

# Pep2Prob Benchmark: Predicting Fragment Ion Probability for MS<sup>2</sup>-based Proteomics

Hao Xu<sup>1,\*</sup>, Zhichao Wang<sup>1,\*</sup>, Shengqi Sang<sup>2</sup>, Pisit Wajanasara<sup>1</sup>, Nuno Bandeira<sup>1</sup>

<sup>1</sup>University of California, San Diego, <sup>2</sup>Stanford University

{hax019, zhiw036, pwajanasara, bandeira}@ucsd.edu, {sangsq@stanford.edu}

## Abstract

Proteins perform nearly all cellular functions and constitute most drug targets, making their analysis fundamental to understanding human biology in health and disease. Tandem mass spectrometry (MS<sup>2</sup>) is the major analytical technique in proteomics that identifies peptides by ionizing them, fragmenting them, and using the resulting mass spectra to identify and quantify proteins in biological samples. In MS<sup>2</sup> analysis, peptide fragment ion probability prediction plays a critical role, enhancing the accuracy of peptide identification from MS<sup>2</sup> spectra as a complement to the intensity information. Current approaches rely on global statistics of fragmentation, which assumes that a fragment’s probability is uniform across all peptides. Nevertheless, this assumption is oversimplified from a biochemical principle point of view and limits accurate prediction. To address this gap, we present **Pep2Prob**, the first comprehensive dataset and benchmark designed for peptide-specific fragment ion probability prediction. The proposed dataset contains fragment ion probability statistics for 608,780 unique precursors (each precursor is a pair of peptide sequence and charge state), summarized from more than 183 million high-quality, high-resolution, HCD MS<sup>2</sup> spectra with validated peptide assignments and fragmentation annotations. We establish baseline performance using simple statistical rules and learning-based methods, and find that models leveraging peptide-specific information significantly outperform previous methods using only global fragmentation statistics. Furthermore, performance across benchmark models with increasing capacities suggests that the peptide-fragmentation relationship exhibits complex nonlinearities requiring sophisticated machine learning approaches. Pep2Prob provides a standardized evaluation framework that will accelerate algorithmic innovation in computational proteomics while introducing a biologically significant prediction task to the machine learning community.

## 1 Introduction

Proteomics, the large-scale study of proteins, provides critical insights into cellular function, disease mechanisms, and potential therapeutic targets [32]. Tandem mass spectrometry (MS<sup>2</sup>) has emerged as the predominant analytical technique for identifying and quantifying proteins in complex biological samples [1]. In MS<sup>2</sup>-based proteomics, proteins are first digested enzymatically into peptides, ionized and separated by their mass-to-charge ratio ( $m/z$ ); then the selected peptide ions are fragmented, generating fragment ion spectra that serve as “fingerprints” for peptide identification.

The interpretation of MS<sup>2</sup> spectra represents a fundamental pattern recognition challenge where computational methods match observed spectral patterns to peptide sequences. Core approaches—database search [17, 35], de novo sequencing [9, 34, 42], and spectral library matching [7, 27, 37]—all require accurate prediction of peptide fragmentation behavior as a critical prerequisite.

---

\*Equal Contribution

Fragment ion probability prediction (see Figure 1) serves as a key intermediate task that directly impacts several downstream applications including peptide identification [18], post-translational modification (PTM) localization [40], and protein quantification [21]. This probability provides complementary information to intensity modeling, effectively serving as a confidence weighting mechanism that distinguishes signal from noise in complex biological samples.

Current fragment ion prediction methods widely used in peptide identification workflows [2, 4, 17] rely on global statistics that treat all peptides uniformly, assuming fragmentation probabilities depend only on a fragment’s partial information, namely fragment type (e.g., *b*-ions vs. *y*-ions) and charge state. Such approaches ignore peptide-specific features entirely—analogous to predicting natural language tokens using only part-of-speech tags while ignoring word identity and context. In reality, peptide fragmentation is governed by complex sequence-dependent factors: neighboring amino acids influence bond stability, charge mobility varies with sequence composition, and local chemical environments create position-specific effects. By modeling  $\mathbb{P}(\text{fragment}|\text{peptide})$  as  $\mathbb{P}(\text{fragment})$ , existing methods sacrifice crucial predictive information, resulting in suboptimal performance for peptides that deviate from average fragmentation patterns.

To address these limitations, we introduce **Pep2Prob**, the first large-scale dataset together with a comprehensive benchmark specifically designed for peptide-specific fragment ion probability prediction, available at <https://huggingface.co/datasets/bandeiralab/Pep2Prob>. Pep2Prob comprises fragment probability statistics for 608,780 unique precursors, derived from over 183 million high-resolution higher-energy collisional dissociation (HCD) MS<sup>2</sup> spectra with validated peptide assignments. We establish comprehensive benchmarks using models of increasing capacity—including two simple statistical baselines, linear regression, neural network, and transformer—systematically evaluating their ability to capture peptide-specific fragmentation patterns. Our empirical analysis reveals two key findings:

1. Incorporating peptide sequence information yields substantial performance improvements over global statistics. In particular, a simple empirical statistical method incorporating only fragment sequence information achieves  $\approx 0.18$  test set L1 loss compared to  $\approx 0.24$  from conventional global modeling methods using only ion type and charge information.
2. After including peptide-specific information, we observe that prediction accuracy continuously improves with increasing model capacity: test set L1 losses decrease from 0.126 (linear regression) to 0.069 (neural network) to 0.056 (transformer). This trend indicates that the relationship between peptide sequences and fragmentation probabilities exhibits complex nonlinearities that simple models fail to capture. These results demonstrate that effective fragment ion prediction requires sophisticated machine learning (ML) approaches.

Pep2Prob offers the proteomics and ML communities a standardized framework to advance fragment ion prediction beyond its current limitations. We anticipate that the complex sequence-to-fragmentation relationships captured in our dataset will inspire novel ML algorithms for this task, yielding immediate practical benefits for downstream proteomics applications, including peptide identification, PTM localization, and biomarker discovery.

## 2 Problem Formulation, Task Definition and Notations

A peptide is a short chain of amino acids linked by peptide bonds, serving as the fundamental unit of protein identification in proteomics. In MS<sup>2</sup>, peptides are first ionized and isolated as precursor ions, which are specific peptides with defined charge states:

$$\text{precursor} = (\text{peptide } \textit{sequence}, \textit{charge state}) \quad \text{e.g.} \quad (\text{PEPTIDE}, 2+). \quad (2.1)$$

In the example above, each letter in ‘PEPTIDE’ represents an amino acid, and 2+ indicates the precursor has 2 positive charges. These precursors are then fragmented through collisional activation, breaking peptide bonds to produce characteristic fragment ions. The resulting MS<sup>2</sup> spectrum, a collection of fragment ion peaks at specific mass-to-charge ratios, serves as a molecular fingerprint that enables peptide identification.

As illustrated in Figure 1(A), fragment ions are the products of peptide fragmentation, each uniquely specified by three key attributes:

$$\text{fragment ion} = (\text{ion } \textit{type}, \textit{charge}, \textit{position } \textit{number}) \quad (2.2)$$

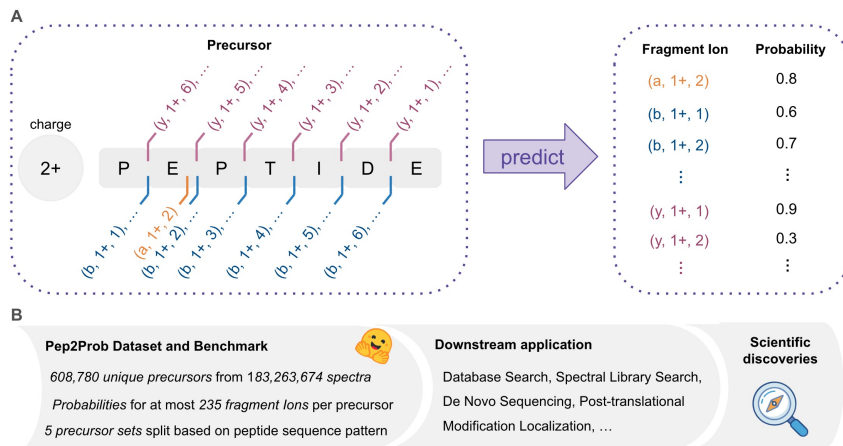


Figure 1: (A) Illustration of the task of fragment ion probability prediction, aiming to estimate the likelihood that each potential fragment ion will be observed in the  $MS^2$  spectrum of a precursor. (B) Pep2Prob establishes the *first* machine learning dataset and benchmark for peptide-specific fragment ion probability prediction, with rich applications in proteomics study.

The ion type  $t$  indicates which portion of the original peptide is retained:  $a$ - and  $b$ -ions contain the prefix while  $y$ -ions contain the suffix. Here  $a$ - and  $b$ -ions further distinguish the chemical structure of the prefix fragment, where  $a$ -ions result from the loss of CO from corresponding  $b$ -ions ( $a$ -ion is only possible when  $c = +1$  and  $n = 2$ ). Note that it is commonly assumed that each fragment is produced by a single cut, hence each fragment is either a prefix or a suffix. Next, the charge state  $c$  specifies how many positive charges the fragment carries. Finally, the position number  $n$  denotes where the peptide bond was cleaved, counted from the respective end. For instance,  $(b, 2+, 3)$  denotes a doubly-charged  $b$ -ion fragment containing the first three amino acids, and  $(y, 1+, 5)$  denotes a singly-charged  $y$ -ion fragment containing the last five amino acids.

During  $MS^2$  analysis, fragmented ions are then separated by their mass-to-charge ratios and detected to generate a spectrum. Formally, a spectrum is a set of peaks:

$$S = \{s_1, s_2, \dots\} \quad s_i = (m/z_i, I_i) \quad (2.3)$$

where each element is a peak defined by its mass-to-charge ratio ( $m/z_i$ ) and intensity ( $I_i$ ). Each observed peak originates from a specific fragment ion of the precursor, while the intensity quantifies how frequently that particular fragmentation occurs.

Not all theoretically possible fragments appear as peaks in the spectrum, and only a small subset generates detectable signals. This could be due to differences in the stability of the peptide bond, the preferences of amino acid-specific fragmentation, and instrument detection limits. The task of **fragment ion probability prediction** aims to estimate the likelihood that each potential fragment will be observed in the  $MS^2$  spectrum of a precursor. We mathematically denote this likelihood as:

$$\mathbb{P}(f|p) \stackrel{\text{def}}{=} \mathbb{P}(\text{fragment ion } f \text{ shows up in the precursor } p\text{'s spectrum}) \quad (2.4)$$

The quantity serves as complementary information to intensity values, enhancing peptide identification algorithms by distinguishing likely fragments from theoretical possibilities.

Existing methods [17, 2, 4] for modeling  $\mathbb{P}(f|p)$  only take the fragment ion  $f$  as input and ignore all precursor  $p$  information. This amounts to assuming that  $\mathbb{P}(f|p)$  is uniform across all precursors/peptides, which is unreasonable from a biochemical perspective: different peptide sequences exhibit distinct fragmentation patterns due to varying amino acid properties and bond stabilities. The aim of Pep2Prob is to demonstrate that  $\mathbb{P}(f|p)$  has a nontrivial dependency on precursor information (peptide sequence and charge state), and that incorporating this information in the prediction of fragment ion probability improves performance significantly.

### 3 Pep2Prob Dataset Construction

Pep2Prob is built upon 227 human HCD mass spectrometry datasets and their associated peptide-spectrum matches, which are publicly accessible via the Mass Spectrometry Interactive Virtual Environment (**MassIVE**) repository (<http://massive.ucsd.edu>) [20]. We also leverage the MassIVE Knowledge Base (**MassIVE-KB**) in vivo library (version 2.0.15) [38], which was curated from these HCD mass spectrometry datasets, for precursor selection. The MassIVE-KB library contains a reference spectrum for each of the 5,948,126 precursors with a global precursor false discovery rate estimated at 0.1%.

Precursors from MassIVE-KB were filtered based on the following criteria: (1) peptide sequence lengths between 7 and 40 residues, (2) absence of modifications, and (3) at least 10 associated spectra. This rigorous filtering process resulted in a curated dataset of 183,263,674 spectra corresponding to 608,780 unique precursors. The distributions of precursor counts across sequence lengths and charge states are summarized in Appendix Figure A.1.

From this refined spectral dataset, we construct the Pep2Prob dataset as briefly described below and detailed in subsequent sections. We annotate each precursor’s associated spectra and aggregate these annotations to calculate occurrence probabilities for all possible fragment ions for each precursor. Furthermore, we implement a novel train/test splitting strategy specifically designed to prevent structurally similar peptides from appearing in both training and test sets.

#### 3.1 Mass Spectrum Annotation and Feature Representation

Mass spectrum annotation is the process of assigning one theoretically possible fragment ion to each observed peak in an  $MS^2$  spectrum. Each annotated fragment ion is a tuple defined in (2.2).

For each precursor  $p$  in the Pep2Prob dataset, the peptide sequence length ranges from  $N_p \in [7, 40]$  and the charge state from  $C \in [1, 8]$  (see Appendix Figure A.1). We now define the annotation space—the set of theoretically possible fragment ions across all precursors in the dataset. Although precursor charge states may reach up to 8, we only consider fragment ions with charges 1, 2, or 3 when constructing the annotation space, as higher-charged fragments are rare and less reliably observed in high-resolution HCD spectra. Let  $c_{\max} = 3$  and  $N_{\max} = 40$ . The annotation space  $\mathcal{F} = \{f = (t, c, n)\}$  consists of:

1. One entry for the  $a$ -ion with +1 charge, which is always included,
2.  $c_{\max}(N_{\max} - 1)$  entries for  $b$ -ions, spanning charges 1 to  $c_{\max}$  and positions  $j \in \{1, \dots, N_{\max} - 1\}$ ,
3. Similarly,  $c_{\max}(N_{\max} - 1)$  entries for  $y$ -ions with the same possible charge and position combinations as  $b$ -ions.

The annotation space therefore has a total size  $d = 235$ .

Each observed  $MS^2$  spectrum is a list of peaks, denoted as

$$S = \{(MZ_i, I_i) : 1 \leq i \leq d'\},$$

where  $d'$  is the number of peaks observed in the spectrum,  $MZ_i$  is the mass-to-charge ratio ( $m/z$ ) of the  $i$ -th peak, and  $I_i$  is its intensity. The number  $d'$  varies across spectra and is typically much smaller than  $d$ . To annotate the spectrum, we compute the theoretical  $m/z$  value for each of the  $d$  fragment ions in  $\mathcal{F}$  and match each observed peak  $S$  within a predefined tolerance window  $\delta = 0.05$ , i.e.,

$$|MZ_i - m/z| \leq \delta,$$

to the most possible fragment ion (see details in Appendix A.3). If a match is found, the intensity  $I_i$  is recorded; otherwise, the corresponding entry is set to zero. This produces a length- $d$  intensity vector for each precursor-spectrum pair. Stacking these vectors across all precursors yields a 2D matrix where: Each row corresponds to a precursor-spectrum pair; Each column corresponds to a fragment ion  $f = (t, c, n)$ ; Each entry is either a matched intensity from the spectrum or zero.

**Fragment ion mask.** While the annotation space has a fixed dimension  $d = 235$ , not all fragment ions are valid for each precursor  $p$  due to sequence length and charge state constraints. To account for this, we define a binary *ion mask*  $\pi(f, p) \in \{0, 1\}$  that indicates whether fragment ion  $f = (t, c, n)$

is theoretically possible for a given precursor  $p$  with sequence length  $N_p$  and charge state  $C$ . The entries of the mask are defined by: for  $f \in \mathcal{F}$ ,

$$\pi(f, p) \stackrel{\text{def}}{=} \begin{cases} 1, & \text{if } c \leq \min\{C, 3\} \text{ and } n < N_p, \text{ or } t = a, \\ 0, & \text{otherwise.} \end{cases} \quad (3.1)$$

This mask is applied during both training and evaluation. It ensures that predictions and loss computations are restricted to fragment ions that are chemically valid for the given precursor. This reduces the dimensionality of the output space and improves training efficiency and model interpretability.

### 3.2 Fragment Ion Probability Statistics

To estimate the statistical likelihood of fragment ion appearances, we construct the fragment ion probability dataset based on repeated  $\text{MS}^2$  measurements for the same precursor. Suppose a given precursor  $p$  is observed in  $D$  distinct  $\text{MS}^2$  spectra. Each spectrum is first annotated and normalized to form a non-negative intensity vector  $\mathbf{I}^{(i)} = \{I_f^{(i)}\}_{f \in \mathcal{F}}$ , where  $I_f^{(i)}$  is the intensity for fragment  $f$  and the  $\ell_1$  norm  $\|\mathbf{I}^{(i)}\|_1 = 1$  for all  $i \in \{1, \dots, D\}$ . We then estimate the probability that each fragment ion appears across the  $D$  spectra. We use  $\epsilon = 10^{-6}$  as the threshold for fragment presence. The empirical probability that fragment ion  $f \in \mathcal{F}$  appears is computed by

$$\mathbb{P}(f|p) = \frac{1}{D} \sum_{i=1}^D \mathbb{1}\{I_f^{(i)} > \epsilon\}. \quad (3.2)$$

If the computed value is below 0.001, we set it to zero to suppress unreliable noise. This results in an ion fragment probability vector for precursor  $p$ :  $\mathbf{P}_p = \{\mathbb{P}(f|p) : f \in \mathcal{F}\} \in [0, 1]^d$ . Stacking these probability vectors across all precursors  $p$  yields our Pep2Prob dataset.

**Learning objective.** Our goal is to train machine learning models that take a precursor  $p$  as input and predict the corresponding ion probability vector  $\mathbf{P}_p$ . Denote the model prediction for precursor  $p$  by

$$\hat{\mathbf{P}}_p = \{\hat{\mathbb{P}}(f|p) : f \in \mathcal{F}\} \in [0, 1]^d.$$

During training and evaluation, the ion mask  $\pi$  defined by (3.1) is applied to both the predicted and ground-truth probability vectors to restrict computations to valid fragment ion dimensions.

### 3.3 Train and Test Split

Many precursors in our dataset share common sequence patterns, particularly long prefixes or suffixes — e.g., “ABCDEFGHJK”, “ADCDEFGHIJK”, “ABCCDEFGHIJK”, and “EFGHIJK”. It is reasonable to *not* split such similar precursors into train and test datasets separately, otherwise it may lead to data leakage [16] and overoptimistic evaluation [26, 14, 30] due to similar sequences appearing in both training and test sets. If similar sequences are split across the training and test sets, the model may easily generalize to the test examples by memorizing patterns seen during training.

We construct an undirected graph where each node represents a precursor. An edge is added between two nodes if their peptide sequences satisfy: **Identical** — they are the same; **SharePrefix** or **ShareSuffix** — they share a common prefix or suffix of length 6. If none of these conditions are met, the pair is labeled as having **NoConnection**, and no edge is added. Given that the minimum peptide sequence length is 7 (Figure A.1 in the Appendix), this criterion effectively captures meaningful local similarities without being overly inclusive.

The resulting graph decomposes into disconnected components, each representing a group of similar sequences. To create train/test splits, we sort these components by size and distribute them into five balanced folds using a greedy strategy: each new component is assigned to the currently smallest fold. One fold is used for testing, and the remaining four for training. This component-based split ensures no shared local motifs between training and test sets, leading to more realistic and robust evaluation of the model’s generalization ability.

## 4 Benchmarking Fragment Ion Probability Prediction

### 4.1 Baseline Models and Experimental Setup

We benchmark a set of methods on the Pep2Prob dataset with two goals: (1) to establish baseline performance metrics; (2) to determine how strongly the fragment-ion probability  $\mathbb{P}(f|p)$  depends on  $p$  and how complex that dependency is, thereby gauging the model capacity needed for accurate fragment-ion prediction.

We first apply two statistical-rule-based approaches: one global method that ignores precursor information entirely and one sequence-aware method that partially incorporates precursor context. Additionally, we implement three machine learning baselines—linear regression, neural networks, and transformers—with varying capacities to model complex relationships. Each ML model incorporates the fragment-ion mask  $\pi(f, p)$  in (3.1) and is trained to minimize L1 loss defined as:

$$\ell_1(\mathbf{P}_p, \hat{\mathbf{P}}_p) \stackrel{\text{def}}{=} \frac{\sum_{f \in \mathcal{F}} |\mathbb{P}(f|p) - \hat{\mathbb{P}}(f|p)| \cdot \pi(f, p)}{\sum_{f \in \mathcal{F}} \pi(f, p)}. \quad (4.1)$$

**Global Modeling (Global).** Our initial baseline computes fragment ion probabilities solely based on ion type  $t$  and charge  $c$  information, excluding  $n$  and  $p$ . This predictor assumes  $\mathbb{P}(f|p)$  is independent of the peptide sequence and the position number [4]. It is widely used in current peptide identification methods via database search and de novo sequencing, such as MSGF+ [17] and Bandeira et al. [2].

The predictor is built upon global statistics aggregated across precursors in the training set. For each unique fragment ion  $f$ , the model estimates:

$$\hat{\mathbb{P}}_{\text{Global}}(f = (t, c, n)) = \frac{\sum_{p \in \mathcal{D}_{\text{train}}} \sum_{f'=(t', c', n')} \pi(f', p) \cdot u(p) \cdot \mathbb{1}(t = t', c = c') \cdot \mathbb{P}(f'|p)}{\sum_{p \in \mathcal{D}_{\text{train}}} \sum_{f'=(t', c', n')} \pi(f', p) \cdot u(p) \cdot \mathbb{1}(t = t', c = c')} \quad (4.2)$$

where  $u(p)$  is the number of spectra associated with precursor  $p$ , and  $\pi(f, p)$  is the indicator function that equals 1 if precursor  $p$  can theoretically produce fragment ion  $f$ , and 0 otherwise.

**Bag-of-Fragment-Ion (BoF).** This predictor extends global modeling by incorporating minimal precursor information. It computes fragment ion probabilities based on both the fragment  $f$  and the fragment’s amino acid sequence, which is determined jointly by  $f$  (specifically its ion type and position number) and  $p$ . We denote this amino acid sequence as  $\xi(f, p)$ . For example, if  $p$ ’s peptide sequence is ‘ABCDE’ and  $f = (b, 1+, 2)$ , then  $\xi(f, p) = \text{‘AB’}$ ; if  $f = (y, 1+, 3)$ , then  $\xi(f, p) = \text{‘CDE’}$ .

The model’s prediction rule is:

$$\hat{\mathbb{P}}_{\text{BoF}}(f|p) = \frac{\sum_{p' \in \mathcal{D}_{\text{train}}} \pi(f, p') \cdot u(p') \cdot \mathbb{1}(\xi(f, p) = \xi(f, p')) \cdot \mathbb{P}(f|p')}{\sum_{p' \in \mathcal{D}_{\text{train}}} \pi(f, p') \cdot u(p') \cdot \mathbb{1}(\xi(f, p) = \xi(f, p'))} \quad (4.3)$$

This estimator conditions on fragments having identical amino acid sequences, capturing sequence-specific fragmentation patterns while remaining computationally tractable.

**Linear regression model (LR).** To establish the linear regression baseline, we need to first encode each precursor  $p$  into a fixed-length one-hot vector  $\mathbf{x}_p$ :

$$\mathbf{x}_p = \mathbf{x}_c \oplus \mathbf{x}_{\text{seq}} \quad (4.4)$$

which concatenates two components: a one-hot encoding of the charge state  $\mathbf{x}_c$  and a one-hot encoding of the peptide sequence  $\mathbf{x}_{\text{seq}}$ , where the latter is formed by concatenating one-hot vectors for each amino acid position. Zero padding is applied to ensure uniform vector length across all precursors. We train an independent linear regressor for each fragment ion  $f$ :

$$\hat{\mathbb{P}}_{\text{Linear}}(f|p) = \mathbf{w}_f^T \mathbf{x}_p + b_f \quad (4.5)$$

Note that when training the linear regression baseline model, the loss is evaluated on *all*  $(f, p)$  pairs, including invalid ones.  $\mathbb{P}(f|p)$  is taken to be  $-1$  on these pairs.

**Residual neural network (ResNet).** The neural network baseline extends the linear model with nonlinear transformations and deeper architecture. Using the same one-hot encoding  $\mathbf{x}_p$  as input, we



train a feedforward network predicting  $\mathbb{P}(f|p)$  for all  $f$ 's simultaneously. The architecture consists of four fully-connected layers with residual connections [13]:

$$\mathbf{h}_1 = \text{ReLU}(W_1 \mathbf{x}_p + \mathbf{b}_1) \quad (4.6)$$

$$\mathbf{h}_2 = \text{ReLU}(W_2 \mathbf{h}_1 + \mathbf{b}_2)/2 + \mathbf{h}_1/2 \quad (4.7)$$

$$\mathbf{h}_3 = \text{ReLU}(W_3 \mathbf{h}_2 + \mathbf{b}_3)/2 + \mathbf{h}_2/2 \quad (4.8)$$

$$\hat{\mathbb{P}}_{\text{NN}}(f|p) = [\text{sigmoid}(\mathbf{W}_4 \mathbf{h}_3 + \mathbf{b}_4)]_f \quad (4.9)$$

Dropout with rate 0.15 is applied after activations during training for regularization. The model minimizes the L1 loss over valid fragment-precursor pairs  $(f, p)$ , i.e., pairs with  $\pi(f, p) = 1$ .

**Transformer.** The Transformer baseline replaces the ResNet with a stack of self-attention layers to capture long-range dependencies in the precursor input. Using the same token-encoding of the precursor  $p$  (amino-acid tokens plus charge tokens) followed by a block of padding of length  $d$  (the number of fragments to predict), we train a decoder-only Transformer [36] to output  $\hat{\mathbb{P}}_{\text{TF}}(f | p)$  for all fragment indices  $f \in \mathcal{F}$  in one shot. Let  $m = N_p + d$  be the combined sequence length, where  $N_p$  is the length of the peptide sequence. The model is defined by

$$\mathbf{X}^{(0)} = \text{Embed}(p) + P \in \mathbb{R}^{m \times d_0}, \quad (4.10)$$

$$\mathbf{X}^{(\ell)} = \text{DecoderLayer}(\mathbf{X}^{(\ell-1)}, \text{mask} = \text{subsequentMask}(m)), \quad \ell = 1, \dots, L, \quad (4.11)$$

$$\hat{\mathbb{P}}_{\text{TF}}(f | p) = [\text{sigmoid}(\mathbf{W}_o \mathbf{X}^{(L)} + b_o)]_f, \quad (4.12)$$

where  $P$  is a positional embedding and each DecoderLayer is a standard transformer decoder layer, consisting of multi-head self-attention with  $H$  heads, feed-forward network of width  $d_{\text{ff}}$ , residual connections, and layer normalization at each sub-layer. We apply a dropout rate of 0.2 inside every attention and feed-forward block. Finally, a linear output head  $\mathbf{W}_o \in \mathbb{R}^{1 \times d}$  followed by a sigmoid produces per-position probabilities.

**Experimental configuration.** The linear regression model is optimized using SGDRegressor from scikit-learn, due to the large amount of training samples. Here, the loss is set to 'epsilon\_insensitive' with 'epsilon=0' to mimic L1 loss. For ResNet, the hidden layer size is 512. ResNet is trained for 200 epochs with a batch size of 512, and optimized using AdamW with a learning rate of  $10^{-3}$ . For the transformer model, we set  $d_0 = 180$ ,  $H = 4$ ,  $L = 4$ ,  $d_{\text{ff}} = 512$ . All trainings minimize the same masked L1 loss over valid fragment positions as before. Code for reproducing benchmark experiments is available at <https://github.com/Bandeira-Lab/pep2prob-benchmark>.

**Computational resources.** The transformer benchmark model is trained on a node with 4 NVIDIA A100 GPUs and 256 GiB memory for 4 hours per 100 epochs. All 4 other benchmark experiments are performed on a machine with Intel® Core™ i9-13900K  $\times$  32 CPUs and 128 GiB memory. Each benchmark experiment takes up to 12 minutes, including both training and evaluation.

**Threshold for model outputs.** Consistent with our approach in constructing the Pep2Prob dataset, we apply a threshold to the model outputs during evaluation. Specifically, we set prediction values below 0.001 to zero, which maintains ion fragment probability at 99% precision. More generally, an optimal threshold  $\tau$  could be determined for each precursor to recover the support of ion probability with desired precision-recall characteristics.

## 4.2 Evaluation Pipeline

**Type 1: Norm-based metrics for probability values.** We first use norm-based metrics for comparing predicted and true fragment ion probability vectors. Common choices include mean squared error (MSE), which penalizes large deviations; L1 loss, which is more robust to outliers; and normalized spectral angle (SA) [33], a scale-invariant metric measuring angular similarity. Higher SA values indicate better alignment between predicted and true probability vectors.

**Type 2: Support-recovery metrics for fragment ion existence.** To evaluate how well the predicted ion fragment probabilities capture true fragment presence, we compare the supports of the true and predicted probability vectors. For each precursor, we consider an ion as present if its true probability is nonzero and as predicted present if its predicted probability exceeds a threshold  $\tau$ . Given the precursor, we consider the support of the true probability vector  $\mathbf{P}_p$  and the predicted probability vector  $\hat{\mathbf{P}}_p$  as  $S = \{f \in \mathcal{F} : \mathbb{P}(f|p) > 0\}$ ,  $\hat{S} = \{f \in \mathcal{F} : \hat{\mathbb{P}}(f|p) > \tau\}$ , where we use  $\tau = 0.001$ .

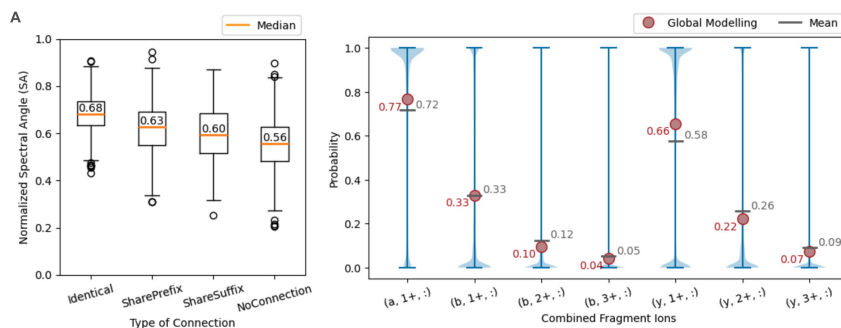


Figure 2: (A) The box plot of the normalized spectral angle (SA) based on the intersected fragment ion probabilities between precursors paired by **Identical**, **SharePrefix**, **ShareSuffix**, and **NoConnection**, defined in Section 3.3. (B) The probability distribution of fragment ions combined for all peptide sequences and fragmentation positions, is shown for Pep2Prob in blue (mean value in gray) and for Global Modeling in red.

Based on this, we compute standard classification metrics between  $S$  and  $\hat{S}$ : **accuracy** (the fraction of correct predictions), **sensitivity** (the proportion of true fragments correctly identified), and **specificity** (the proportion of non-fragments correctly excluded).

Each evaluation metric is computed at two levels: Precursor level, where we calculate the metric for each precursor and then average over all precursors, yielding an overall score per precursor; Fragment ion level, where we compute the metric for each individual ion and average over all ions in the dataset, providing an score per ion. The complete definitions of all the metrics are in Appendices C.1 and C.2.

## 5 Results and Analysis

**Data split based on sequence patterns.** Following the proposed training-test split in Section 3.3, we found that 339,102 precursors (55.7%) share identical peptide sequences and are therefore connected. Additionally, 429,863 precursors (70.6%) and 480,023 precursors (78.8%) share prefixes and suffixes of length 6 with others, respectively. Finally, we identified 204,716 isolated graph components, which are divided into 5 sets, each containing approximately 132,259 precursors from a total of around 58,000 precursors. Figure 2(A) shows that two precursors sharing common sequence patterns tend to have similar probabilities of the fragment ion occurring in both precursors, which validates the motivation for our training-test split strategy in Section 3.3.

**Distributional comparison on fragment ion probabilities.** Furthermore, Figure 2(B) illustrates the distribution of fragment ion probabilities across different ion types and charges in our dataset. We compare the Global Modeling approach (in red) to the empirical distribution of Pep2Prob (in blue). Notably, Global Modeling tends to overestimate or underestimate the probabilities of certain ion types. These discrepancies underscore the necessity of incorporating precursor-specific information to improve the accuracy of fragment ion probability estimation.

**Baseline comparison on fragment ion probability.** The left column of Table 1 shows Type 1 evaluations in Section 4.2. We see a clear hierarchy: the Global baseline yields the highest error and lowest SA similarity, indicating its inability to capture sequence-specific fragmentation patterns. BoF and LR progressively reduce prediction error, but only modestly. Deep architectures, ResNet and self-attention-based transformer, have higher model capacity, yielding lower error and higher SA for fragment ion probabilities. This progression underscores that the relation between fragmentation probabilities and precursor is complex and requires high-capacity ML models to learn effectively.

**Baseline comparison on fragment ion existence.** The right column of Table 1 evaluates the models' predictions on the existence of fragment ions (Type 2 evaluation). The global model achieves maximal Sensitivity at the expense of Specificity because it always assumes all theoretically possible fragment ions exist. BoF and LR already achieve high specificity and sensitivity, respectively. The ResNet further refines this balance by learning local sequence contexts, improving both true-positive and true-negative rates. Transformer achieves the highest overall accuracy and specificity, albeit with a modest drop in sensitivity. These results illustrate that sophisticated ML models are essential not only for precise probability estimates but also for reliable fragment-existence predictions.



Table 1: Model performance comparison on fragmentation ion probability prediction at the precursor level. Best results are in **bold**, second-best ones are underlined. Analogous trends hold for fragment ion level in Tables D.1 and D.2 in the Appendix.

Model	Type 1 Evaluation for Probability Values			Type 2 Evaluation for fragment ion existence		
	L1	MSE	SA	Acc	Sen	Spec
Global	0.2437 $\pm$ 0.0002	0.0994 $\pm$ 0.0002	0.5578 $\pm$ 0.0004	0.6993 $\pm$ 0.0007	<b>1.0000 <math>\pm</math> 0.0000</b>	0.0000 $\pm$ 0.0000
BoF	0.1788 $\pm$ 0.0001	0.1184 $\pm$ 0.0001	0.5086 $\pm$ 0.0006	0.8027 $\pm$ 0.0008	0.4435 $\pm$ 0.0009	0.7683 $\pm$ 0.0005
LR	0.1258 $\pm$ 0.0002	0.0540 $\pm$ 0.0002	0.6951 $\pm$ 0.0004	0.7661 $\pm$ 0.0053	0.9213 $\pm$ 0.0021	0.3771 $\pm$ 0.0286
ResNet	0.0687 $\pm$ 0.0002	0.0213 $\pm$ 0.0000	0.8182 $\pm$ 0.0003	0.8708 $\pm$ 0.0017	0.8766 $\pm$ 0.0026	0.7150 $\pm$ 0.0038
Transformer	<b>0.0560 <math>\pm</math> 0.0005</b>	<b>0.0165 <math>\pm</math> 0.0003</b>	<b>0.8453 <math>\pm</math> 0.0016</b>	<b>0.9530 <math>\pm</math> 0.0020</b>	0.7963 $\pm$ 0.0055	<b>0.9215 <math>\pm</math> 0.0039</b>

## 6 Related Work

**ML for MS<sup>2</sup>-based Proteomics.** The application of machine learning to MS<sup>2</sup> data for proteomic studies has evolved rapidly in recent years, with significant advances in (1) predicting peptide sequence from MS<sup>2</sup> spectra: de novo peptide sequencing [24, 34, 41, 42] and (2) predicting properties in MS<sup>2</sup> of peptide sequence: fragment ion intensity prediction [8, 12, 28, 43], retention time prediction [3, 12, 43], and post-translational modification site prediction [23]. Researchers have explored a diverse range of machine learning methods for these tasks, from gradient tree boosting algorithms [5, 6, 11] to more sophisticated models, including convolutional neural networks in [19], recurrent neural networks in [31], transformer architectures in [8, 41], and few-shot learning frameworks in [29]. Recent advances have further enhanced performance by incorporating embeddings from protein language models [15, 25, 39]. However, fragment ion probability prediction still relies on simple global modeling [4] that ignores important peptide-specific fragmentation patterns but widely used in popular peptide identification pipelines, such as MSGF+ [17].

**MS<sup>2</sup> Datasets for ML in Proteomics.** While public mass spectrometry (MS) data repositories like PRIDE [22] and MassIVE [20] host vast MS<sup>2</sup> data collections, they present challenges for applying machine learning directly due to variable quality and unidentified spectra. More structured resources have emerged for specific tasks recently, e.g., the nine-species dataset proposed in [34] for de novo sequencing and the PROSPECT series datasets [10, 28] for predicting fragment ion intensity and retention time. The construction of these resources relies on the identification of mass spectra in public MS repositories. For instance, MassIVE-KB [38] addresses quality concerns by consolidating the best-representative spectra for each precursor across multiple experiments, thereby enhancing the reliability of model training [42].

## 7 Conclusion and Limitations

In summary, we have presented Pep2Prob, the first comprehensive dataset and benchmark designed specifically for peptide-specific fragment ion probability prediction in MS<sup>2</sup>-based proteomics. Our benchmark experiments demonstrate two key findings: first, incorporating peptide sequence information significantly improves prediction accuracy compared to global approaches; second, the relationship between peptide sequences and fragmentation patterns exhibits intricate nonlinearities requiring sophisticated ML approaches to efficiently learn complex features.

One limitation of Pep2Prob is that it only captures marginal distributions of fragment ions conditioned on precursors, without modeling correlations between different ions’ occurrences. Incorporating these correlations could further improve fragmentation modeling, nevertheless constructing a dataset for this purpose would require parsing a large number of spectra for each precursor, significantly increasing computational and storage requirements.

Beyond statistical modeling, Pep2Prob has two additional limitations regarding data source diversity. First, it excludes post-translational modifications (PTMs), which significantly alter fragmentation patterns, due to challenges in confidently identifying modified peptides at scale. Second, it contains only higher-energy collisional dissociation (HCD) spectra from Orbitrap instruments, not covering alternative fragmentation methods (ETD, CID) or different instrument platforms. Future work should address these limitations by incorporating modified peptides and expanding coverage across diverse fragmentation methods and instrument types, thereby enhancing the practical utility of fragment ion prediction in proteomics workflows.

## 8 Acknowledgements

N.B., P.W., H.X. partial funding from National Institutes of Health (grant GM148372). S.S. was supported by the SITP postdoctoral fellowship at Stanford University. We thank Jeremy Carver and Sijie Zhu for helpful discussions during the project development.

## References

- [1] ASLAM, B., BASIT, M., NISAR, M. A., KHURSHID, M., AND RASOOL, M. H. Proteomics: technologies and their applications. Journal of chromatographic science (2016), 1–15.
- [2] BANDEIRA, N., OLSEN, J. V., MANN, M., AND PEVZNER, P. A. Multi-spectra peptide sequencing and its applications to multistage mass spectrometry. Bioinformatics 24, 13 (2008), i416–i423.
- [3] BOUWMEESTER, R., GABRIELS, R., HULSTAERT, N., MARTENS, L., AND DEGROEVE, S. DeepLC can predict retention times for peptides that carry as-yet unseen modifications. Nature methods 18, 11 (2021), 1363–1369.
- [4] DANČÍK, V., ADDONA, T. A., CLAUSER, K. R., VATH, J. E., AND PEVZNER, P. A. De novo peptide sequencing via tandem mass spectrometry. Journal of computational biology 6, 3–4 (1999), 327–342.
- [5] DEGROEVE, S., MADDELEIN, D., AND MARTENS, L. Ms2pip prediction server: compute and visualize ms2 peak intensity predictions for cid and hcd fragmentation. Nucleic acids research 43, W1 (2015), W326–W330.
- [6] DEGROEVE, S., AND MARTENS, L. Ms2pip: a tool for ms/ms peak intensity prediction. Bioinformatics 29, 24 (2013), 3199–3203.
- [7] DORL, S., WINKLER, S., MECHTLER, K., AND DORFER, V. Ms ana: Improving sensitivity in peptide identification with spectral library search. Journal of Proteome Research 22, 2 (2023), 462–470.
- [8] EKVALL, M., TRUONG, P., GABRIEL, W., WILHELM, M., AND KALL, L. Prosit transformer: a transformer for prediction of ms2 spectrum intensities. Journal of Proteome Research 21, 5 (2022), 1359–1364.
- [9] FRANK, A., AND PEVZNER, P. Pepnovo: de novo peptide sequencing via probabilistic network modeling. Analytical chemistry 77, 4 (2005), 964–973.
- [10] GABRIEL, W., SHOUMAN, O., SCHRÖDER, E. A., BÖSSL, F., AND WILHELM, M. Prospect ptms: Rich labeled tandem mass spectrometry dataset of modified peptides for machine learning in proteomics. Advances in Neural Information Processing Systems 37 (2024), 131154–131196.
- [11] GABRIELS, R., MARTENS, L., AND DEGROEVE, S. Updated ms<sup>2</sup>pip web server delivers fast and accurate ms<sup>2</sup> peak intensity prediction for multiple fragmentation methods, instruments and labeling techniques. Nucleic acids research 47, W1 (2019), W295–W299.
- [12] GESSULAT, S., SCHMIDT, T., ZOLG, D. P., SAMARAS, P., SCHNATBAUM, K., ZERWECK, J., KNAUTE, T., RECHENBERGER, J., DELANGHE, B., HUHMER, A., ET AL. Prosit: proteome-wide prediction of peptide tandem mass spectra by deep learning. Nature methods 16, 6 (2019), 509–518.
- [13] HE, K., ZHANG, X., REN, S., AND SUN, J. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (2016), pp. 770–778.
- [14] HOBBOHM, U., SCHARF, M., SCHNEIDER, R., AND SANDER, C. Selection of representative protein data sets. Protein Science 1, 3 (1992), 409–417.
- [15] HOU, Z., YANG, Y., MA, Z., WONG, K.-C., AND LI, X. Learning the protein language of proteome-wide protein-protein binding sites via explainable ensemble deep learning. Communications Biology 6, 1 (2023), 73.
- [16] JOERES, R., BLUMENTHAL, D. B., AND KALININA, O. V. Data splitting to avoid information leakage with datasail. Nature Communications 16, 1 (2025), 3337.

- [17] KIM, S., AND PEVZNER, P. A. Ms-gf+ makes progress towards a universal database search tool for proteomics. *Nature communications* 5, 1 (2014), 5277.
- [18] KONG, A. T., LEPREVOST, F. V., AVTONOMOV, D. M., MELLACHERUVU, D., AND NESVIZHSKII, A. I. Msfragger: ultrafast and comprehensive peptide identification in mass spectrometry-based proteomics. *Nature methods* 14, 5 (2017), 513–520.
- [19] LIU, K., LI, S., WANG, L., YE, Y., AND TANG, H. Full-spectrum prediction of peptides tandem mass spectra using deep neural network. *Analytical chemistry* 92, 6 (2020), 4275–4283.
- [20] MASSIVE. Mass spectrometry interactive virtual environment. Available at: <http://massive.ucsd.edu/>, 2025. Accessed: April 16, 2025.
- [21] PAN, S., AEBERSOLD, R., CHEN, R., RUSH, J., GOODLETT, D. R., MCINTOSH, M. W., ZHANG, J., AND BRENTNALL, T. A. Mass spectrometry based targeted protein quantification: methods and applications. *Journal of proteome research* 8, 2 (2009), 787–797.
- [22] PEREZ-RIVEROL, Y., BANDLA, C., KUNDU, D. J., KAMATCHINATHAN, S., BAI, J., HEWAPATHIRANA, S., JOHN, N. S., PRAKASH, A., WALZER, M., WANG, S., ET AL. The pride database at 20 years: 2025 update. *Nucleic Acids Research* 53, D1 (2025), D543–D553.
- [23] POKHAREL, S., PRATYUSH, P., HEINZINGER, M., NEWMAN, R. H., AND KC, D. B. Improving protein succinylation sites prediction using embeddings from protein language model. *Scientific reports* 12, 1 (2022), 16933.
- [24] QIAO, R., TRAN, N. H., XIN, L., CHEN, X., LI, M., SHAN, B., AND GHODSI, A. Computationally instrument-resolution-independent de novo peptide sequencing for high-resolution devices. *Nature Machine Intelligence* 3, 5 (2021), 420–425.
- [25] RAO, R., BHATTACHARYA, N., THOMAS, N., DUAN, Y., CHEN, X., CANNY, J., ABBEEL, P., AND SONG, Y. S. Evaluating protein transfer learning with tape. In *Advances in Neural Information Processing Systems* (2019).
- [26] SANDER, C., AND SCHNEIDER, R. Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins: Structure, Function, and Bioinformatics* 9, 1 (1991), 56–68.
- [27] SHAO, W., ZHU, K., AND LAM, H. Refining similarity scoring to enable decoy-free validation in spectral library searching. *Proteomics* 13, 22 (2013), 3273–3283.
- [28] SHOUMAN, O., GABRIEL, W., GIURCOIU, V.-G., STERNLICHT, V., AND WILHELM, M. Prospect: Labeled tandem mass spectrometry dataset for machine learning in proteomics. *Advances in Neural Information Processing Systems* 35 (2022), 32882–32896.
- [29] TARN, C., AND ZENG, W.-F. pdeep3: toward more accurate spectrum prediction with fast few-shot learning. *Analytical Chemistry* 93, 14 (2021), 5815–5822.
- [30] TEUFEL, F., GÍSLASON, M. H., ALMAGRO ARMENTEROS, J. J., JOHANSEN, A. R., WINTHER, O., AND NIELSEN, H. Graphpart: homology partitioning for biological sequence analysis. *NAR genomics and bioinformatics* 5, 4 (2023), lqad088.
- [31] TIWARY, S., LEVY, R., GUTENBRUNNER, P., SALINAS SOTO, F., PALANIAPPAN, K. K., DEMING, L., BERNDL, M., BRANT, A., CIMERMANCIC, P., AND COX, J. High-quality ms/ms spectrum prediction for data-dependent and data-independent acquisition data analysis. *Nature methods* 16, 6 (2019), 519–525.
- [32] TOPOL, E. J. The revolution in high-throughput proteomics and ai. *Science* 385, 6716 (2024), eads5749.
- [33] TOPRAK, U. H., GILLET, L. C., MAIOLICA, A., NAVARRO, P., LEITNER, A., AND AEBERSOLD, R. Conserved peptide fragmentation as a benchmarking tool for mass spectrometers and a discriminating feature for targeted proteomics. *Molecular & Cellular Proteomics* 13, 8 (2014), 2056–2071.
- [34] TRAN, N. H., ZHANG, X., XIN, L., SHAN, B., AND LI, M. De novo peptide sequencing by deep learning. *Proceedings of the National Academy of Sciences* 114, 31 (2017), 8247–8252.
- [35] TYANOVA, S., TEMU, T., AND COX, J. The maxquant computational platform for mass spectrometry-based shotgun proteomics. *Nature protocols* 11, 12 (2016), 2301–2319.

- [36] VASWANI, A., SHAZEER, N., PARMAR, N., USZKOREIT, J., JONES, L., GOMEZ, A. N., KAISER, Ł., AND POLOSUKHIN, I. Attention is all you need. Advances in neural information processing systems 30 (2017).
- [37] WANG, J., PÉREZ-SANTIAGO, J., KATZ, J. E., MALLICK, P., AND BANDEIRA, N. Peptide identification from mixture tandem mass spectra. Molecular & Cellular Proteomics 9, 7 (2010), 1476–1485.
- [38] WANG, M., WANG, J., CARVER, J., PULLMAN, B. S., CHA, S. W., AND BANDEIRA, N. Assembling the Community-Scale Discoverable Human Proteome. Cell Systems 7, 4 (Oct. 2018), 412–421.e5.
- [39] WEISSENOW, K., HEINZINGER, M., AND ROST, B. Protein language-model embeddings for fast, accurate, and alignment-free protein structure prediction. Structure 30, 8 (2022), 1169–1177.
- [40] WITZE, E. S., OLD, W. M., RESING, K. A., AND AHN, N. G. Mapping protein post-translational modifications with mass spectrometry. Nature methods 4, 10 (2007), 798–806.
- [41] YILMAZ, M., FONDRIE, W., BITTREMIEUX, W., OH, S., AND NOBLE, W. S. De novo mass spectrometry peptide sequencing with a transformer model. In International Conference on Machine Learning (2022), PMLR, pp. 25514–25522.
- [42] YILMAZ, M., FONDRIE, W. E., BITTREMIEUX, W., MELENDEZ, C. F., NELSON, R., ANANTH, V., OH, S., AND NOBLE, W. S. Sequence-to-sequence translation from mass spectra to peptides with a transformer model. Nature communications 15, 1 (2024), 6427.
- [43] ZENG, W.-F., ZHOU, X.-X., WILLEMS, S., AMMAR, C., WAHLE, M., BLUDAU, I., VOYTIK, E., STRAUSS, M. T., AND MANN, M. Alphapeptdeep: a modular deep learning framework to predict peptide properties for proteomics. Nature Communications 13, 1 (2022), 7238.

# Technical Appendices and Supplementary Material

## A Dataset

### A.1 Precursor

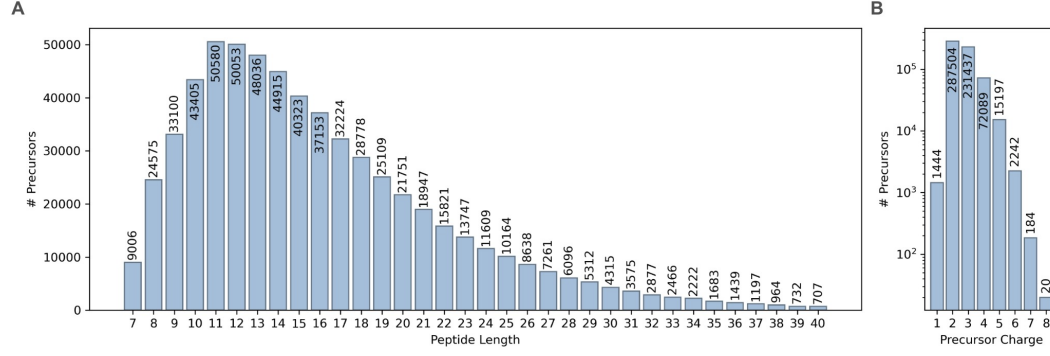


Figure A.1: The distribution of precursor counts across (A) sequence lengths and (B) charge state.

### A.2 Fragment Ions

In Sec 3.1, we define an annotation space of 235 fragment ions. Each fragment ion is a triple of (ion type, charge, position number). Here is the list of all considered fragment ions:

- 1 a-ion:  $[(a', 1, 2)]$
- 117 b-ions:  $[(b', c, n) \forall c \in [1, 2, 3], n \in [1, 2, \dots, 39]]$
- 117 y-ions:  $[(y', c, n) \forall c \in [1, 2, 3], n \in [1, 2, \dots, 39]]$

### A.3 Details for Spectrum Annotation

When annotating an MS<sup>2</sup> mass spectrum according to its identified precursor, i.e., (peptide sequence, charge), the  $m/z$  values of all possible fragment ions are calculated, where the fragment charges cannot exceed the precursor charge and the position numbers are bounded by the length of the peptide.

The  $m/z$  value  $mz$  for a fragment ion with type  $t$ , charge state  $c$ , and position number  $n$  from precursor peptide sequence  $S$ :

$$mz = \frac{M_{fragment} + c \cdot M_{proton}}{c},$$

where  $M_{fragment}$  is the neutral mass of the fragment ion and  $M_{proton} = 1.0073$  Da. The neutral fragment mass depends on the ion types:

- For prefix (N-terminal) ions, a- and b-ions:

$$M_{fragment} = \sum_{i=1}^p M_{AA_i} + M_{offset}^t;$$

- For suffix (C-terminal) ions, y-ions:

$$M_{fragment} = \sum_{i=p+1}^L M_{AA_i} + M_{offset}^t,$$

where  $L$  is the peptide length,  $M_{AA_i}$  is the mass of amino acid at position  $i$ , and  $M_{offset}^t$  is the ion type specific mass offset. Tables A.1 and A.2 show the amino acid masses  $M_{AA}$  and ion type mass offsets  $M_{offset}$ .

Table A.1: Monoisotopic masses of amino acids

Amino Acid	Single Letter	Mass (Da)
Alanine	A	71.03711378471
Arginine	R	156.10111102359997
Asparagine	N	114.04292744114001
Aspartic acid	D	115.02694302383001
Cysteine	C	103.00918478471
Glutamic acid	E	129.04259308796998
Glutamine	Q	128.05857750527997
Glycine	G	57.02146372057
Histidine	H	137.05891185845002
Isoleucine	I	113.08406397713001
Leucine	L	113.08406397713001
Lysine	K	128.09496301399997
Methionine	M	131.04048491299
Phenylalanine	F	147.06841391298997
Proline	P	97.05276384885
Serine	S	87.03202840427001
Threonine	T	101.04767846841
Tryptophan	W	186.07931294985997
Tyrosine	Y	163.06332853254997
Valine	V	99.06841391299

Table A.2: Ion type mass offsets for fragment ion calculation

Ion Type	Mass Offset (Da)	Description
b	1.0073	Addition of hydrogen ion ( $H$ )
a	-26.9876	Loss of $CO$ and Addition of $H$
y	19.0178	Addition of $H_2O + H$

Peak annotation employs a greedy assignment strategy. After computing the theoretical  $m/z$  values for all applicable fragment ions, they are matched to experimental peaks using a 0.05 Da tolerance. Assignment uses a greedy strategy: for each peak in the spectrum, we identify candidate fragment ions within a 0.05 Da mass tolerance, then assign the fragment ion with the highest probability from the global modeling among unassigned candidates. The order of fragment ions given by the global modeling is:  $(y, 1, :) > (b, 1, :) > (y, 2, :) > (a, 1, 2) > (b, 2, :) > (y, 3) > (b, 3)$ . Peaks without valid candidates remain unannotated, and we enforce a one-to-one correspondence between peaks and fragment ions.

## B Additional Experimental Details

### Training Details for Transformer.

- **Data loading:** Batch size  $B = 1024$ , shuffle for training, `num_workers=4` (train) / 1 (evaluation).
- **Optimizer:** AdamW with initial learning rate  $1 \times 10^{-3}$  and weight decay  $1 \times 10^{-2}$ .
- **Scheduler:** ReduceLROnPlateau (factor 0.2, patience 5 epochs, minimum LR  $1 \times 10^{-6}$ ), stepped on validation L1 loss.
- **Loss:** Masked L1 loss over valid fragment positions, see (4.1). where  $S_b$  indexes fragments with valid (theoretically possible) ions in sample  $b$ .
- **Training:** 100 epochs on NVIDIA GPUs (4xA100).



## C Evaluation Details

### C.1 Norm-based Metrics

Here, we provide a concise summary of loss functions and evaluation metrics for the fragment ion probability vector  $\mathbf{P}_p$  for precursor  $p$ . Our primary focus will be on norm-based metrics, although alternative metrics, such as divergence-based metrics and binary cross-entropy, are also available. Norm-based metrics are simple, differentiable, and widely used when the magnitude of error matters uniformly.

**Norm-based Metrics** Except for L1 loss defined in (4.1), we also consider the following metrics between  $\hat{\mathbf{P}}_p$  and  $\mathbf{P}_p$ :

$$\begin{aligned} \text{MSE} &= \frac{\sum_{f \in \mathcal{F}} \pi(f, p) (\hat{\mathbb{P}}(f|p) - \mathbb{P}(f|p))^2}{\sum_{f \in \mathcal{F}} \pi(f, p)}, \\ \text{SA} &= 1 - \frac{2}{\pi} \arccos \frac{\langle \mathbf{P}_p, \hat{\mathbf{P}}_p \rangle}{\max\{\|\mathbf{P}_p\|_2 \|\hat{\mathbf{P}}_p\|_2, \epsilon\}}. \end{aligned} \quad (\text{C.1})$$

In this scenario, the value of SA is restricted to the range  $[-1, 1]$ . Consequently, the larger the value of SA, the more favorable alignment exists between the predicted and the target probability vectors.

### C.2 Support-Recovery Metrics

First, we want to know how many ion fragment probabilities are not zero, in which case we identify the nonzero coordinates in our predictions. For each precursor, we define the support of the true probability vector  $\mathbf{P}_p$  and the predicted probability vector  $\hat{\mathbf{P}}_p$  as

$$S = \{f \in \mathcal{F} : \mathbb{P}(f|p) > 0\}, \hat{S} = \{f \in \mathcal{F} : \hat{\mathbb{P}}(f|p) > \tau\},$$

where  $\tau > 0$  is the threshold. In our experiments, we take  $\tau = 0.001$ . Then, the standard evaluation metrics for evaluating the existence of fragment ions are defined as follows:

$$\begin{aligned} \text{Accuracy} &= \frac{|S \cap \hat{S}| + |([d] \setminus S) \setminus \hat{S}|}{d}, & \text{fraction of correct predictions;} \\ \text{Sensitivity} &= \frac{|S \cap \hat{S}|}{|\hat{S}|}, & \text{fraction of true positives that are recovered;} \\ \text{Specificity} &= \frac{|([d] \setminus S) \setminus \hat{S}|}{|[d] \setminus S|}, & \text{fraction of true negatives that are correctly predicted.} \end{aligned}$$

Here:

- $S \cap \hat{S}$  = true positives (TP),
- $\hat{S} \setminus S$  = false positives (FP),
- $S \setminus \hat{S}$  = false negatives (FN),
- $([d] \setminus S) \setminus \hat{S}$  = true negatives (TN).

### C.3 (Optional) Regression Confusion Matrix.

Let  $\mathbb{P}(f|p), \hat{\mathbb{P}}(f|p) \in [0, 1]$ , be the true and predicted values at fragment ion  $f$  for  $f \in \mathcal{F}$ . We extend the support recovery metrics to the regression confusion matrix. We partition  $[0, 1]$  into ten equal intervals:

$$B_i = \begin{cases} [\frac{i}{10}, \frac{i+1}{10}), & i = 0, 1, \dots, 8, \\ [\frac{9}{10}, 1], & i = 9. \end{cases}$$

Define the bin-index function  $\beta: [0, 1] \rightarrow \{0, 1, \dots, 9\}$  by  $\beta(v) := \min(\lfloor 10v \rfloor, 9)$ . The confusion matrix  $\mathbf{C} \in \mathbb{N}^{10 \times 10}$  has entries

$$\mathbf{C}_{ij} = \frac{\#\{f \in \mathcal{F} : \beta(\hat{\mathbb{P}}(f|p)) = i, \beta(\mathbb{P}(f|p)) = j\}}{\#\{f \in \mathcal{F} : \beta(\mathbb{P}(f|p)) = j\}},$$

for  $i, j = 1, \dots, 10$ , where we normalize by the number of true values in the corresponding bin.

## D Additional Results

Table D.1: Model performance comparison on fragmentation ion probability prediction. The best results are in **bold**, the second best ones are underlined.

Model	Fragment ion level			Precursor level		
	L1	MSE	SA	L1	MSE	SA
Global	0.2399 $\pm$ 0.0002	0.1007 $\pm$ 0.0002	0.5205 $\pm$ 0.0007	0.2437 $\pm$ 0.0002	0.0994 $\pm$ 0.0002	0.5578 $\pm$ 0.0004
BoF	0.1788 $\pm$ 0.0002	0.1185 $\pm$ 0.0003	0.4673 $\pm$ 0.0009	0.1788 $\pm$ 0.0001	0.1184 $\pm$ 0.0001	0.5086 $\pm$ 0.0006
LR	0.1293 $\pm$ 0.0003	0.0561 $\pm$ 0.0002	0.6597 $\pm$ 0.0008	0.1258 $\pm$ 0.0002	0.0540 $\pm$ 0.0002	0.6951 $\pm$ 0.0004
ResNet	0.0739 $\pm$ 0.0003	0.0238 $\pm$ 0.0001	0.7845 $\pm$ 0.0006	0.0687 $\pm$ 0.0002	0.0213 $\pm$ 0.0000	0.8182 $\pm$ 0.0003
Transformer	<b>0.0591 <math>\pm</math> 0.0005</b>	<b>0.0180 <math>\pm</math> 0.0003</b>	<b>0.8145 <math>\pm</math> 0.0015</b>	<b>0.0560 <math>\pm</math> 0.0005</b>	<b>0.0165 <math>\pm</math> 0.0003</b>	<b>0.8453 <math>\pm</math> 0.0016</b>

Table D.2: Model performance comparison on fragmentation ion existence. The best results are in **bold**, the second best ones are underlined. **Acc**: accuracy; **Sen**: sensitivity; **Spec**: specificity.

Model	Fragment ion level			Precursor level		
	Acc	Sen	Spec	Acc	Sen	Spec
Global	0.6573 $\pm$ 0.0009	<b>1.0000 <math>\pm</math> 0.0000</b>	0.0000 $\pm$ 0.0000	0.6993 $\pm$ 0.0007	<b>1.0000 <math>\pm</math> 0.0000</b>	0.0000 $\pm$ 0.0000
BoF	0.8141 $\pm$ 0.0006	0.3801 $\pm$ 0.0011	<u>0.8335 <math>\pm</math> 0.0005</u>	0.8027 $\pm$ 0.0008	0.4435 $\pm$ 0.0009	0.7683 $\pm$ 0.0005
LR	0.7345 $\pm$ 0.0043	<u>0.9105 <math>\pm</math> 0.0021</u>	0.3688 $\pm$ 0.0144	0.7661 $\pm$ 0.0053	<u>0.9213 <math>\pm</math> 0.0021</u>	0.3771 $\pm$ 0.0286
ResNet	0.8602 $\pm$ 0.0018	0.8529 $\pm$ 0.0025	0.7342 $\pm$ 0.0039	0.8708 $\pm$ 0.0017	<u>0.8766 <math>\pm</math> 0.0026</u>	0.7150 $\pm$ 0.0038
Transformer	<b>0.9512 <math>\pm</math> 0.0021</b>	0.7634 $\pm$ 0.0055	<b>0.9248 <math>\pm</math> 0.0038</b>	<b>0.9530 <math>\pm</math> 0.0020</b>	0.7963 $\pm$ 0.0055	<b>0.9215 <math>\pm</math> 0.0039</b>

### D.1 Model Comparison on Fragmentation Ion Probability Prediction

Table D.1 shows that across both fragment-ion and precursor-level assessments, transformer consistently achieves the lowest prediction errors and the best spectral-angle alignment, making it the most accurate model for fragment-ion probability calibration. The ResNet comes in as a clear runner-up, confirming that deep nets offer substantial gains over simpler methods. Linear regression provides moderate improvements above the BoF approach, which itself outperforms the naive Global baseline. In sum, there is a clear performance hierarchy, from the global model up through BoF and linear regression to the deep networks, culminating in the transformer’s superior ability to model peptide fragmentation probabilities.

### D.2 Case Studies on ResNet

Table D.3 shows the precursors with the bottom 10 ranks for sensitivity and specificity in a test set. From the table, the presence of fragment ions is underpredicted for precursors with long peptide sequences and high charge states, including 12 out of 15 precursors with peptide lengths of  $\geq 34$ , as well as all precursors with a charge of 3+. The 15 precursors listed have low specificity but high sensitivity, characterized by short peptides (lengths around 10) and a charge of 1+.

The increasing number of possible fragment ions depends on the length of the peptide sequence and the precursor charge. This presents the challenge that model predictions must adequately consider the variance of possible fragment ions for each precursor input.

Table D.3: Precursors with the bottom 10 precursors on sensitivity and specificity in the No.1 test set. **PID**: the precursor index; **Seq**: the precursor sequence; **#PSM**: the number of spectra identified to the precursor; **Len**: the length of the precursor sequence; **ACC**, **Sen** and **Spec**: the same as in Table D.2.

PID	Seq	Charge	#PSMs	Len	Acc	Sen	Spec
The bottom 15 precursors on sensitivity							
248155	IVERPLPGYPDAEAEPESSAGAAEEPSGAGSEELIK	3	635	38	1.00	0.45	1.00
260783	KGSITSVQAIYVPADDLTDPAATTFAHLDAITVLSR	3	1315	37	0.99	0.48	0.98
162400	GAEASAASEEEAGPQATEPSTPSGPESGTPASAEQNE	3	1091	38	0.86	0.49	0.90
224171	IFPPETSASVAATPPPSTASAPAAVNSSASADKPLSNMK	3	949	39	0.95	0.51	0.94
231482	IKQDSNLIGPEGGVLSSSTVVPQVQAVFPEGALTK	3	255	34	0.91	0.51	0.89
309458	LKPAFIKPYGTVTAANSSFLTDGASAMLIMAEK	3	248	34	0.94	0.52	0.91
37623	AQAALQAVNSVQSGNLALAASAAVDAGMAMAGQSPVLR	3	786	39	0.97	0.52	0.96
129471	ESQPSPAQEAGYSTLAQSYPSDLPEEPSSPQER	3	169	34	0.82	0.52	0.79
146277	FIGAGAATVGVSAGGAGIGTVFGSLIIGYAR	3	3347	31	0.97	0.52	0.94
299795	LGGGMPGLGQGPPTDAPAVDTAEQVYISSLALLK	3	782	34	0.96	0.52	0.95
272819	KPPVPLDWAQVQSQGEETNASDQNEPQLGLK	3	593	32	0.95	0.52	0.91
8860	AEDGFEDQILIPVPAPAGGDDDDYIEQTLVTVAAGK	3	463	36	0.93	0.52	0.91
564061	VMDMLHSMGPDVTVVITSSDLSPQGSNYLIVLGSQR	3	209	36	0.89	0.52	0.87
6820	ADIDVSGPSVDTDAFDLDIEGPEGK	3	881	25	0.95	0.53	0.90
93339	EAKPGAAEPEVGVPSLSPSSPSSWTETDVEER	3	558	34	0.96	0.53	0.94
The bottom 15 precursors on Specificity							
240926	IQLVEEELDR	1	56	10	1.00	1.00	0.00
243342	ISEQFTAMFR	1	64	10	0.95	1.00	0.00
249743	IVVVTAGVR	1	91	9	1.00	1.00	0.00
219883	IDIIPNPQER	1	73	10	0.95	1.00	0.00
90206	DYGVLLLEGSLALR	1	36	14	1.00	1.00	0.00
68977	DITSDTSGDFR	1	103	11	0.95	1.00	0.00
67159	DIDEVSLLR	1	32	10	1.00	1.00	0.00
71929	DLFDPIIEDR	1	42	10	0.95	1.00	0.00
85797	DTQIQLDDAVR	1	57	11	0.90	1.00	0.00
600998	YLTVAIFR	1	84	9	0.94	1.00	0.00
373876	NLQGISSFR	1	40	9	0.88	1.00	0.00
386668	NYYEQWGK	1	38	8	0.93	1.00	0.00
340892	LVIITAGAR	1	97	9	1.00	1.00	0.00
238210	INVYYNEATGGK	1	143	12	0.95	0.95	0.00
588833	WTLLQEQGTK	1	39	10	1.00	0.95	0.00