidate set $\hat{\mathcal{P}}_k$. For each candidate $\hat{P} \in \hat{\mathcal{P}}_k$, we match theoretical fragments to observed MS2 traces by m/z and extract spectral evidence, including intensity, mass error, and co-elution. The reranker $r_\xi(\hat{P})$ aggregates this evidence into a scalar score. Given the ground-truth peptide P_k, we construct a target distribution from candidate sequence similarity:

$$\pi_k(\hat{P}) = \text{softmax}_{\hat{P} \in \hat{\mathcal{P}}_k} \left(\frac{\text{sim}(\hat{P}, P_k)}{\tau} \right), \qquad \mathcal{L}_{\text{rank}} = - \sum_{\hat{P} \in \hat{\mathcal{P}}_k} \pi_k(\hat{P}) \log q_{\xi,k}(\hat{P}), \tag{14}$$

where $q_{\xi,k}(\hat{P}) = \text{softmax}_{\hat{P} \in \hat{\mathcal{P}}_k}\big(r_\xi(\hat{P})\big)$ is the predicted distribution and τ is the target temperature.

3.6 Inference

We employ bidirectional decoding with separately trained N-to-C and C-to-N decoders, following prior de novo sequencing approaches (Tran et al., 2017). Candidates from both directions are pooled and rescored with the spectrum-evidence reranker, and the highest-scoring candidate is selected. Further inference details are provided in Appendix C.

4 Experiments

4.1 Experimental Setup

Datasets. For attribution pretraining, we use two CPTAC DIA studies: clear cell renal cell carcinoma (ccRCC; 194 runs, 110 patients) and pancreatic ductal adenocarcinoma (PDA; 178 runs, 89 patients), totaling 372 runs, without peptide identification (Thangudu et al., 2024). For downstream evaluation, we use the ovarian cyst (OC) and urinary tract infection (UTI) cohorts from MassIVE MSV000082368 (Tran et al., 2019; Ebrahimi & Guo, 2023). The pretraining and evaluation data come from distinct studies, instruments, and sample types, enabling cross-cohort evaluation. Ground-truth peptide-spectrum matches (PSMs) are obtained by MSFragger (Kong et al., 2017).

Splits. We evaluate two complementary settings. The *sample-holdout* holds out complete runs to test generalization to new biological samples, while allowing peptide overlap with training. The *unseen-peptide* split partitions peptide sequences into train, validation, and test sets at an approximately 8:1:1 ratio, with no sequence overlap across splits, providing a stricter test of sequence generalization.

Metrics. Following standard de novo sequencing evaluation and NovoBench (Zhou et al., 2024), we report amino-acid precision, amino-acid recall, and peptide recall. These measure the fractions of predicted residues that are correct, ground-truth residues recovered, and ground-truth peptides exactly matched, respectively. Results are averaged over three random seeds, with checkpoints selected on validation performance.

Table 1: **De novo sequencing performance on OC and UTI.** Mean results are reported with standard deviations in parentheses. Best and second-best results are **bolded** and underlined, respectively.

Dataset	Model	Unseen Peptide			Sample Holdout		
		AA Prec.	AA Rec.	Pep. Rec.	AA Prec.	AA Rec.	Pep. Rec.
OC	**FREEDIA(Ours)**	**0.6906** (0.0046)	**0.6884** (0.0046)	**0.4355** (0.0084)	**0.9034** (0.0031)	**0.9036** (0.0029)	0.8275 (0.0031)
	Cascadia	0.1771 (0.0103)	0.1775 (0.0103)	0.0760 (0.0077)	0.3412 (0.0300)	0.3407 (0.0302)	0.2847 (0.0300)
	DIANovo	0.6073 (0.0099)	0.6034 (0.0133)	0.3746 (0.0113)	0.6327 (0.0050)	0.6336 (0.0062)	0.4715 (0.0050)
	Transformer-DIA	0.2060 (0.0079)	0.1976 (0.0099)	0.0519 (0.0091)	0.8766 (0.0123)	0.8741 (0.0108)	**0.8478** (0.0123)
	DeepNovo-DIA	0.3168 (0.0085)	0.3145 (0.0077)	0.1822 (0.0066)	0.6664 (0.0047)	0.6632 (0.0043)	0.6349 (0.0047)
	DiffNovo	0.0498 (0.0008)	0.0495 (0.0022)	0.0006 (0.0006)	0.8049 (0.0164)	0.8074 (0.0165)	0.7950 (0.0164)
UTI	**FREEDIA(Ours)**	**0.6980** (0.0045)	**0.6987** (0.0037)	**0.4368** (0.0076)	**0.8790** (0.0021)	**0.8788** (0.0022)	**0.7913** (0.0037)
	Cascadia	0.1726 (0.0053)	0.1737 (0.0062)	0.0737 (0.0101)	0.2835 (0.0061)	0.2839 (0.0057)	0.2246 (0.0057)
	DIANovo	0.5952 (0.0102)	0.5976 (0.0085)	0.3774 (0.0145)	0.6156 (0.0065)	0.6157 (0.0064)	0.4453 (0.0093)
	Transformer-DIA	0.2385 (0.0088)	0.2366 (0.0135)	0.0585 (0.0035)	0.8361 (0.0041)	0.8284 (0.0035)	0.7910 (0.0069)
	DeepNovo-DIA	0.3518 (0.0036)	0.3469 (0.0046)	0.2028 (0.0021)	0.5911 (0.0031)	0.5882 (0.0021)	0.5522 (0.0035)
	DiffNovo	0.0444 (0.0024)	0.0438 (0.0026)	0.0000 (0.0000)	0.7305 (0.0141)	0.7327 (0.0118)	0.7222 (0.0149)

4.2 MAIN RESULTS

Table 1 compares FREEDIA with existing DIA de novo sequencing methods under unseen-peptide and sample-holdout settings. FREEDIA achieves the best performance across all metrics in the unseen-peptide setting and remains strong under sample holdout, demonstrating robust generalization beyond peptides observed during training. Transformer-DIA and DiffNovo show much larger drops when peptide overlap is removed, suggesting greater reliance on recurring sequences, whereas DIANovo degrades less, highlighting the potential benefit of pretraining on diverse data. Cascadia performs relatively lower, which may partly reflect its different formulation without explicit precursor-feature conditioning (Sanders et al., 2025). Together, these results underscore the importance of learning transferable spectral representations for generalizable DIA de novo sequencing.

4.3 ABLATION STUDIES

To isolate the contribution of each component, we ablate attribution pretraining and candidate reranking in Table 2. Consistent with our architectural design, removing pretraining causes a larger degradation in AA precision and recall, whereas removing the reranker more strongly affects peptide recall. Removing both leads to the largest overall degradation, confirming that the two components provide complementary benefits. These results indicate that pretraining primarily improves residue-level spectral interpretation and generation, while reranking is particularly important for selecting the correct complete peptide among generated candidates.

Table 2: Ablation study on OC.

Model	AA Prec.	AA Rec.	Pep Rec.
FREEDIA	**0.6906** **(0.0046)**	**0.6884** **(0.0046)**	**0.4355** **(0.0084)**
w/o Pretrain	0.5709 (0.0109)	0.5700 (0.0104)	0.3628 (0.0072)
w/o Reranker	0.6468 (0.0053)	0.6457 (0.0055)	0.3481 (0.0104)
w/o Both	0.4977 (0.0049)	0.5035 (0.0083)	0.2933 (0.0100)

Figure 4: **Analysis of precursor–fragment attribution.** (a) Precursor-conditioned attribution scores. (b) Alignment between attribution scores and precursor–fragment co-elution correlation. (c) Recovery of ground-truth peptide fragments from unassigned traces by attribution score or co-elution correlation.

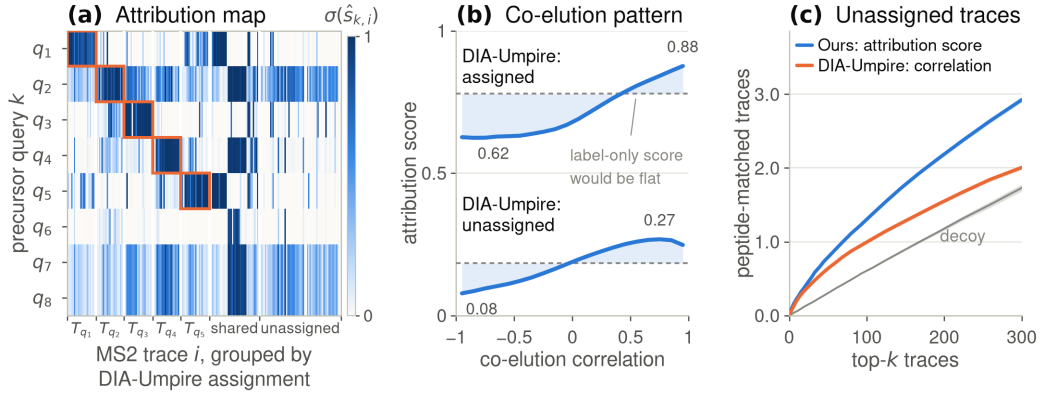
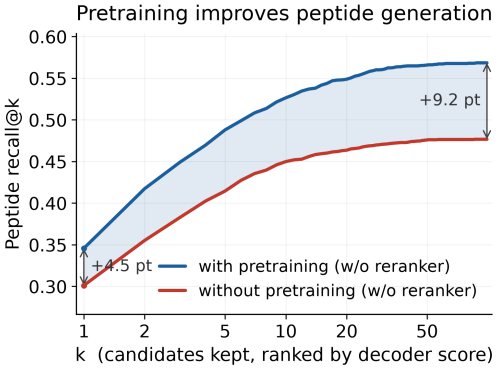


Figure 3 compares peptide recall across the top-k candidates, ranked by decoder score without reranking, with and without attribution pretraining. Pretraining consistently improves top-k recall, with the gain increasing from 4.5 points at $k = 1$ to 9.2 points for larger candidate sets. The widening gap shows that pretraining places the correct peptide in the candidate set more often, although decoder scores do not always rank it first. This separates their roles: pretraining improves candidate coverage, while reranking improves final selection. This motivates reranking (Qiu et al., 2025) to turn higher top-k recall into improved peptide-level accuracy.

Figure 3: Effect of pretraining on generation.



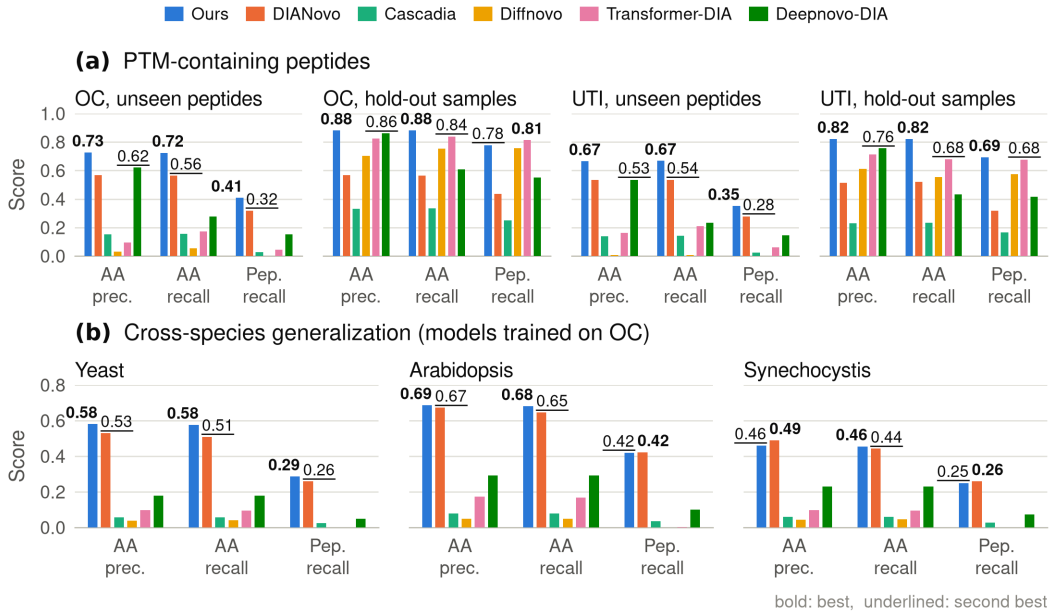
4.4 Analysis of Precursor–Fragment Attribution

Figure 4(a) verifies that attribution pretraining learns the intended precursor-conditioned assignments. We apply a sigmoid to $\hat{s}_{k,i}$ for visualization. Columns are MS2 traces grouped by DIA-Umpire assignment, including traces uniquely assigned to one query (T_{q_k}), shared across queries, and unassigned traces. The approximately block-diagonal map shows high scores for traces assigned to each query and low scores elsewhere. On the OC unseen-peptide test split, mean scores are 0.746 for a query's own traces, 0.138 for traces assigned to another precursor, and 0.053 for unassigned traces, showing that attribution separates target, interference, and background traces as intended.

Figure 4(b) then asks whether the continuous attribution score recovers graded co-elution evidence that is discretized by DIA-Umpire's binary assignments. Co-elution correlation, computed from the observed chromatographic traces, is one of the criteria used by DIA-Umpire to determine these assignments. Regardless of the binary label, the attribution score increases with correlation: from 0.624 to 0.876 for assigned pairs (12.43M) and from 0.078 to 0.268 for unassigned pairs (39.54M). Thus, FreeDIA captures graded precursor–fragment evidence beyond binary assignments.

Finally, Figure 4(c) tests whether the graded attribution score is more useful than co-elution correlation for recovering peptide-matched traces among MS2 traces left unassigned by DIA-Umpire. We define peptide-matched traces as unassigned traces that match theoretical fragments of the corresponding ground-truth peptide. We then rank the unassigned traces by either attribution score or correlation and count peptide-matched traces among the top-k. Attribution-based ranking consistently recovers more peptide-matched traces, indicating that FreeDIA better prioritizes sequence-relevant evidence beyond the original co-elution criterion.

Figure 5: **Generalization across modifications and species.** (a) Performance on PTM-containing peptides in OC and UTI. (b) Cross-species generalization of models trained on OC.



4.5 Generalization Across Species and Modifications

We first evaluate robustness to post-translational modifications (PTMs) by restricting evaluation to peptides containing at least one variable modification (Appendix Table 7). As shown in Figure 5(a), FreeDIA achieves the best AA precision and recall across all four OC and UTI settings, and the best peptide recall in all but the OC hold-out setting, where it is second. Accurate sequencing of modified peptides is particularly important because PTMs expand peptide diversity and alter fragmentation patterns, making them challenging for de novo sequencing. The strong performance of FreeDIA on this subset demonstrates its robustness to such modification-induced spectral variation.

We further evaluate cross-species generalization on yeast, Arabidopsis, and Synechocystis using models trained on OC, excluding overlapping peptides and without species-specific fine-tuning (Appendix Table 8). These species span distant lineages—fungus, plant, and bacterium—with peptide sequence distributions distinct from human. Figure 5(b) shows that FreeDIA transfers strongly across all three, particularly at the amino-acid level, suggesting that spectrum-centric pretraining captures precursor–fragment relationships that remain transferable across large sequence shifts. DIANovo also generalizes well, potentially benefiting from broader cross-species exposure, whereas FreeDIA remains robust despite pretraining only on human DIA data.

5 Conclusion

DIA de novo peptide sequencing remains challenging because multiplexed spectra mix fragments from multiple precursors, while existing pretraining often relies on database-identified peptides. We introduced FreeDIA , a search-free pretraining framework that learns precursor–fragment attribution from chromatographic supervision and transfers it to decoding and reranking. Across unseen-peptide, PTM, and cross-species evaluations, FreeDIA shows strong generalization, while attribution analysis shows that pretraining learns precursor-specific, graded spectral evidence beyond binary assignments. These results support search-free pretraining for unidentified DIA data.

This work focuses on establishing the pretraining task and architecture, and therefore uses a relatively limited DIA pretraining corpus. In addition, the supervision is derived from DIA-Umpire assignments and thus inherits the inductive biases and coverage of the underlying rule, although these assignments are used only as supervision rather than hard filtering. Future work will scale pretraining to more diverse species, instruments, laboratories, and acquisition settings, with the goal of developing a broadly transferable foundation model for DIA proteomics.

AI USE STATEMENT

We used generative AI tools to assist with methodological discussion and feedback, refinement of the presentation of the proposed framework, and translation between Korean and English. We did not use generative AI tools to generate experimental data or results, and other required-disclosure tasks not relevant to this work were not applicable. Additionally, generative AI tools were used for language editing and readability improvements, manuscript organization, literature summarization, brainstorming, and the creation and refinement of scientific figures. All AI-assisted content was critically reviewed, revised, and verified by the authors, including methodological descriptions and claims against the underlying experiments and relevant literature. The authors take full responsibility for the final content of this work, including any text or artifacts produced with the assistance of generative AI.

REFERENCES

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

Shiva Ebrahimi and Xuan Guo. Transformer-based de novo peptide sequencing for data-independent acquisition mass spectrometry. In *2023 IEEE 23rd International Conference on Bioinformatics and Bioengineering (BIBE)*, pp. 28–35. IEEE, 2023.

Shiva Ebrahimi, Jiancheng Li, and Xuan Guo. Diffnovo: A transformer-diffusion model for de novo peptide sequencing. In *International Conference on Bioinformatics and Computational Biology*, pp. 96–104. Springer, 2025.

Kevin Eloff, Konstantinos Kalogeropoulos, Amandla Mabona, Oliver Morell, Rachel Catzel, Esperanza Rivera-de Torre, Jakob Berg Jespersen, Wesley Williams, Sam PB van Beljouw, Marcin J Skwark, et al. Instanovo enables diffusion-powered de novo peptide sequencing in large-scale proteomics experiments. *Nature Machine Intelligence*, 7(4):565–579, 2025.

Ludovic C Gillet, Pedro Navarro, Stephen Tate, Hannes Röst, Nathalie Selevsek, Lukas Reiter, Ron Bonner, and Ruedi Aebersold. Targeted data extraction of the ms/ms spectra generated by data-independent acquisition: a new concept for consistent and accurate proteome analysis. *Molecular & Cellular Proteomics*, 11(6), 2012.

Korrawe Karunratanakul, Hsin-Yao Tang, David W Speicher, Ekapol Chuangsuwanich, and Sira Sriswasdi. Uncovering thousands of new peptides with sequence-mask-search hybrid de novo peptide sequencing framework. *Molecular & Cellular Proteomics*, 18(12):2478–2491, 2019.

Andy T Kong, Felipe V Leprevost, Dmitry M Avtonomov, Dattatreya Mellacheruvu, and Alexey I Nesvizhskii. Msfragger: ultrafast and comprehensive peptide identification in mass spectrometry–based proteomics. *Nature methods*, 14(5):513–520, 2017.

Frank Koopmans, Jenny TC Ho, August B Smit, and Ka Wan Li. Comparative analyses of data independent acquisition mass spectrometric approaches: Dia, wisim-dia, and untargeted dia. *Proteomics*, 18(1):1700304, 2018.

Kai Li, Guo Ci Teo, Kevin L Yang, Fengchao Yu, and Alexey I Nesvizhskii. diatracer enables spectrum-centric analysis of diapasef proteomics data. *Nature Communications*, 16(1):95, 2025.

Zheng Ma, Zeping Mao, Ruixue Zhang, Jiazhen Chen, Lei Xin, Baozhen Shan, Ali Ghodsi, and Ming Li. Disentangling the complex multiplexed dia spectra in de novo peptide sequencing. *arXiv preprint arXiv:2411.15684*, 2024.

Zijie Qiu, Jiaqi Wei, Xiang Zhang, Sheng Xu, Kai Zou, Zhi Jin, Zhiqiang Gao, Nanqing Dong, and Siqi Sun. Universal biological sequence reranking for improved de novo peptide sequencing. *arXiv preprint arXiv:2505.17552*, 2025.

Justin Sanders, Bo Wen, Paul A Rudnick, Richard S Johnson, Christine C Wu, Michael Riffle, Sewoong Oh, Michael J MacCoss, and William Stafford Noble. A transformer model for de novo sequencing of data-independent acquisition mass spectrometry data. *Nature Methods*, 22 (7):1447–1453, 2025.

Ratna R Thangudu, Michael Holck, Deepak Singhal, Alexander Pilozzi, Nathan Edwards, Paul A Rudnick, Marcin J Domagalski, Padmini Chilappagari, Lei Ma, Yi Xin, et al. Nci's proteomic data commons: a cloud-based proteomics repository empowering comprehensive cancer analysis through cross-referencing with genomic and imaging data. *Cancer Research Communications*, 4 (9):2480–2488, 2024.

Ngoc Hieu Tran, Xianglilan Zhang, Lei Xin, Baozhen Shan, and Ming Li. De novo peptide sequencing by deep learning. *Proceedings of the National Academy of Sciences*, 114(31):8247–8252, 2017.

Ngoc Hieu Tran, Rui Qiao, Lei Xin, Xin Chen, Chuyi Liu, Xianglilan Zhang, Baozhen Shan, Ali Ghodsi, and Ming Li. Deep learning enables de novo peptide sequencing from data-independent-acquisition mass spectrometry. *Nature methods*, 16(1):63–66, 2019.

Chih-Chiang Tsou, Dmitry Avtonomov, Brett Larsen, Monika Tucholska, Hyungwon Choi, Anne-Claude Gingras, and Alexey I Nesvizhskii. Dia-umpire: comprehensive computational framework for data-independent acquisition proteomics. *Nature methods*, 12(3):258–264, 2015.

Melih Yilmaz, William E Fondrie, Wout Bittremieux, Carlo F Melendez, Rowan Nelson, Varun Ananth, Sewoong Oh, and William Stafford Noble. Sequence-to-sequence translation from mass spectra to peptides with a transformer model. *Nature communications*, 15(1):6427, 2024.

Jingbo Zhou, Shaorong Chen, Jun Xia, Sizhe Sizhe Liu, Tianze Ling, Wenjie Du, Yue Liu, Jianwei Yin, and Stan Z Li. Novobench: Benchmarking deep learning-based\emph {De Novo} sequencing methods in proteomics. *Advances in Neural Information Processing Systems*, 37: 104776–104791, 2024.

A Trace Embedding Details

We describe the embedding function g_{emb} introduced in Section 3.2. Each trace is encoded using its m/z, chromatographic profile, intensity statistics, and temporal context, together with a learned MS-level embedding.

Profile and intensity features. Starting from the intensity profile $\mathbf{x}_i$, we apply a log transformation and normalize the resulting profile:

$$\begin{aligned} \bar{\mathbf{x}}_i &= \log(1 + \mathbf{x}_i), \qquad \tilde{\mathbf{x}}_i = \frac{\bar{\mathbf{x}}_i}{\|\bar{\mathbf{x}}_i\|_\infty}, \\ \mathbf{v}_i &= \left(\|\bar{\mathbf{x}}_i\|_\infty, \, \frac{n_i}{C} \right). \end{aligned} \tag{15}$$

where n_i is the number of cycles with nonzero intensity. The normalized profile $\tilde{\mathbf{x}}_i$ describes the relative elution pattern, while $\mathbf{v}_i$ retains the peak log-intensity and the fraction of observed cycles.

Mass and temporal features. Following Casanovo (Yilmaz et al., 2024), we encode m/z using sinusoidal features:

$$\phi_\mu(\mu_i) = \left[\sin\left(\frac{2\pi\mu_i}{\lambda_k} \right), \cos\left(\frac{2\pi\mu_i}{\lambda_k} \right) \right]_{k=1}^{d/2}, \tag{16}$$

where the wavelengths λ_k are geometrically spaced from 10^{-3} to 10^4. The cycle-level retention-time offsets $\Delta\mathbf{t} \in \mathbb{R}^C$ provide the temporal context shared by traces in the same group. The profile encodes the apex position within this window, so ρ_i is not embedded separately.

Feature fusion. Three feed-forward networks map the normalized profile, intensity statistics, and time offsets to d-dimensional vectors. We concatenate the four feature embeddings, apply layer normalization, and project them into one token:

$$\mathbf{e}_i = \mathrm{LN}\left[\underbrace{\phi_\mu(\mu_i)}_{\text{mass}}; \underbrace{\psi_{\text{s}}(\tilde{\mathbf{x}}_i)}_{\text{profile}}; \underbrace{\psi_{\text{l}}(\mathbf{v}_i)}_{\text{intensity}}; \underbrace{\psi_{\text{t}}(\Delta\mathbf{t})}_{\text{timing}} \right] W + \mathbf{u}_{\ell_i}, \tag{17}$$

where $[\cdot;\cdot]$ denotes feature concatenation, $W \in \mathbb{R}^{4d \times d}$, and $\mathbf{u}_{\ell_i} \in \mathbb{R}^d$ is the learned MS-level embedding. No positional encoding is added over trace indices; reordering the traces therefore only reorders the encoder outputs.

B Trace Window Construction

Trace linking. We construct MS1 and MS2 traces separately within each isolation window by linking signals whose m/z values agree within 20 ppm across consecutive acquisition cycles. Each trace group $\mathcal{G} = (\mathcal{T}^1, \mathcal{T}^2)$ corresponds to an isolation-window and retention-time-window pair; each run therefore contributes multiple groups per isolation window.

Window size and coverage. Each group contains $C = 2w + 1$ neighboring cycles, with $w = 8$ and $C = 17$. This window is chosen to represent the elution profile together with neighboring temporal context for precursor–fragment co-elution. Table 3 summarizes trace statistics for ccRCC from the pretraining corpus and for the downstream OC and UTI cohorts. Across these datasets, the median precursor FWHM is 3 cycles and the median number of observed cycles per trace is 7. A fixed number of cycles corresponds to different retention-time spans because acquisition cycle times differ between datasets.

Some precursor profiles extend beyond the window. We quantify boundary truncation as the fraction of precursor profiles whose intensity remains above half maximum at either window boundary. The corresponding rates are reported in Table 3; for these profiles, co-elution is estimated from the portion observed within the window.

Comparison with a wider window. We also evaluated $w = 16$ ($C = 33$). At matched training epochs, validation attribution average precision was 0.795 for $C = 33$, compared with 0.806 for $C = 17$, while the number of traces per group increased by a factor of 1.47. We therefore retain $C = 17$ for the reported experiments. This comparison evaluates the two specified window sizes and does not establish a globally optimal window size.

Table 3: **Trace window statistics.** Statistics are computed using 40 groups per run from 40 ccRCC runs (56,847 precursors) and all OC and UTI runs (≈3,900 precursors per cohort). FWHM is computed from linear intensity. Window span is the elapsed time between the first and last included cycles.

	ccRCC (pretrain)	OC	UTI
Cycle time (s)	4.00	2.09	2.05
Window span, $C = 17$ (s)	64.0	33.5	32.7
Precursor FWHM, median (cycles)	3	3	3
Observed cycles per trace, median	7	7	7
Boundary truncation (%)	17.6	13.1	12.7
MS2 traces per group, median	4,622	2,634	3,336

C Inference Details

Decoding. Each precursor query is supplied with the observed m/z and charge from the same DIA-Umpire analysis used to construct pretraining supervision, without peptide identification. We perform beam search separately with the N-to-C and C-to-N decoders, using beam width $B = 50$ for each direction. Early termination is permitted: a hypothesis that emits <eos> with unused precursor mass receives a penalty proportional to the remaining mass. Within each directional beam, hypotheses satisfying the precursor-mass tolerance take priority over those that do not, with length-normalized log probability determining the order within each group.

Candidate pooling and rescoring. Candidates from both directions are pooled and deduplicated to form $\hat{\mathcal{P}}_k$. Each candidate is rescored by both decoders under teacher forcing. Its decoder score is $s_{\mathrm{dec}}(\hat{P}) = \ell_{\rightarrow}(\hat{P}) + \ell_{\leftarrow}(\hat{P})$, where each ℓ denotes the length-normalized log probability evaluated in the corresponding decoding direction.

Spectrum-evidence reranking. For each candidate $\hat{P} \in \hat{\mathcal{P}}_k$, we match the theoretical b^+, y^+, b^{2+}, and y^{2+} ions to their nearest MS2 traces within 20 ppm. The resulting features describe match presence, log intensity, relative mass error, and precursor–fragment co-elution. A Transformer encodes features across cleavage positions and pools them into a candidate representation. This representation is combined with candidate-level features, including decoder score and rank, peptide length, precursor-mass residual, charge, and agreement between ion ladders across candidates. A second Transformer jointly processes the candidates, allowing each score to depend on the other candidates in $\hat{\mathcal{P}}_k$.

We average the scores of three rerankers trained with different seeds and standardize the resulting ensemble score within each pool, denoted $\tilde{r}_\xi(P; \mathcal{P}_k)$. This reranker ensemble is used when evaluating each generator seed; the reported standard deviations reflect variation across the three generator training seeds. The final prediction combines decoder likelihood with spectral evidence:

$$\hat{P}_k = \arg\max_{\hat{P} \in \hat{\mathcal{P}}_k} \left[s_{\mathrm{dec}}(\hat{P}) + \lambda\, \tilde{r}_\xi(\hat{P}; \hat{\mathcal{P}}_k) \right]. \tag{18}$$

The mixing weight λ is selected using the validation fold for each cohort and evaluation setting. The selected values are $\lambda = 1.0$ for unseen-peptide evaluation and $\lambda = 0.25$ for sample-holdout evaluation in both OC and UTI.

Reranker training. The similarity $\mathrm{sim}(\hat{P}, P_k)$ in Eq. (14) is [the exact similarity definition used in the implementation]. We construct reranker training pools using two-fold cross-fitting. The fine-tuning training set is divided into two halves; a generator is fine-tuned on each half and used to

generate candidates for the opposite half. Each training pool is therefore produced by a generator that was not fine-tuned on the corresponding PSM. Validation pools are generated by the generator fine-tuned on the full training set and are used only to select λ.

Computational resources. Experiments use a single node with four NVIDIA RTX PRO 6000 Blackwell GPUs (96 GB each) and an AMD EPYC 9655 CPU. Pretraining and directional fine-tuning use four GPUs, while decoding and reranker training use one GPU. Table 4 summarizes model sizes and computational costs.

Table 4: **Model sizes and computational costs.** Decoding time is measured on the OC unseen-peptide test split (3,825 PSMs) with $B = 50$ in each direction and includes candidate pooling and teacher-forced rescoring. Reranker feature extraction is reported separately.

Component	**Parameters**
Encoder	39.2 M
Peptide decoder	25.8 M
Attribution and precursor heads	0.56 M
Reranker, per model	0.61 M (hidden width 128)
Stage	**Runtime**
Pretraining, 30 epochs in `bf16`	10.8 h
Fine-tuning	1–3 h per direction
Bidirectional decoding and rescoring	≈40 min
Reranker feature extraction (CPU)	A few minutes per split
Reranker training	≈4 min per model

D TRAINING DETAILS

Table 5: Initialisation, training and decoding of the compared models. Downstream checkpoints are selected using the validation fold for each cohort and evaluation setting. As in Table 1, results are reported as means over three random seeds, with standard deviations in parentheses, unless otherwise stated.

model	initialisation	training	decoding
FREEDIA	CPTAC stage-1 encoder; decoder random	see below	bidirectional beam 50, soft mass penalty, reranked
DIANovo	released Astral checkpoints, resized to the widened alphabet	both stages fine-tuned: 5 and 8 epochs (seed 0: 10 and 5), val every 2,000 steps	two-stage, beam 20, knapsack
Cascadia	released pretrained checkpoints	fine-tuned per fold: batch 32, lr 10^{-5}, $\leq$ 30 epochs; per-scan normalisation and top-150 peaks enabled (+23% rel. peptide recall vs. authors' default)	greedy, no mass constraint
DeepNovo-DIA	random; the released checkpoint uses the authors' alphabet, not ours	see below; batch 32, five spectra stacked, one feature per example	bidirectional beam 5 (authors' default), knapsack, 0.01 Da
Transformer-DIA	random; the released checkpoint uses the authors' alphabet, not ours	see below; batch 128, five spectra stacked, one feature per example	beam 10, mass-constrained termination, 50 ppm, isotopes 0–1
DiffNovo	random; the released checkpoint uses the authors' alphabet, not ours	see below; batch 256, one scan per example	beam 10, diffusion denoising, mass post-correction, 50 ppm

Cascadia decodes greedily with no precursor-mass constraint: on OC *sample_holdout* its prediction closes the precursor mass on 32.2% of PSMs and is exactly right on 31.6%, i.e. the mass closes essentially only when the peptide is already correct.

FREEDIA. Attribution pretraining transfers to peptide sequencing through the pretrained encoder representations; attribution scores are not directly used in the peptide decoder. *Stage 1*: encoder of 12 layers, $d = 512$, 8 heads, feed-forward 2,048, dropout 0.1 (39.2 M parameters); 30 epochs on 372 CPTAC runs; AdamW, lr 4×10^{-4}, weight decay 0.01, 1,400 warm-up steps, cosine decay; 16 groups per GPU on four GPUs in `bf16`; ≤ 32 queries per group; $\lambda_{\text{own}} = \lambda_{\text{prec}} = \lambda_{\text{chg}} = 0.5$; final-epoch checkpoint. *Stage 2*: decoder of 6 layers, feed-forward 2,048 (25.8 M); encoder unfrozen after the first 3 epochs; ≤ 120 epochs, early stopping with patience 10 on validation amino-acid precision, which also selects the checkpoint; AdamW, lr 10^{-4}, 300 warm-up steps; $\lambda_{\text{mass}} = 1$; N-to-C and C-to-N decoders trained separately. *Reranker*: width 128, two layers over sites and two over candidates (0.61 M); 30 epochs, AdamW lr 3×10^{-4}, batch 32, $\tau = 0.05$; scores from three independently trained rerankers are averaged; the resulting reranker ensemble, trained on OC unseen-peptide pools, is used across all evaluation settings. Mixing weight λ from each cell's validation fold: 1.0 on *unseen_peptide*, 0.25 on *sample_holdout*.

DeepNovo-DIA. The released checkpoint is built on the authors' alphabet, whose modifications differ from ours and whose 26 tokens cannot be resized to it, so every fold is trained from random initialisation: convolutional spectrum encoder and LSTM decoder, dropout 0.25/0.5 (conv/dense) (15.4 M parameters). Adam at its TensorFlow default lr 10^{-3} with no warm-up and no decay, gradients clipped at global norm 5, applied per step; batch 32; sequences bucketed by length at 12/22/32. Each feature is one training example: the five spectra nearest the feature's retention-time centre are stacked into one input, each discretised to 150,000 bins at 0.02 Da, and shorter neighbourhoods are zero-padded to five; precursors above 3,000 Da are dropped at annotation. Validation runs once per epoch on a fixed 1,500-feature stack drawn from the cell's validation fold and selects the checkpoint by validation peptide recall; training stops after 20 epochs without improvement, capped at 200. The length cap of 30 is a training-time filter only, removing 11–13 features per fold (0.04%); it does not bound decoding, where the beam extends until the precursor mass is spent.

Transformer-DIA. The released checkpoint is built on the authors' alphabet, whose modifications differ from ours (`N+0.984`, `Q+0.984`, `+43.006`, `-17.027`), so every fold is trained from random initialisation: 9 encoder and 9 decoder layers, $d = 512$, 8 heads, feed-forward 1,024, dropout 0 (53.4 M parameters, of which 24.7 M encoder and 28.7 M decoder). Adam at lr 10^{-4} with weight decay 10^{-5}, 1,000 linear warm-up steps followed by cosine decay over a 600,000-step horizon, applied per step; batch 128. Each feature is one training example, in the same input format as DeepNovo-DIA: the five spectra nearest the feature's retention-time centre are stacked into one input, each discretised to 150,000 bins at 0.02 Da, and shorter neighbourhoods are zero-padded to five; the encoder additionally receives the top 150 peaks of the centre scan between 50 and 3,000 m/z and the feature's MS1 elution profile. Precursors above 3,000 Da are dropped at annotation, as for DeepNovo-DIA. Validation runs once per epoch under teacher forcing and selects the checkpoint by validation amino-acid recall; training stops after 20 epochs without improvement, capped at 200.

DiffNovo. The released checkpoint is built on the authors' alphabet, whose modifications differ from ours, so every fold is trained from random initialisation: 9 encoder and 9 decoder layers, $d = 512$, 8 heads, feed-forward 1,024, dropout 0 (49.2 M parameters, of which 20.5 M encoder and 28.7 M decoder). Adam at lr 10^{-4} with weight decay 10^{-5}, 1,000 linear warm-up steps followed by cosine decay over a 600,000-step horizon, applied per step; batch 256. Each scan is one training example: unlike DeepNovo-DIA and Transformer-DIA, DiffNovo has no neighbour axis, its encoder consuming one flat list of $(m/z, \text{intensity})$ pairs plus a scalar retention time, so each of the five retention-time-nearest MS2 scans of a feature is a separate example rather than a row of a stacked tensor and we give every scan its own spectrum identifier; the input is the top 150 peaks of each spectrum between 50 and 3,000 m/z, and precursors above 3,000 Da are dropped at annotation, as for DeepNovo-DIA. Validation runs once per epoch under teacher forcing and selects the checkpoint by validation amino-acid recall; training stops after 30 epochs without improvement, capped at 200. At test time the full pipeline runs: beam search of width 10, the diffusion reverse process, and the post-correction that substitutes residues until the precursor mass closes; a beam terminates when its mass falls within 50 ppm of the precursor, over isotopes 0–1, or overruns it, and otherwise at a length of 100.

E Comparison Protocol

Fair comparison setup.

Comparison setup. All methods use common train, validation, and test partition definitions for each cohort and evaluation setting. The sample-holdout split partitions complete runs, whereas the unseen-peptide split partitions peptide sequences without sequence overlap between train, validation, and test sets. For the unseen-peptide and cross-species evaluations, we additionally verified that test peptide sequences do not overlap with the peptide labels used to train the reranker.

All methods use MSFragger identifications at $q \leq 0.01$ as ground truth and, whenever supported by their architecture, DIA-Umpire-derived precursor m/z and charge. For cross-species evaluation, the common cohort manifests are processed by model-specific input pipelines, resulting in different evaluation PSM sets as detailed in Appendix F.

Table 6: Evaluation settings used for comparison.

Setting	Protocol
Data split	Identical train/validation/test partitions are used for all methods within each cohort and evaluation setting.
Ground truth	Ground-truth labels are drawn from common MSFragger identifications at $q \leq 0.01$; cross-species evaluation PSM sets differ after model-specific input processing.
Precursor information	Models receive the same DIA-Umpire-derived precursor m/z and charge when supported by their original architecture.
Modification vocabulary	The evaluation alphabet includes fixed carbamidomethyl-C and five modification tokens: M+15.995, C−17.027, Q−17.027, E−18.011, and +42.011. Baseline vocabularies are extended when possible rather than removing unsupported test examples.
Scoring	Predictions are converted to residue masses and evaluated with the same Novor-style matching criterion: 0.5 Da prefix-mass tolerance and 0.1 Da residue-mass tolerance. I/L and equivalent modification spellings are collapsed.
Missing predictions	Empty, unparseable, or unsupported predictions are counted as misses rather than removed from evaluation.

These choices standardize the partition definitions, annotation source, and scoring criteria, although cross-species comparisons use model-specific evaluation PSM sets. Architecture-specific constraints that cannot be modified without changing the original method are retained and reported separately.

F Details for Section 4.5

PTM-containing peptides (Table 7). The label vocabulary follows the FragPipe search that produced the training labels. Carbamidomethyl-Cys is a fixed modification, folded into the C token, and is not counted. Five variable modifications have their own tokens: Met oxidation (Ox), protein N-terminal acetylation (Ac), and the N-terminal cyclisations pyro-Glu, pyro-Gln and pyro-carbamidomethyl-Cys (pyro-cmC). A PSM is PTM-containing when at least one of its tokens is a variable modification; the PTM metrics score the modified residue as its own symbol, so a dropped or misplaced modification is an error. All five types occur in the training folds, so the PTM evaluation measures the recovery of known modification types on unseen peptides or held-out samples, not generalization to modification types absent from training.

Cross-species test folds (Table 8). All three cohorts are single-species whole-cell lysates acquired by Orbitrap DIA with fixed windows of about 10 *m/z* and searched against the species' own UniProt reference proteome. Before sampling the test folds, we excluded exact peptide-sequence matches, treating I and L as equivalent, against the following reference sets: the OC and UTI training and validation peptides, the stage-1 pretraining peptide set used for overlap checking, an in-silico human tryptic-peptide corpus, and the human, yeast, and *E. coli* reference proteomes. Proteome-based

Table 7: PTM-containing subsets. Cells are PSMs (distinct peptidoforms) carrying each variable modification; a PSM with two modifications is counted under both. Training folds are shown for reference.

Set	PSMs	PTM PSMs	Ox (M)	Ac (N-term)	pyro-Glu	pyro-Gln	pyro-cmC
OC, unseen peptides	3,825	455 (11.9%)	141 (30)	34 (5)	2 (2)	155 (33)	125 (29)
OC, held-out samples	4,336	544 (12.5%)	188 (149)	46 (36)	28 (20)	159 (115)	136 (96)
UTI, unseen peptides	5,848	864 (14.8%)	384 (90)	51 (15)	60 (18)	216 (37)	174 (22)
UTI, held-out samples	5,186	728 (14.0%)	515 (390)	45 (35)	24 (19)	93 (57)	68 (47)
OC train fold	28,256	3,253 (11.5%)	1,197 (324)	227 (65)	221 (58)	819 (159)	859 (162)
UTI train fold	35,599	3,566 (10.0%)	1,780 (602)	325 (110)	209 (72)	679 (182)	652 (163)

matching considered both in-silico tryptic peptides and contiguous protein subsequences. For the yeast test cohort, the yeast reference proteome was omitted from the exclusion sets. After exclusion the overlap between test peptides and OC training peptides, and between test peptides and the human proteome, is zero.

Table 8: Zero-shot cohorts. "In OC train" and "In human proteome" count exact-sequence matches among all peptides identified in the cohort; every such peptide is in the excluded set. "Manifest test PSMs" reports the test-set size recorded in the cohort manifest, rather than the evaluation denominator used in Table 9.

Species	PRIDE	Instrument	Runs	Peptides	In OC train	In human proteome	Excluded (all sources)	Test peptides	Manifest test PSMs
S. cerevisiae	PXD075335	Orbitrap Exploris 480	10	6,177	21	93	124 (2.0%)	6,053	9,807
A. thaliana (leaf)	PXD067382	Orbitrap Exploris 480	16	7,417	37	122	209 (2.8%)	7,208	9,821
Synechocystis sp. PCC 6803	PXD059631	Q Exactive HF	20	4,933	5	23	91 (1.8%)	4,842	9,756

Pretraining ablation (Table 9). The two models share the architecture, the stage-2 recipe, the OC unseen-peptide training fold and the seed; the scratch model starts from random initialisation instead of the stage-1 checkpoint. Both are decoded identically (beam 10, bidirectional ensemble) and scored with pooled metrics on the same PSMs, so the differences are attributable to pretraining alone; the decoding configuration differs from that of the main results, and the absolute values are not comparable to Figure 5. The evaluation denominators are 9,678 PSMs for yeast, 9,792 for Arabidopsis, and 9,514 for Synechocystis, whereas the corresponding cohort manifests contain 9,807, 9,821, and 9,756 test PSMs, respectively.

Table 9: Effect of pretraining, seed 0. Δ is pretrained minus scratch in percentage points.

Test set	Model	n	AA prec.	AA recall	Pep. recall
OC unseen peptides	Pretrained	3,825	0.607	0.606	0.333
	Scratch	3,825	0.462	0.469	0.287
	Δ (pp)		+14.5	+13.7	+4.6
OC unseen, PTM subset	Pretrained	455	0.588	0.588	0.305
	Scratch	455	0.447	0.455	0.237
	Δ (pp)		+14.2	+13.3	+6.8
Yeast	Pretrained	9,678	0.492	0.489	0.202
	Scratch	9,678	0.195	0.215	0.082
	Δ (pp)		+29.7	+27.5	+12.0
Arabidopsis	Pretrained	9,792	0.583	0.582	0.301
	Scratch	9,792	0.263	0.301	0.107
	Δ (pp)		+32.1	+28.1	+19.5
Synechocystis	Pretrained	9,514	0.365	0.367	0.178
	Scratch	9,514	0.165	0.204	0.089
	Δ (pp)		+20.0	+16.3	+9.0

Table 10: Ablation study on UTI. Mean results over three random seeds are reported with standard deviations in parentheses.

Model	AA Prec.	AA Rec.	Pep. Rec.
FREEDIA	**0.6980** (0.0045)	**0.6987** (0.0037)	**0.4368** (0.0076)
w/o Pretrain	0.5862 (0.0078)	0.5872 (0.0081)	0.3701 (0.0038)
w/o Reranker	0.6543 (0.0037)	0.6575 (0.0038)	0.3486 (0.0039)
w/o Both	0.5104 (0.0113)	0.5158 (0.0112)	0.2891 (0.0036)

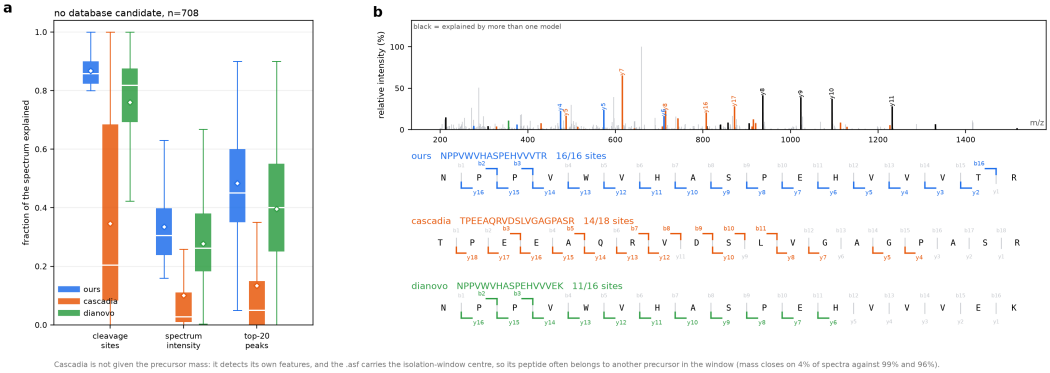


Figure 6: **Recovering peptides beyond database search.** (a) Fraction of pseudo-spectrum evidence explained by each model on spectra without an available database candidate. (b) Representative spectrum showing fragment evidence supporting the peptide predicted by FREEDIA relative to DIANovo and Cascadia.

G ADDITIONAL ABLATION RESULTS ON UTI

Table 10 reports the ablation results on UTI. The trends are consistent with those observed on OC: removing pretraining causes a larger drop in AA precision and recall, whereas removing the reranker more strongly reduces peptide recall. Specifically, removing pretraining decreases AA precision by 11.2 percentage points, while removing the reranker decreases peptide recall by 8.8 points. Removing both components results in the largest degradation across all metrics, further supporting their complementary roles.

H CASE STUDY: RECOVERING PEPTIDES BEYOND DATABASE SEARCH

Database search can report only peptides represented in its search space, leaving potentially interpretable spectra unidentified. From the OC cohort, we therefore examined high-quality DIA-Umpire features that were not identified by database search and applied a calibrated spectral-evidence gate to FREEDIA's de novo predictions. Among 19,597 spectra passing the initial quality filters, 880 satisfied the evidence gate, of which 862 yielded peptides that could not be explained by any candidate available to the original database search.

Figure 6(a) compares how completely each model's predicted peptide accounts for the same pseudo-spectrum. Across spectra answered by all three models, FREEDIA explains more cleavage-site and spectral evidence than DIANovo and Cascadia; Cascadia should be interpreted separately because it is not provided the precursor mass and may predict a different precursor within the same isolation window. Because the candidate set itself is selected using FREEDIA's spectral evidence, its distribution is necessarily bounded by the gate. Figure 6(b) shows a representative case in which FREEDIA and DIANovo share most of the sequence but differ at the C-terminus; the distinguishing

y ions are present only for the FREEDIA prediction, indicating that the preference is supported by observed fragment evidence rather than sequence plausibility alone.