# Appendix for UniNovo : a universal tool for *de novo* peptide sequencing

Kyowon Jeong[1], Sangtae Kim[2], and Pavel A. Pevzner[2]

[1] Department of Electrical and Computer Engineering, University of California, San Diego, CA.
kwj@ucsd.edu
[2] Department of Computer Science and Engineering, University of California, San Diego, CA.
{sak008, ppevzner}@ucsd.edu

## A1 How to train UniNovo

### A1.1 Vector operations

Before we describe the training of UniNovo, we first define the following vector operations. Let $V$ and $W$ be Boolean vectors with $n$ elements.

- $|V|$ is the number of elements in $V$ (i.e., $n$).
  $\Rightarrow$ For $V = (0, 1, 0, 1, 0)$, $|V| = 5$.
- $\langle V\rangle$ is the number of non-zero elements in $V$.
  $\Rightarrow$ For $V = (0, 1, 0, 1, 0)$, $\langle V\rangle = 2$.
- $V \cdot W$ denotes the elementwise multiplication between $V$ and $W$.
  $\Rightarrow$ For $V = (0, 1, 0, 1, 0)$ and $W = (1, 1, 1, 0, 0)$, $V \cdot W = (0, 1, 0, 0, 0)$.
- Given an integer $k$, a vector $V^k$ is a vector obtained by shifting all elements of $V$ by $k$. More formally, $V^k$ is a vector of cardinality $n$ whose elements are given by

  $$V^k(i) = \begin{cases} V(i-k) & \text{if } 1 \le i-k \le n \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

  for $i = 1, \cdots, n$.
  $\Rightarrow$ For $V = (0, 1, 0, 1, 0)$, $V^{-2} = (0, 1, 0, 0, 0)$ and $V^{+1} = (0, 0, 1, 0, 1)$.

### A1.2 Description of the training

UniNovo takes the PSMs in the training dataset $\mathcal{T}$ and learns important ion types and features automatically. The training of UniNovo consists of two stages: ion type selection and feature detection.

**Ion type selection** In the ion type selection step, the frequently observed ion types are selected from the training dataset using the *offset frequency function* (OFF) introduced by Dancik et al. 1999 [1]. And the probabilities in the ion type matrices for the ion types (e.g., $\alpha$ and $\beta$ in Figure 1 (a)) are learned.

Given an ion type $\delta$, OFF outputs the empirical probability that a $\delta$-ion peak is observed for a fragmentation site in the training dataset. We define OFF as follows: The input to OFF is the training dataset $\mathcal{T}$. OFF is defined by

$$OFF(\delta) = \frac{\sum_{(P,S)\in\mathcal{T}} \overbrace{\langle S \cdot P^{\delta}\rangle}^{\#\ \delta\text{-ion peaks in } S}}{\sum_{(P,S)\in\mathcal{T}} \underbrace{\langle P\rangle}_{\#\ \text{fragmentation sites in } P}}. \tag{2}$$

Out of all ion types $\delta$ satisfying $-38 < \delta < 38$, we pick 8 ion types $\delta$ with the highest values of $OFF(\delta)$. We denote the set of the selected ion types as *ion type set* $\Delta$ (see Table A2 for the list of the ion types in the ion type set for each dataset).

After learning the ion type set $\Delta$, we learn $\alpha$ and $\beta$ for each ion type in $\Delta$. Given an ion type $\delta$, $\alpha$ is simply given by $\alpha := OFF(\delta)$. $\beta$ can be obtained by

$$\beta = \frac{\sum_{(P,S)\in\mathcal{T}} \overbrace{\langle S\rangle - \langle S \cdot P^{\delta}\rangle}^{\#\ \text{non-}\delta\text{-ion peaks in } S}}{\sum_{(P,S)\in\mathcal{T}} \underbrace{|P| - \langle P\rangle}_{\#\ \text{non-fragmentation sites in } P}}. \tag{3}$$

We also learn the empirical probability that a random mass $i$ is a fragmentation site. To learn this probability, first an element in the peptide of each PSM is selected randomly. The probability is estimated by the frequency of the selected elements being fragmentation sites. The learned probability is called *prior fragmentation probability* and is denoted by $p$.

**Feature detection step** The feature detection step aims to detect the features that the peaks of the ion types in $\mathit{\Delta}$ often satisfy. Besides, the probabilities in the feature-ion type matrices ($\mu$ and $\nu$ in Figure 1 (a)) are learned.

The features are detected using OFF with a slight modification, which is called a *feature frequency function (FFF)*. Given an ion type $\delta$ and a feature $f$, FFF outputs the empirical probability that a $\delta$-ion peak satisfies $f$.

The inputs to FFF are the training dataset $\mathcal{T}$, an ion type $\delta$, and a feature $f$. FFF for $\delta \in \mathit{\Delta}$ is defined by

$$FFF(\delta, f) = \frac{\sum\limits_{(P,S)\in\mathcal{T}} \overbrace{\langle S \cdot S^{-f} \cdot P^{\delta} \rangle}^{\#\ \delta\text{-ion peaks satisfying } f \text{ in } S}}{\sum\limits_{(P,S)\in\mathcal{T}} \underbrace{\langle S \cdot P^{\delta} \rangle}_{\#\ \delta\text{-ion peaks in } S}}. \tag{4}$$

We select all features $f$ such that $FFF(\delta, f) > 0.15$ and $-38 < x < 38$ for $\delta \in \mathit{\Delta}$. The selected features are called an *offset features*. Since the features satisfying $FFF(\delta, f) > 0.15$ are selected regardless of the size of the feature set, the total number of features in UniNovo is not fixed. In general, the total number was about several thousands.

In addition, the features $f = m(a), a \in A$ are selected, and the selected features are called *linking features*. A linking feature characterizes two peaks whose mz difference equals to an amino acid mass. The set of selected offset and linking features is named as the *feature set* and is denoted by $F$.

Given an ion type $\delta \in \mathit{\Delta}$ and a feature $f \in F$, we learn $\mu$ and $\nu$. $\mu$ is simply given by $FFF(\delta, f)$ whereas $\nu$ is given by

$$\nu = \frac{\sum\limits_{(P,S)\in\mathcal{T}} \overbrace{\langle S \cdot S^{-f} \rangle - \langle S \cdot S^{-f} \cdot P^{\delta} \rangle}^{\#\ \text{non-}\delta\text{-ion peaks satisfying } f \text{ in } S}}{\sum\limits_{(P,S)\in\mathcal{T}} \underbrace{\langle S \rangle - \langle S \cdot P^{\delta} \rangle}_{\#\ \text{non-}\delta\text{-ion peaks in } S}}. \tag{5}$$

## A2 How to extend UniNovo algorithm for the realistic model

### A2.1 Changes in the model/definitions

(a)

| # Partition | Precursor mass (×121.6) |
|---|---|
| 1 | $< 9$ |
| 2 | 9-13 |
| 3 | 13-17 |
| 4 | 17-20 |
| 5 | $> 20$ |

(b)

| $I(S_i)$ | Intensity rank of a peak $i$ |
|---|---|
| 10 | 1-10 |
| 9 | 11-20 |
| 8 | 21-30 |
| 7 | 31-40 |
| 6 | 41-50 |
| 5 | 51-60 |
| 4 | 61-70 |
| 3 | 71-80 |
| 2 | 81-90 |
| 1 | 91-150 |
| 0 | ≥150 |

(c)

| $R(u,t)$ | $u/t$ |
|---|---|
| $-\infty$ | $\infty$ |
| $-4$ | 5-$\infty$ |
| $-3$ | 2.5-5 |
| $-2$ | 1.7-2.5 |
| $-1$ | 1.3-1.7 |
| 0 | 1.0-1.3 |
| 1 | 0.8-1.0 |
| 2 | 0.6-0.8 |
| 3 | 0.4-0.6 |
| 4 | 0.2-0.4 |
| 5 | 0.0-0.2 |

**Table A1.** Partitioning of spectra and peak intensities. (a) partitioning of the spectra by their parent mass. 121.6 is the average amino acid mass. (b) the intensity level of a peak $i$ in a spectrum $S$, denoted by $I(S_i)$. The intensity level of a peak is decided by its intensity rank (the $i$th highest intensity peak = rank $i$). (c) Definition of the intensity ratio function $R : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{Z}$. This function is used to define a feature.

In practice, UniNovo considers the model in which

- the mass tolerances of MS1 and MS2 are given by users
- the mono-isotopic masses of amino acids are used (e.g., the mass of Gly is 57.021464), and the m/z positions of peaks are real numbers (when MS1 tolerance is smaller than 0.1 Da; otherwise, integer amino acid masses and m/z values are used)
- all spectra are divided into 5 groups according to their parent mass ranges (see Table A1 (a))
- the intensities of peaks are divided into 11 levels (see $I(S_i)$ in Table A1 (b))
- only 150 peaks with high intensities are considered per a spectrum
- both $N$- and $C$- terminal ions of any charge values up to 4 are considered (e.g., $b$, $y$, $b^2$ ions) (see Table A2)

The peak $i$ in a spectrum $S$ refers to the peak whose m/z value equals to $i$ within MS1 mass tolerance.[1] The raw intensity of the peak $i$ is denoted by $S_i$ and the intensity level of it is by $I(S_i)$. If multiple such peaks are present, simply pick the highest intensity one. UniNovo learns/applies all parameters ($\alpha$, $\beta$, $\mu$, and $\nu$) separately for different groups of spectra. Also, the parameters are learned separately for the fragmentation sites corresponding to the enzyme specific amino acids (e.g., C-terminal K or R for tryptic peptides). Except those amino acids, the current version of UniNovo does not take amino acid specific information (e.g., the different propensities of amino acids or the effect of proline on fragmentations) into account. Many studies have reported that different amino acids alter the fragmentation characteristics of MS/MS spectra [11,4]. By considering amino acids differently as in PepNovo, the performance of UniNovo could be improved; however, the number of annotated spectra necessary for the training of UniNovo should be also increased by orders of magnitude to avoid overfitting, which may weaken the universal property of UniNovo. A possible

[1] Even if a different mass tolerance is used, no fundamental change is necessary for UniNovo algorithm. We only need to redefine what the definition of peak $i$ in a spectrum is.

| Dataset | | Ion types in the ion type set $\Delta$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| CID2 | $\delta$ | $y$ | $b$ | $y+i^*$ | $b-H_2O$ | $b+i$ | $b-NH_3$ | $y-H_2O$ | $y-NH_3$ |
| | $OFF(\delta)$ | 0.74 | 0.62 | 0.47 | 0.35 | 0.29 | 0.26 | 0.16 | 0.14 |
| CIDL2 | $\delta$ | $y$ | $b$ | $y+i$ | $b+i$ | $b-H_2O$ | $b-NH_3$ | $y-H_2O$ | $y-NH_3$ |
| | $OFF(\delta)$ | 0.66 | 0.57 | 0.40 | 0.30 | 0.24 | 0.21 | 0.16 | 0.14 |
| CIDA2 | $\delta$ | $b$ | $y$ | $b+i$ | $y+i$ | $b-H_2O$ | $b-NH_3$ | $y-H_2O$ | $y-NH_3$ |
| | $OFF(\delta)$ | 0.60 | 0.55 | 0.34 | 0.27 | 0.19 | 0.15 | 0.14 | 0.14 |
| ETD2 | $\delta$ | $z+i$ | $z$ | $z+2i^{**}$ | $y$ | $c$ | $c-H$ | $c+i$ | $y+i$ |
| | $OFF(\delta)$ | 0.56 | 0.47 | 0.29 | 0.25 | 0.23 | 0.22 | 0.16 | 0.15 |
| ETD3 | $\delta$ | $c$ | $z$ | $z+i$ | $c+i$ | $y$ | $z+2i$ | $c-H$ | $c+2i$ |
| | $OFF(\delta)$ | 0.56 | 0.51 | 0.46 | 0.36 | 0.25 | 0.20 | 0.15 | 0.13 |
| ETDL3 | $\delta$ | $c$ | $z$ | $z+i$ | $c+i$ | $y$ | $z+2i$ | $c+2i$ | $a+i$ |
| | $OFF(\delta)$ | 0.66 | 0.60 | 0.44 | 0.42 | 0.20 | 0.18 | 0.15 | 0.12 |
| ETDL4 | $\delta$ | $c$ | $z+i$ | $z$ | $c+i$ | $z+2i$ | $z^2$ | $c^2$ | $y$ |
| | $OFF(\delta)$ | 0.44 | 0.33 | 0.33 | 0.28 | 0.20 | 0.17 | 0.15 | 0.13 |
| ETDA3 | $\delta$ | $c$ | $z$ | $z+i$ | $c+i$ | $z+2i$ | $y$ | $c+2i$ | $a+i$ |
| | $OFF(\delta)$ | 0.60 | 0.50 | 0.40 | 0.38 | 0.19 | 0.16 | 0.13 | 0.13 |
| ETDA4 | $\delta$ | $c$ | $z+i$ | $z$ | $c+i$ | $z+2i$ | $z^2$ | $c^2$ | $c+2i$ |
| | $OFF(\delta)$ | 0.42 | 0.32 | 0.30 | 0.27 | 0.21 | 0.16 | 0.15 | 0.12 |
| HCD2 | $\delta$ | $y$ | $b$ | $y+i$ | $a$ | $b-H_2O$ | $y-H_2O$ | $y-NH_3$ | $y^2$ |
| | $OFF(\delta)$ | 0.52 | 0.21 | 0.20 | 0.10 | 0.08 | 0.08 | 0.08 | 0.04 |
| HCD3 | $\delta$ | $y$ | $b$ | $y^2$ | $y+i$ | $y^2+i$ | $a$ | $b-H_2O$ | $b+i$ |
| | $OFF(\delta)$ | 0.26 | 0.16 | 0.12 | 0.07 | 0.06 | 0.05 | 0.03 | 0.02 |

**Table A2.** The ion types in the ion type set $\Delta$ and their OFF values for different datasets. $(*, **)$: $y+i$ denotes the $y$-ion of a fragmented peptide with one isotope, and $z+2i$ denotes the $z$-ion of a fragmented peptide with two isotopes.

idea to mitigate such a negative effect may be to cluster amino acids into a small number of groups (e.g., basic and non-basic groups) and to train the parameters separately for each group, which will be included in our future work.

The definition of a feature $f$ is changed so that it can accommodate the changed model. Before we define a feature $f$, define the *intensity ratio function* $R(u,t)$, a function from two real numbers $u, t$ to an integer, as in Table A1 (c). A feature $f = (t, x, r, T, z_1, z_2)$ is now a vector with 6 elements (instead of a single integer in the manuscript): intensity $t$, mass offset $x$, intensity ratio $r$, terminal $T$, base peak charge $z_1$, and support peak charge $z_2$. The mass offset $x$ represents a mass gain/loss, and $T$ shows if the feature represents the relation between the ions of the same terminal ($T = 0$) or not ($T = 1$). Given a spectrum $S$ from a peptide of mass $n$,[2] a peak $i$ in $S$ is said to *satisfy* $f = (t, x, r, T, z_1, z_2)$ if $I(S_i) = t$ and there exists another peak $j$ such that $R(S_i, S_j) = r$ where $j$ is given by

$$j = \begin{cases} \frac{z_1 \cdot (i-\epsilon)+x}{z_2} + \epsilon & \text{if } T = 0 \\ \frac{n-(z_1 \cdot (i-\epsilon)+x)}{z_2} + \epsilon & \text{otherwise.} \end{cases}$$

where $\epsilon$ is the mass of a proton. With the new definition, a feature can characterize the m/z (by specifying $x$) and intensity relation (by specifying $t$ and $r$) between two peaks of ion types of different terminus (by specifying $T$) and/or different charges (by specifying $z_1$ and $z_2$).

[2] The peptide mass $n$ can be calculated from the parent mass of the spectrum.

### A2.2 Iterative training/running for better ion type inference

The $FPV$'s for different ion types can be used to assign a probability distribution $\rho$ (over ion types in $\mathit{\Delta}$ and *noise*) to each peak such that $\rho(\delta)$ is the probability that the peak is a $\delta$-ion peak and $\rho$(noise) is the probability that the peak is not a $\delta$-ion peak for all $\delta \in \mathit{\Delta}$ (termed a *noisy* peak).

The distribution $\rho$ is meaningful only when it is far from a uniform distribution. However, if the spectra in the training dataset $\mathcal{T}$ contain abundant noisy peaks or peaks of different ion types with similar characteristics, the distribution often has a uniform-like distribution. Thus, it can be more informative given a training dataset consisting of spectra containing few noisy peaks and peaks of different ion types with distinctive characteristics.

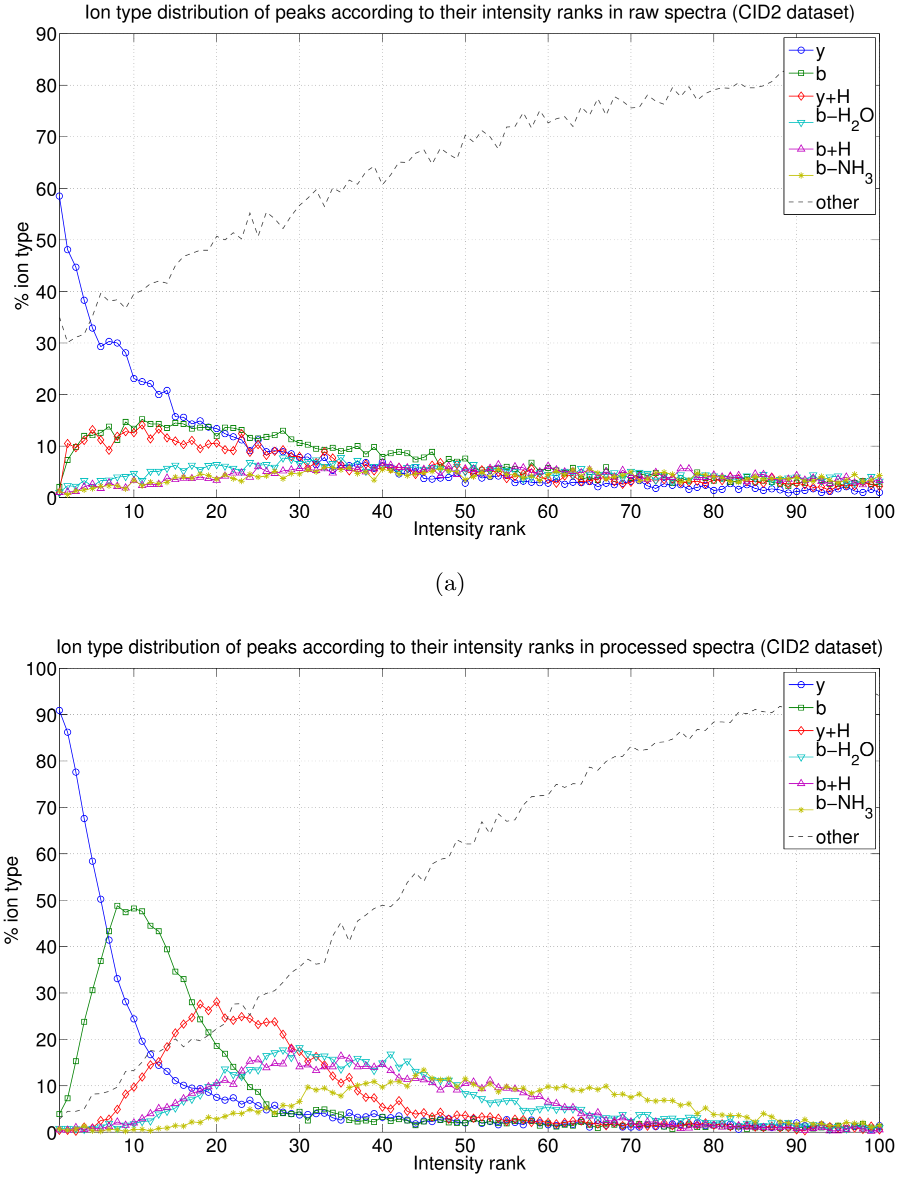
To obtain such a training dataset, we generate *processed PSMs* from PSMs in the (original) training dataset. Given an ion type set $\mathit{\Delta}$ and the distribution $\rho$ of the peak $i$ in a spectrum $S$, the *processed spectrum* $S'$ is a spectrum satisfying

$$S'_i = \sum_{\delta \in \mathit{\Delta}} \rho(\delta) \cdot OFF(\delta) \tag{6}$$

for all $i$ such that $S_i > 0$. Since the intensity of a peak in the processed spectrum $S'$ is a weighted summation of the distribution $\rho$ of the peak, it is likely that the peaks in $S'$ are clustered according to the ion types of the peaks (see Figure A1). Denote the (original) training dataset as $\mathcal{T}_1$. The ion type set $\mathit{\Delta}$ and feature set $F_1$ are learned from $\mathcal{T}_1$. For each PSM $(P, S)$ in $\mathcal{T}_1$, the processed spectrum $S'$ is generated from $S$ using features in $F_1$, yielding a *processed PSM* $(P, S')$. The resulting set of the processed PSMs is denoted as $\mathcal{T}_2$. Likewise, we repeat generating $\mathcal{T}_{i+1}$ using features in $F_i$ learned from $\mathcal{T}_i$ for $i = 1, \cdots, 4$.[3] The feature sets $F_1, \cdots, F_4$ are kept by UniNovo.

An input spectrum undergoes the same iterative process. Denote an input spectrum as $S_1$. We generate the (processed) spectrum $S_{i+1}$ from $S_i$ using features in $F_i$ learned from $\mathcal{T}_i$ for $i = 1, \cdots, 4$. The $FPV$ is generated based on the distributions $\rho$ after 5 iterations.

[3] After 5 iterations, no significant changes were observed in the resulting training dataset.



(a)

(b)

**Fig. A1.** Ion type distribution of peaks according to their intensity ranks for raw (a) and processed (after 5 iterations) spectra (b). CID2 dataset was used. A peak with the intensity rank $i$ is the $i$th highest intensity peak in the spectrum. In raw spectra, different ion types are spread over the intensity ranks of peaks. Even in case of the highest intensity peaks, only 60% of them are $y$-ion peaks. The ion types in processed spectra are well clustered according to the intensity ranks of peaks. For example, 90% of the highest intensity peaks are $y$-ion peaks in processed spectra.

## A3 How to derive the posterior probability $Pr(P_i = 1|S_i = 1 \text{ and } S_{i+f} = 1 \text{ for } f \in H)$

### A3.1 One ion type ($\delta = 0$) and independent features

If $H$ is an empty set, it reduces to $Pr(P_i = 1|s_i = 1)$. By Bayes's rule, we have

$$Pr(P_i = 1|s_i = 1) \propto Pr(P_i = 1) \cdot Pr(s_i = 1|P_i = 1) = p \cdot \alpha. \tag{7}$$

Similarly, we obtain $Pr(P_i = 0|s_i = 1) \propto (1-p)\cdot\beta$. Since $Pr(P_i = 1|s_i = 1)+Pr(P_i = 0|S_i^i = 1) = 1$, we obtain

$$Pr(P_i = 1|s_i = 1) = \frac{Pr(P_i = 1|s_i = 1)}{Pr(P_i = 1|s_i = 1) + Pr(P_i = 0|s_i = 1)} = \frac{p \cdot \alpha}{p \cdot \alpha + (1-p) \cdot \beta}. \tag{8}$$

Denote this probability as $\gamma$. Then, we obtain

$$\begin{aligned}
&Pr(P_i = 1|s_i = 1 \text{ and } s_{i+f} = 1 \text{ for } f \in H) && (9)\\
&\propto Pr(P_i = 1|s_i = 1) \cdot Pr(s_{i+f} = 1 \text{ for } f \in H|P_i = 1, S_i = 1) && (10)\\
&= \gamma \cdot Pr(s_{i+f} = 1 \text{ for } f \in H|P_i = 1, s_i = 1) && (11)\\
&= \gamma \cdot \prod_{f \in H} Pr(s_{i+f} = 1|P_i = 1, s_i = 1) && (12)\\
&= \gamma \cdot \prod_{f \in H} \mu_f && (13)
\end{aligned}$$

where $\mu_f$ denotes the probability $\mu$ associated to the feature $f$. The equality between (11) and (12) is obtained from the assumed independence between features. Likewise, we can show $Pr(P_i = 0|s_i = 1 \text{ and } s_{i+f} = 1 \text{ for } f \in H) \propto (1-\gamma) \cdot \prod_{f \in H} \nu_f$ where $\nu_f$ is the probability $\nu$ (Figure 2 (a)) associated to the feature $f$. Therefore, we obtain

$$\begin{aligned}
Pr(P_i = 1|S_i = 1 \text{ and } S_{i+f} = 1 \text{ for } f \in H) &= Pr(P_i = 1|s_i = 1 \text{ and } s_{i+f} = 1 \text{ for } f \in H) && (14)\\
&= \frac{\gamma \cdot \prod_{f \in H} \mu_f}{\gamma \cdot \prod_{f \in H} \mu_f + (1-\gamma) \cdot \prod_{f \in H} \nu_f}. && (15)
\end{aligned}$$

### A3.2 Multiple ion types and multiple but independent features

Next, we consider the case in which multiple ion types are present in the ion type set $\Delta$. For an ion type $\delta \in \Delta$, the expression (15) can be generalized as

$$Pr(P_{i-\delta} = 1|S_i = 1 \text{ and } S_{i+f} = 1 \text{ for } f \in H) = \frac{\gamma_\delta \cdot \prod_{f \in H} \mu_f^\delta}{\gamma_\delta \cdot \prod_{f \in H} \mu_f^\delta + (1-\gamma_\delta) \cdot \prod_{f \in H} \nu_f^\delta} \tag{16}$$

where $\gamma_\delta$ denotes $\gamma$ of the ion type matrix for the ion type $\delta$, and $\mu_f^\delta$ ($\nu_f^\delta$) denotes $\mu$ ($\nu$) for the feature $f$ and the ion type $\delta$. Denote the obtained probability in (16) as $\pi_i^\delta$. For each ion type $\delta$, we derive a fragmentation probability vector $FPV^\delta$ as

$$FPV_i^\delta = \begin{cases} \pi_{i+\delta}^\delta & \text{if } S_{i+\delta} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

for $i = 1, \cdots, n-1$. $FPV_n^\delta$ is again defined to be 1. The final fragmentation probability vector $FPV$ is generated by taking elementwise (weighted) summation of $FPV^\delta$'s for $\delta \in \Delta$. The weights are decided by an MMSE (minimum mean squared error) estimation method as described below.

For simplicity, we start with the case in which the ion type set is given by $\Delta = \{\delta, \delta'\}$. Given a spectrum , UniNovo generates 2 fragmentation probability vectors ($FPV^\delta$ and $FPV^{\delta'}$), and the final $FPV$ is generated by elementwise weighted summation of these fragmentation probability vectors.

The weights are learned from the training dataset $\mathcal{T}$ as follows: For each PSM $(P, S)$ in $\mathcal{T}$, we first generate $FPV^\delta$ and $FPV^{\delta'}$. Given an index $i$, we consider three different cases for $(FPV_i^\delta, FPV_i^{\delta'})$: only $FPV_i^\delta$ is non-zero, only $FPV_i^{\delta'}$ is non-zero, and both are non-zero. The weights are learned separately for each case (and are multiplied separately for each case when we generate the final $FPV$). We describe the last case (both are non-zero) only. Let $\overline{X}$ denotes the sample mean of $X$. For instance, $\overline{FPV_i^\delta}$ denotes the sample mean of $FPV_i^\delta$.

The autocorrelation matrix $\mathbf{R}$ is defined as

$$\mathbf{R} = \begin{bmatrix} \overline{FPV_i^\delta FPV_i^\delta} & \overline{FPV_i^\delta FPV_i^{\delta'}} \\ \overline{FPV_i^{\delta'} FPV_i^\delta} & \overline{FPV_i^{\delta'} FPV_i^{\delta'}} \end{bmatrix}$$

and the crosscorrelation matrix $\mathbf{C}$ is defined as

$$\mathbf{C} = \begin{bmatrix} \overline{FPV_i^\delta P_i} \\ \overline{FPV_i^{\delta'} P_i} \end{bmatrix}.$$

The weight vector is given by

$$W = \mathbf{R}^{-1}\mathbf{C}.$$

When more than two ion types are present in the ion type set $\Delta = \{\delta_1, \cdots, \delta_l\}$, UniNovo generates $l$ fragmentation probability vectors, $FPV^{\delta_1}, \cdots, FPV^{\delta_l}$. The weight vectors are learned as above separately for $2^l - 1$ different cases for $(FPV^{\delta_1}, \cdots, FPV^{\delta_l})$: only $FPV_i^{\delta_1}$ is non-zero, only $FPV_i^{\delta_2}$ is non-zero, $\cdots$, all $FPV_i^{\delta_1}$ to $FPV_i^{\delta_l}$ are non-zero.

### A3.3 Multiple ion types and multiple dependent features:

The above derivations of the posterior probability are valid only if the features in $H$ are mutually independent. However, in practice, some features are often strongly correlated (e.g., a feature describing the loss of a water molecule and another describing the loss of two water molecules). Thus, out of all features that a peak $i$ satisfies, a few "statistically meaningful" features that are weakly correlated are automatically selected for $H$.

To select statistically meaningful and weakly correlated features out of $H$, we first define the *divergence* of a feature $f$. We again assume that only one ion type $\delta = 0$ is present in the ion type set $\Delta$. If two probabilities $Pr(P_i = 1|S_i = 1)$ and $Pr(P_i = 1|S_i = S_{i+f} = 1)$ are similar to each other, we can conclude that the feature $f$ is not helpful to determine the fragmentation sites. The two probabilities are given by $p \cdot \alpha$ and $\gamma \cdot \mu_f$ by the equation (1) in the manuscript. We define two distributions $B$ and $C$ over $\delta = 0$ and $-\infty$ such that $B(0) = p \cdot \alpha$ and $C(0) = \gamma \cdot \mu_f$. $B(-\infty) := 1 - B(0)$ and $C(-\infty) := 1 - C(0)$ are called *noise probabilities*. The *divergence* of the feature $f$ is defined by the Kullback-Leibler (KL) divergence between $B$ and $C$.

When more than one ion types are considered, we define two distributions $B$ and $C$ over ion types $\delta \in \Delta$ and $-\infty$ such that $B(\delta) = p \cdot \alpha_\delta$ and $C(\delta) = \gamma_\delta \cdot \mu_f^\delta$. The noise probabilities for $B$ and $C$ are given by $1 - \sum_{\delta \in \Delta} B(\delta)$ and $1 - \sum_{\delta \in \Delta} C(\delta)$, respectively. The divergence of $f$ is defined by the

KL divergence between $B$ and $C$. The features in the feature set $F$ are ranked according to the divergences (the higher divergence, the higher rank) after the training of UniNovo.

Given a peak $i$, all features that the peak $i$ satisfies are divided into different groups as follows: First, the linking features make one group. Second, the offset features make the second group (in the extended model described in the section A2, the offset features are again divided into different groups according to the combination of terminal $T$, base peak charge $z_1$, and support peak charge $z_2$.[4]) Then, per each group of features, we select the highest ranking feature for the set $H$. All features in $H$ are assumed to be independent.

The features in $H$ are assumed to be independent and the $FPV$ is obtained as above. Figures A2 and A3 in the section A9 (blue bars) show that the $FPV_i$ reliably estimates the probability that $P_i = 1$ for various types of spectra.

[4] The rationale behind this selection is that two ions of the different terminus or charge states are likely to be weakly correlated each other.

## A4 How to derive the accuracy of reconstructions using ***Hunter's bound*** [5]

To estimate the accuracy of a reconstruction (i.e., a probability that the reconstruction is correct), we first learn one more statistic from the training dataset $\mathcal{T}$: *EdgeAccuracy* of edges in the spectrum graph. Given an edge $(i, j)$, $EdgeAccuracy(i, j)$ is an empirical probability of $(i, j)$ being correct. More precisely, each edge $(i, j)$ is characterized by the following quantities: $FPV_i$, $FPV_j$ (quantized into 10 levels), and the minimum amino acid number whose total mass equals to $j - i$. Call these quantities of an edge the *property* of the edge. From the training dataset $\mathcal{T}$, we obtain the empirical probability that an edge with a property is correct for all possible properties. Then, given an edge $(i, j)$ (generated from a query spectrum), denote the learned empirical probability for the property of the edge by $q$. The $EdgeAccuracy(i, j)$ is given by $\min(q, FPV_i, FPV_j)$.[5]

The accuracy of a reconstruction is then derived from $FPV_i$ of its vertices $i$ and $EdgeAccuracy(i, j)$ of its edges $(i, j)$ using an upper bound for the probability of a union proposed by Hunter, 1976 [5]. Given a reconstruction $r = \{i_0, \cdots, i_l\}$ on the spectrum graph (of a spectrum $S$ from an unknown peptide $P$), we consider a probability space $(\Omega_r, \mathcal{F}_r, Pr_r)$ whose sample space $\Omega_r$ is given by

$$\Omega_r = \{P_i = x : i \in r,\ x = 0, 1\}. \tag{18}$$

The set of events $\mathcal{F}_s$ is composed of all subsets of $\Omega_r$. Let $Pr_r(P_i = 1) = FPV_i$, and $Pr_r(P_i = 1, P_{i'} = 1) = EdgeAccuracy(i, i')$ for $i, i'$ in $r$. The probability we want to derive can be written as $Pr_r(\bigcap\limits_{i \in r} P_i = 1)$.

To use Hunter's bound, we construct a complete graph[6] $\mathcal{E}$ whose vertex $i$ represents the event $P_i = 1$ for $i \in r$. The weight of an edge $(i, j)$ is defined by

$$\begin{aligned} w_{(i,j)} &= Pr_r(P_i = 1 \text{ or } P_j = 1) &(19)\\ &= Pr_r(P_i = 1) + Pr_r(P_j = 1) - Pr_r(P_i = 1, P_j = 1) &(20)\\ &= FPV_i + FPV_j - EdgeAccuracy(i, j). &(21) \end{aligned}$$

Hunter's bound gives us the following bound:

$$\begin{aligned} Pr_r(\bigcap_{i \in r} P_i = 1)) &\geq \sum_{i \in r} Pr_r(P_i = 1) - \sum_{(i,j) \in T_{\mathcal{E}}} w_{(i,j)} &(22)\\ = \sum_{i \in r} FPV_i &- \sum_{(i,j) \in T_{\mathcal{E}}} (FPV_i + FPV_j - EdgeAccuracy(i, j)). &(23) \end{aligned}$$

where $T_{\mathcal{E}}$ is the minimum spanning tree on the graph $\mathcal{E}$. The expression in (23) defines the accuracy of the reconstruction $r$.

[5] $FPV_i$ estimates the probability that the vertex $i$ is correct, and $EdgeAccuracy(i, j)$ estimates the probability both the vertices $i$ and $j$ are correct. To construct a probability space based on these estimates (see below), $EdgeAccuracy(i, j)$ is forced to be smaller than both $FPV_i$ and $FPV_j$.

[6] An undirected graph in which every pair of distinct vertices is connected by a unique edge.

## A5 How to merge spectrum graphs generated from spectra of paired acquisition modes

Given multiple spectrum graphs $G^1, \cdots, G^n$, we define a merged spectrum graph $G$. To define the spectrum graph $G$, we need to define the vertices along with their scores. The vertices of $G$ are given by the union of vertices of the input spectrum graphs. The score of a vertex $i$ in $G$ is given by $\sum\limits_{k=1}^{n} G_i^k$. To calculate the accuracy of a reconstruction $r$ on the merged graph $G$, we also need to redefine $FPV$ and $EdgeAccuracy$ of $G$. For each the input spectrum graph $G^k$, $FPV_i$ and $EdgeAccuracy(i, j)$ are defined for each $i$ and $j$. The $FPV_i$ ($EdgeAccuracy(i, j)$) of $G$ is simply defined as the maximum value of these $FPV_i$ ($EdgeAccuracy(i, j)$) of the input spectrum graphs.

## A6 How to derive the spectrum accuracy

The spectrum accuracy of a set of reconstructions predicts a probability that at least one of the reconstructions is correct. Given a set of reconstructions $R = \{r_1, \cdots, r_N\}$, we define a variable $D_i$ for $i = 1, \cdots, N$ as

$$D_i = \begin{cases} 1 & \text{if } r_i \text{ is correct} \\ 0 & \text{otherwise.} \end{cases} \tag{24}$$

We consider a probability space $(\Omega_R, \mathcal{F}_R, Pr_R)$ whose sample space $\Omega_R$ is defined by

$$\Omega_R = \{D_i = x : i = 1, \cdots, N,\ x = 0, 1\}. \tag{25}$$

The set of events $\mathcal{F}_R$ is composed of all subsets of $\Omega_R$. Given two sequences $r_i$ and $r_j$, we define $r_{i,j}$ as a reconstruction whose vertices are the union of those of $r_i$ and $r_j$. For example, $r_1 = \{1, 2, 4, 5\}$ and $r_2 = \{1, 3, 4, 5\}$, $r_{1,2} = \{1, 2, 3, 4, 5\}$.

Denote the accuracy of a reconstruction $r$ by $Accuracy(r)$. Let $Pr_R(D_i = 1) := Accuracy(r_i)$ and $Pr_R(D_i = 1, D_j = 1) := Accuracy(r_{i,j})$ for $i, j = 1, \cdots, N$. We assume that a sequence of (Bernoulli) random variables $D_1, D_2, \cdots, D_N$ forms a Markov chain.[7] The probability we want to estimate can be denoted by $Pr_R(\bigcup_{i=1}^{N} D_i = 1)$. Since $Pr_R(D_i = 1 \cup D_j = 1) = Accuracy(r_i) + Accuracy(r_j) - Accuracy(r_{i,j})$, we obtain

$$Pr_R(\bigcup_{i=1}^{N} D_i = 1) \tag{26}$$

$$= 1 - Pr_R(\bigcap_{i=1}^{N} D_i = 0) \tag{27}$$

$$= 1 - Pr_R(D_1 = 0) Pr_R(D_2 = 0 | D_1 = 0) \cdots Pr_R(D_N = 0 | D_{N-1} = 0, \cdots, D_1 = 0) \tag{28}$$

$$= 1 - Pr_R(D_1 = 0) \prod_{i=2}^{N} Pr_R(D_i = 0 | D_{i-1} = 0) \tag{29}$$

$$= 1 - Pr_R(D_1 = 0) \prod_{i=2}^{N} \frac{Pr_R(D_i = 0, D_{i-1} = 0)}{Pr_R(D_{i-1} = 0)} \tag{30}$$

$$= 1 - Pr_R(D_1 = 0) \prod_{i=2}^{N} \frac{1 - Pr_R(D_i = 1 \cup D_{i-1} = 1)}{Pr_R(D_{i-1} = 0)} \tag{31}$$

$$= 1 - (1 - Accuracy(r_1)) \prod_{i=2}^{N} \frac{1 - (Accuracy(r_i) + Accuracy(r_{i-1}) - Accuracy(r_{i,i-1}))}{1 - Accuracy(r_{i-1})} \tag{32}$$

where the equality between (28) and (29) is obtained from the Markov chain assumption. The right hand side of (32) defines the spectrum accuracy of $R$, denoted by $SpectrumAccuracy(R)$.

[7] This assumption is reasonable if two adjacent reconstructions in $R$ are similar each other whereas other reconstructions are relatively dissimilar, which is often the case since reconstructions in $R$ are sorted in the ascending order of their scores (see below).

## A7 How to generate the output set with a high spectrum accuracy

Using the spectrum accuracy, users can control the accuracy of their output reconstruction sets. Given parameters $SpectrumAccuracyThreshold > 0$ and $N$, UniNovo tries to construct a reconstruction set $R$ such that $SpectrumAccuracy(R) \geq SpectrumAccuracyThreshold$ and $|R| \leq N$ by selecting both accurate and long reconstructions (long reconstructions are not accurate, in general). First UniNovo generates 100 high-scoring reconstructions (the *candidate* reconstruction set). The reconstructions in the candidate reconstruction set are sorted by their scores in descending order. Denote the sorted list of reconstructions as $C = \{r_1, r_2, \cdots, r_{100}\}$. A set of reconstructions $R$ is initialized as an empty set, and an integer $MaxLength$ is initialized as one plus the length of the longest reconstruction in $C$. The reconstructions in $C$ whose length are less than $MaxLength$ are added to $R$ sequentially, starting from $r_1$. When $|R| = N$ or all reconstructions shorter than $MaxLength$ are added to $R$, $SpectrumAccuracy(R)$ is calculated. If $SpectrumAccuracy(R) \geq SpectrumAccuracyThreshold$, UniNovo outputs $R$. Otherwise, $MaxLength$ is decreased by 1, $R$ is again initialized as an empty set, and the above procedure is repeated until $MaxLength = 5$. If no output is generated when $MaxLength = 5$, the input spectrum is declared as a low quality spectrum and is filtered out.

## A8 Parameters for UniNovo, PepNovo+, PEAKS, and pNovo in different datasets

Table A3 shows the parameters of the tested tools for different datasets. For all tools, no spectrum quality filtering or charge/parent mass correction, if any, was used. The MS1 and MS2 tolerances in the 4th and 5th columns are used both for tool parameters and for the error tolerances for the experiments.

(a)

| Tool | Fragmentation method | Enzyme specificity | MS1 tolerance | MS2 tolerance |
|---|---|---|---|---|
| UniNovo | CID | Trypsin | 20 ppm | 0.5 Da |
| PepNovo+ | CID | Trypsin | 0.02 Da | 0.5 Da |
| PEAKS | CID | Trypsin | 20 ppm | 0.5 Da |

(b)

| Tool | Fragmentation method | Enzyme specificity | MS1 tolerance | MS2 tolerance |
|---|---|---|---|---|
| UniNovo | CID | LysC | 20 ppm | 0.5 Da |
| PepNovo+ | CID | Trypsin | 0.02 Da | 0.5 Da |
| PEAKS | CID | LysC | 20 ppm | 0.5 Da |

(c)

| Tool | Fragmentation method | Enzyme specificity | MS1 tolerance | MS2 tolerance |
|---|---|---|---|---|
| UniNovo | CID | AspN | 20 ppm | 0.5 Da |
| PepNovo+ | CID | None | 0.02 Da | 0.5 Da |
| PEAKS | CID | AspN | 20 ppm | 0.5 Da |

(d)

| Tool | Fragmentation method | Enzyme specificity | MS1 tolerance | MS2 tolerance |
|---|---|---|---|---|
| UniNovo | ETD | Trypsin | 20 ppm | 0.5 Da |
| PEAKS | ETD | Trypsin | 20 ppm | 0.5 Da |

(e)

| Tool | Fragmentation method | Enzyme specificity | MS1 tolerance | MS2 tolerance |
|---|---|---|---|---|
| UniNovo | ETD | LysC | 20 ppm | 0.5 Da |
| PEAKS | ETD | LysC | 20 ppm | 0.5 Da |

(f)

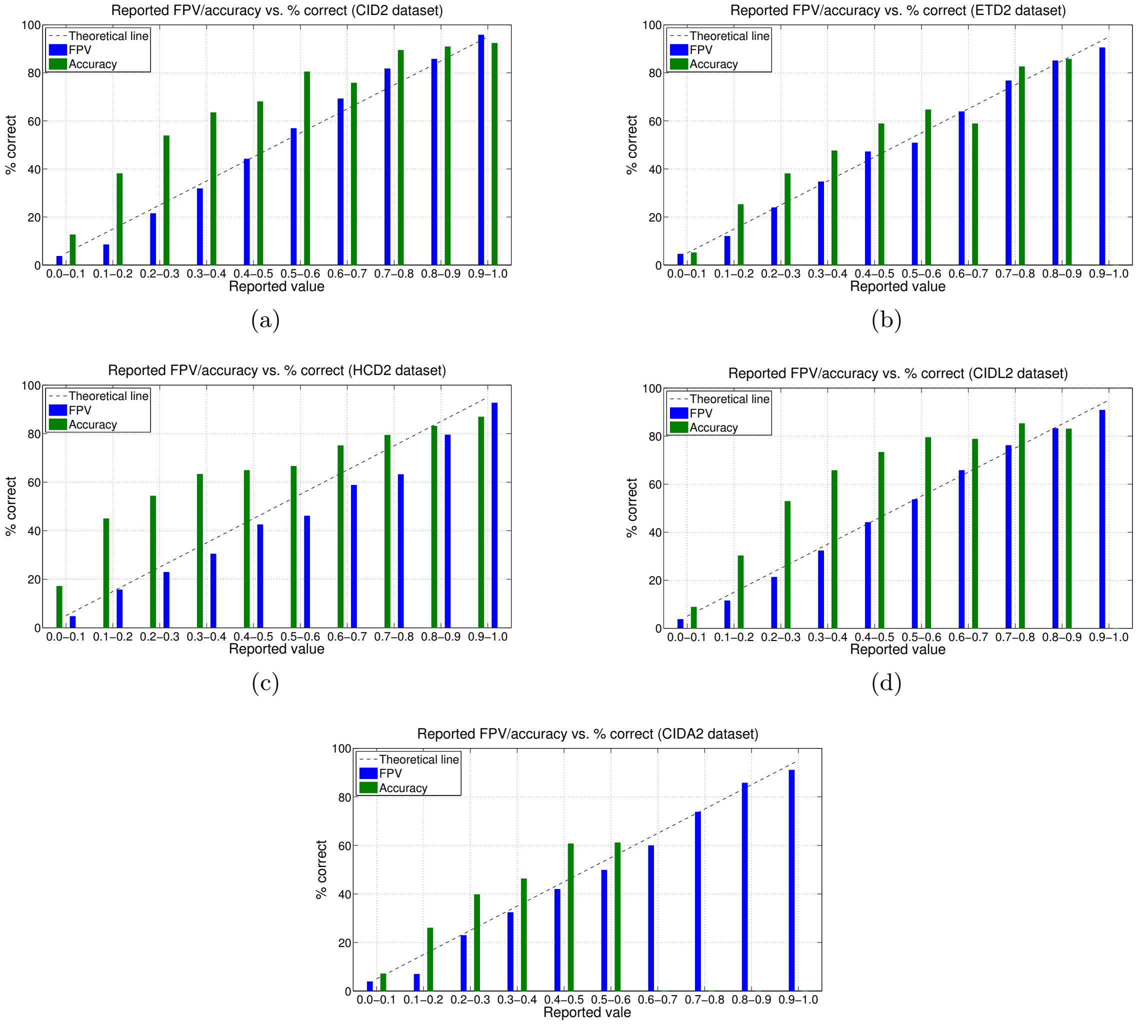
| Tool | Fragmentation method | Enzyme specificity | MS1 tolerance | MS2 tolerance |
|---|---|---|---|---|
| UniNovo | ETD | AspN | 20 ppm | 0.5 Da |
| PEAKS | ETD | AspN | 20 ppm | 0.5 Da |

(g)

| Tool | Fragmentation method | Enzyme specificity | MS1 tolerance | MS2 tolerance |
|---|---|---|---|---|
| UniNovo | HCD | Trypsin | 20 ppm | 20 ppm |
| PepNovo+ | HCD | Trypsin | 0.02 Da | 0.02 Da |
| pNovo | HCD | Trypsin | 20 ppm | 20 ppm |

**Table A3.** The parameters of UniNovo, PepNovo+, PEAKS, and pNovo for different datasets: (a) CID2 (b) CIDL2 (c) CIDA2 (d) ETD2 and ETD3 (e) ETDL3 and ETDL4 (f) ETDA3 and ETDA4 (g) HCD2 and HCD3. For all tools, the carbamidomethylation of Cys (C+57) was set as a fixed modification.

## A9 The estimation results for the $\boldsymbol{FPV_i}$ and the accuracy



(a)

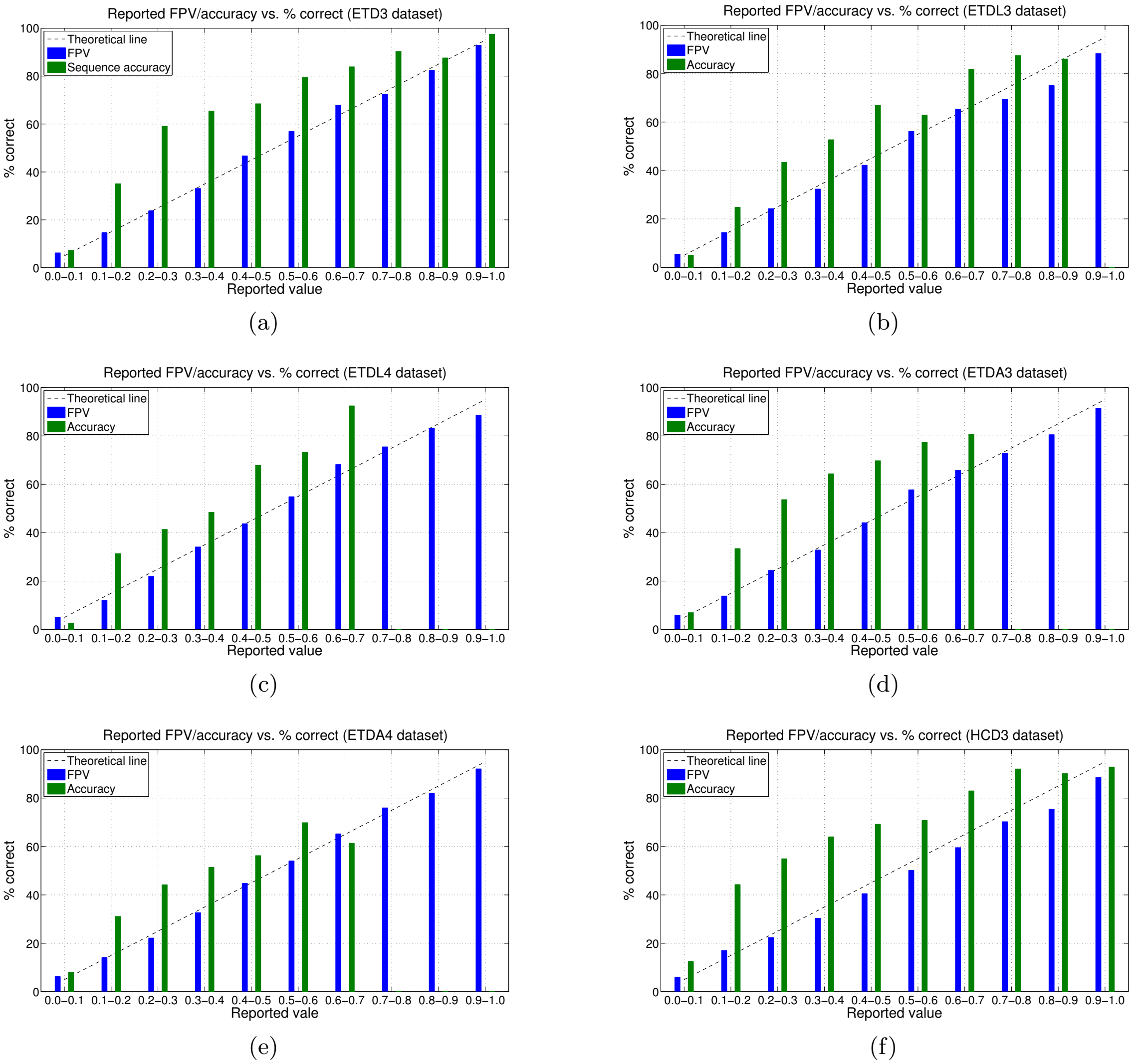
(b)

(c)

(d)

(e)

**Fig. A2.** The estimation results for the $FPV_i$ and the accuracy. The $x$-axis is the range of the reported $FPV_i$ of a mass or the accuracy of a reconstruction, and the $y$-axis is the percentage that the corresponding mass is a fragmentation site or the corresponding reconstruction is correct. Three different types of spectra are tested: (a) CID trypsin charge 2 (b) ETD trypsin charge 2 (c) HCD trypsin charge 2 (d)CID LysC charge 2 (e) CID AspN charge 2 (see Results section for the dataset description). Each dataset consists of 1,000 annotated spectra from distinct peptides (identified by MS-GFDB [8] with the peptide level FDR $< 1\%$) . Some of accuracies (green bars) are not drawn because the sample numbers were too small ($< 50$). $FPV_i$ follows the empirical probability closely (within 5% error). The accuracy tends to slightly underestimate the actual probability, which means that the estimation is conservative.



(a)

(b)

(c)

(d)

(e)

(f)

**Fig. A3.** Figure A2. continued. (a) ETD trypsin charge 3 (b) ETD LysC charge 3 (c) ETD LysC charge 4 spectra (d) ETD AspN charge 3 (e) ETD AspN charge 4 spectra (f) HCD trypsin charge 3.

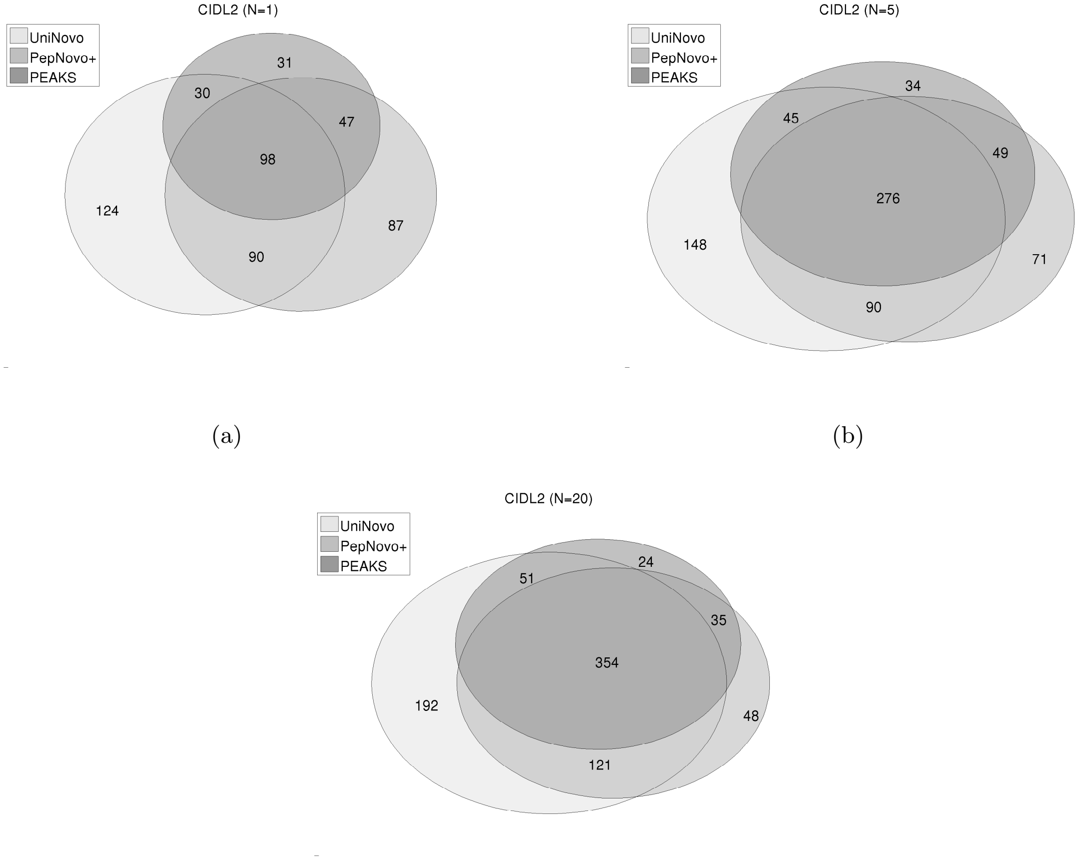
## A10 The datasets

**CID, ETD, and CID/ETD datasets:** CID, ETD, and CID/ETD datasets contain LTQ-Orbitrap spectra (Thermo Fisher Scientific) of trypsin digested peptides from the human HEK293 cell line generated in Albert Heck's laboratory (see [8] for details). The original dataset described in [8] contains the CID/ETD spectral pairs. To obtain the CID dataset, we took only CID spectra from the original dataset and identified them using MS-GFDB at 1% FDR. Out of the identified CID spectra, we randomly pick 1,000 doubly charged spectra that represent distinct tryptic peptides. ETD dataset was generated similarly, and it consists of 1,000 doubly and 1,000 triply charged ETD spectra of distinct tryptic peptides. CID/ETD dataset contains 1,000 pairs of doubly charged and 1,000 pairs of triply charged CID/ETD spectra of distinct tryptic peptides.

**CIDL, CIDA, ETDL, and ETDA datasets:** To benchmark UniNovo on spectra of non-tryptic peptides, we analyzed 4 spectral datasets generated in Joshua Coon's laboratory (see [10] for details). From yeast protein samples, the authors in [10] generated CID and ETD spectra of LysC or AspN digested peptides on a hybrid linear ion trap-orbitrap mass spectrometer (Thermo Fisher Scientific). From the identified CID spectra of LysC (AspN) digested peptides, we randomly pick 1,000 doubly charged spectra representing distinct peptides to generate CIDL (CIDA) dataset. In case of ETD spectra, we selected 1,000 charge 3 and 1,000 charge 4 spectra representing distinct LysC (AspN) digested peptides to generate ETDL (ETDA) dataset.

**HCD dataset:** To generate HCD dataset, we used HCD spectra reported by [3]. The original spectra were acquired by LTQ-Orbitrap Velos (Thermo Fisher Scientific) using one of three different fragmentation methods (CID, ETD, and HCD) from trypsin digested peptides of HEK293 whole cell lysates. We took only the HCD spectra from the original dataset and identified them using MS-GFDB. Out of all identified spectra, we randomly sampled 1,000 doubly charged and 1,000 triply charged spectra of distinct tryptic peptides.

## A11 Venn diagrams of the number of correctly sequenced spectra



(a)

(b)

(c)

**Fig. A4.** The Venn diagrams of the number of correctly sequenced spectra for CIDL2 dataset. (a) $N = 1$ (b) $N = 5$ (c) $N = 20$

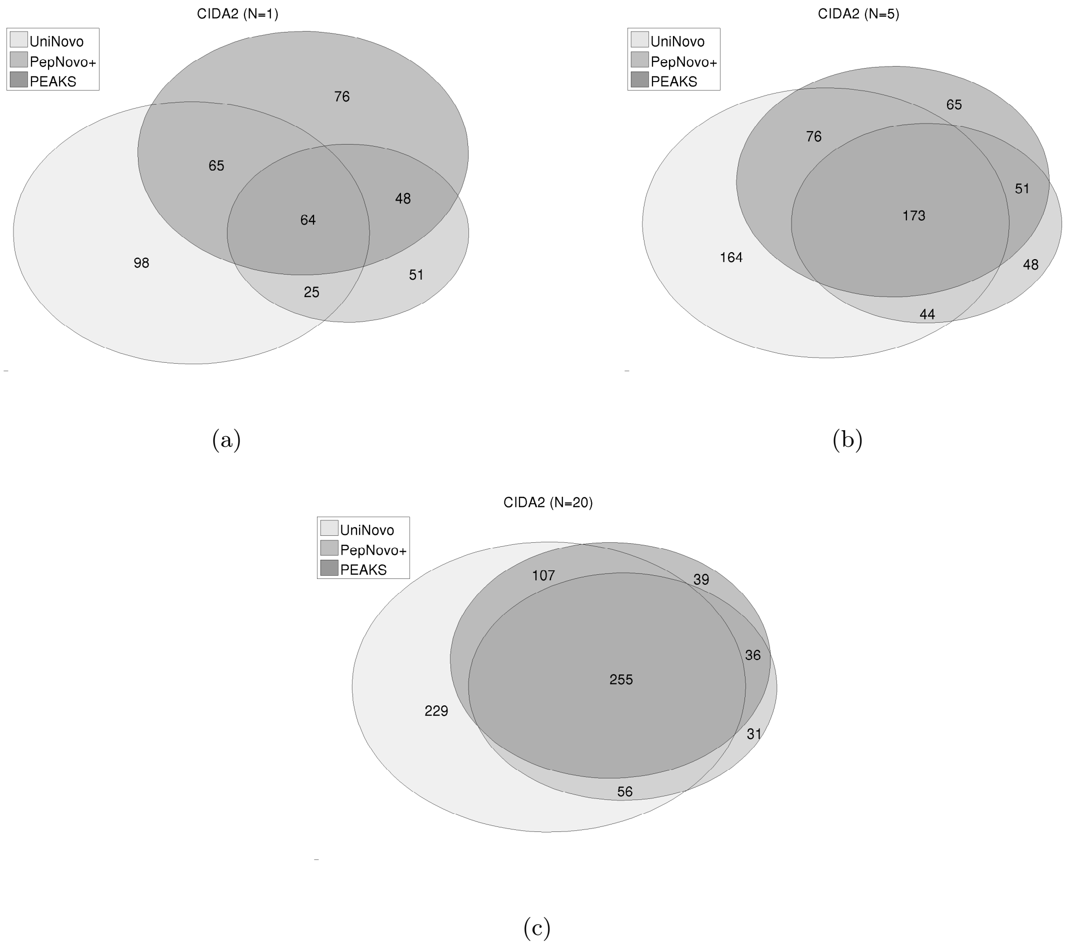
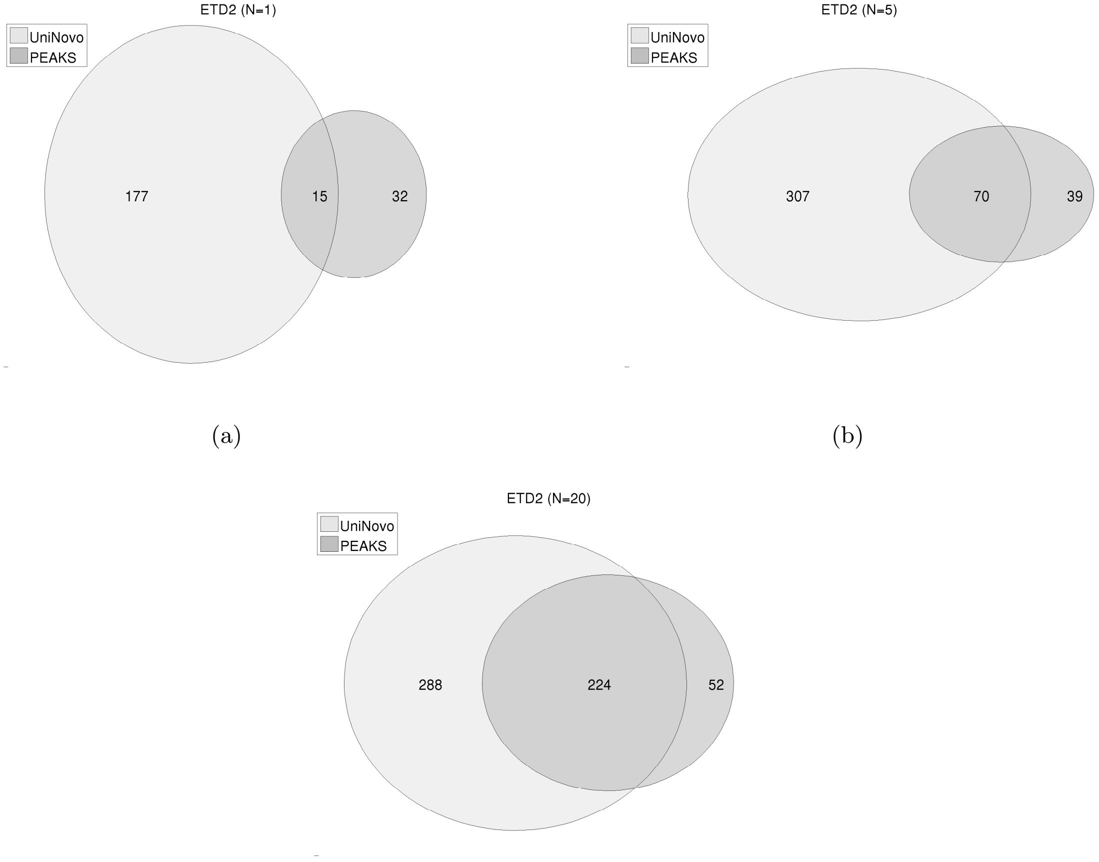


**Fig. A5.** The Venn diagrams of the number of correctly sequenced spectra for CIDA2 dataset. (a) $N = 1$ (b) $N = 5$ (c) $N = 20$

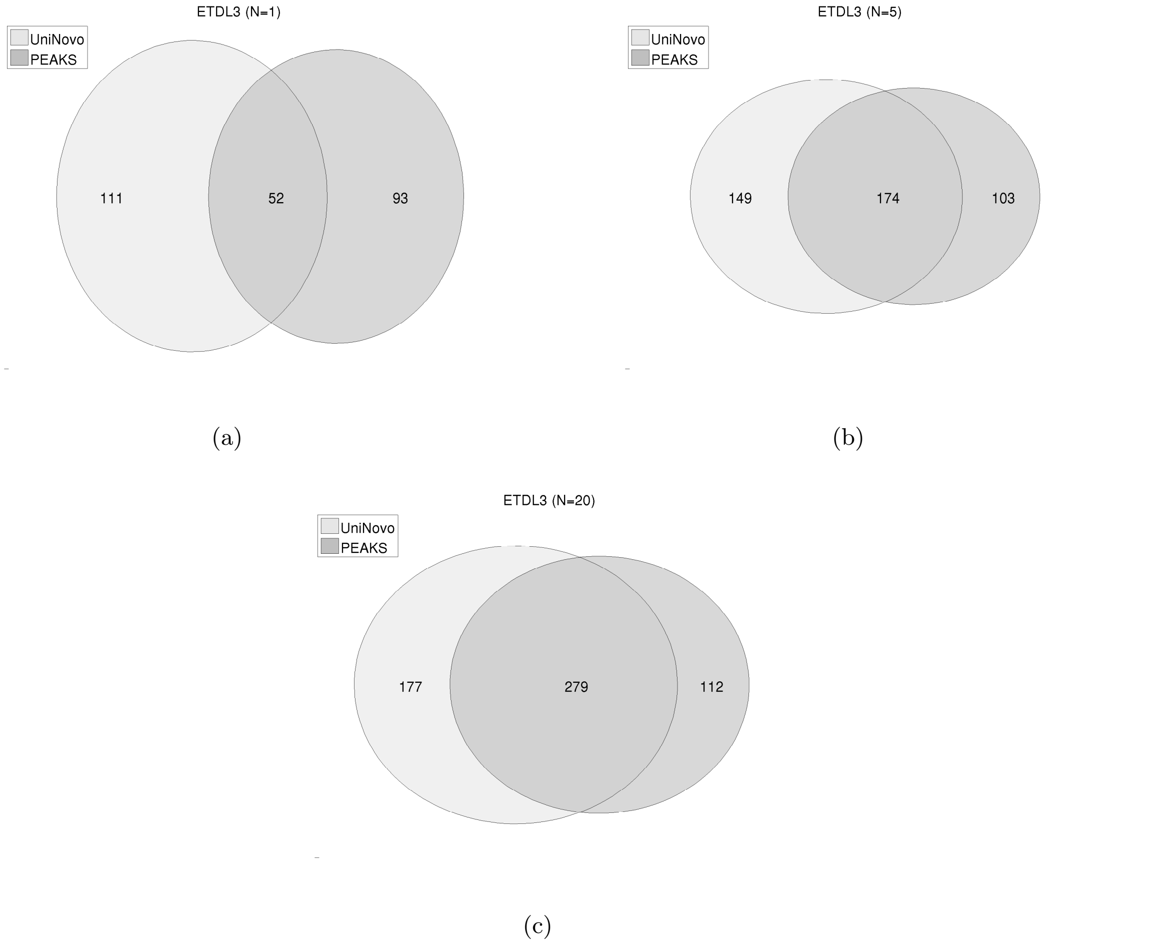


(a)

(b)

(c)

**Fig. A6.** The Venn diagrams of the number of correctly sequenced spectra for ETD2 dataset. (a) $N = 1$ (b) $N = 5$ (c) $N = 20$

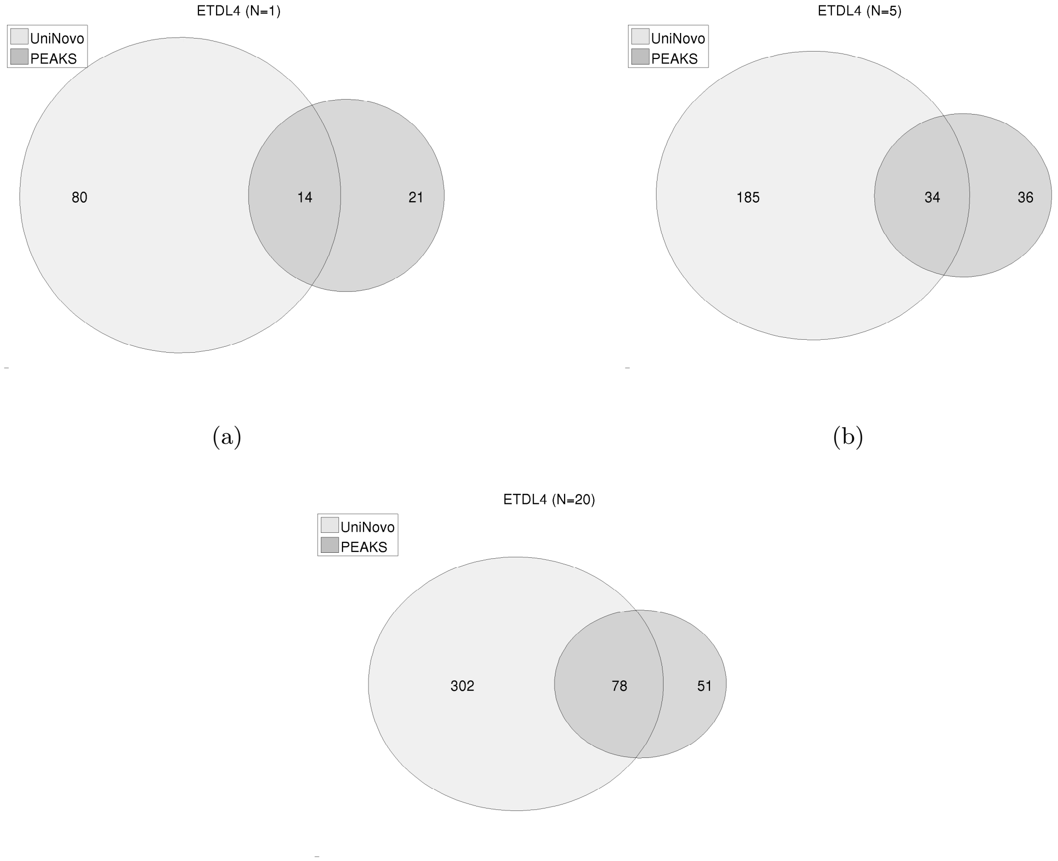


(a)

(b)

(c)

**Fig. A7.** The Venn diagrams of the number of correctly sequenced spectra for ETDL3 dataset. (a) $N = 1$ (b) $N = 5$ (c) $N = 20$

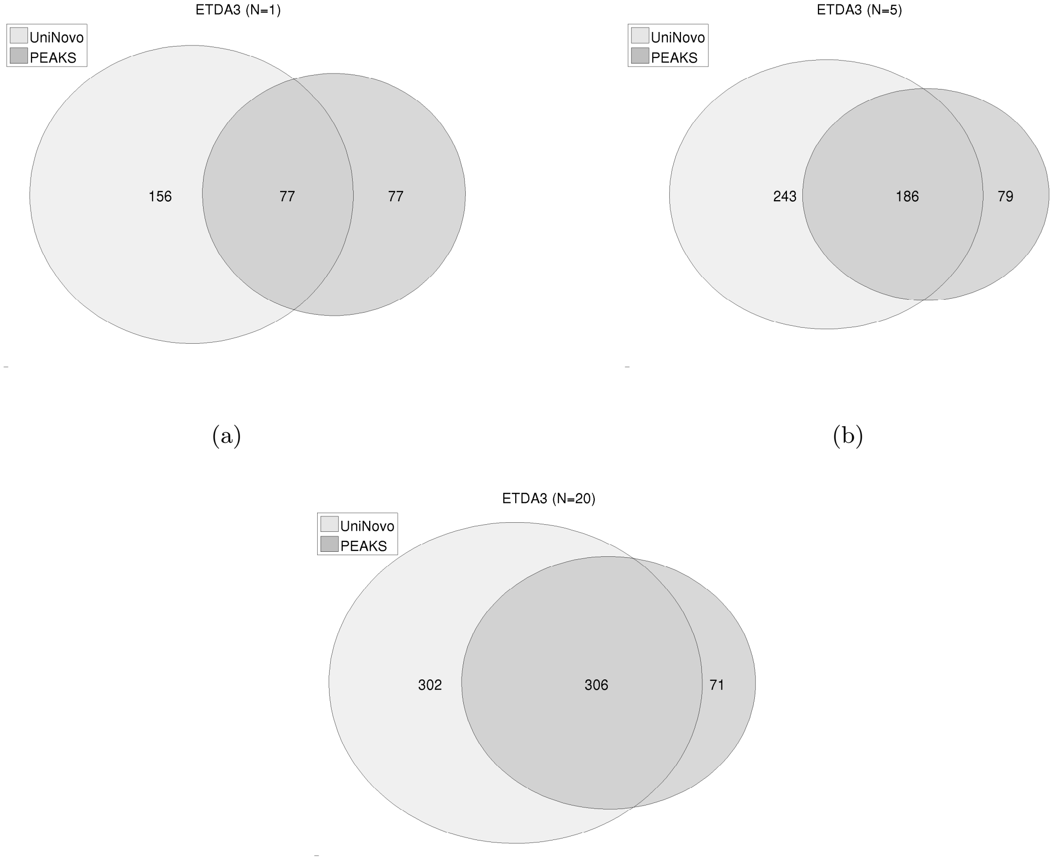


(a)

(b)

(c)

**Fig. A8.** The Venn diagrams of the number of correctly sequenced spectra for ETDL4 dataset. (a) $N = 1$ (b) $N = 5$ (c) $N = 20$

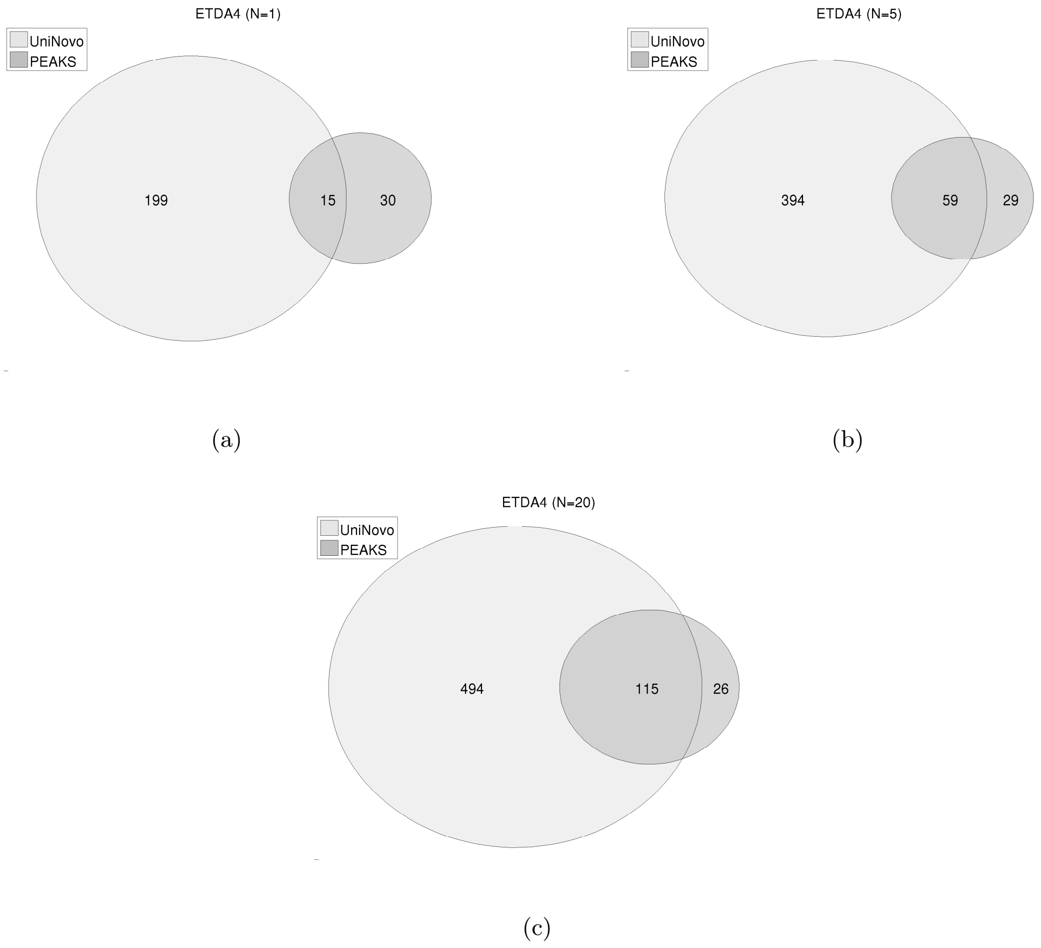


(a)

(b)

(c)

**Fig. A9.** The Venn diagrams of the number of correctly sequenced spectra for ETDA3 dataset. (a) $N = 1$ (b) $N = 5$ (c) $N = 20$

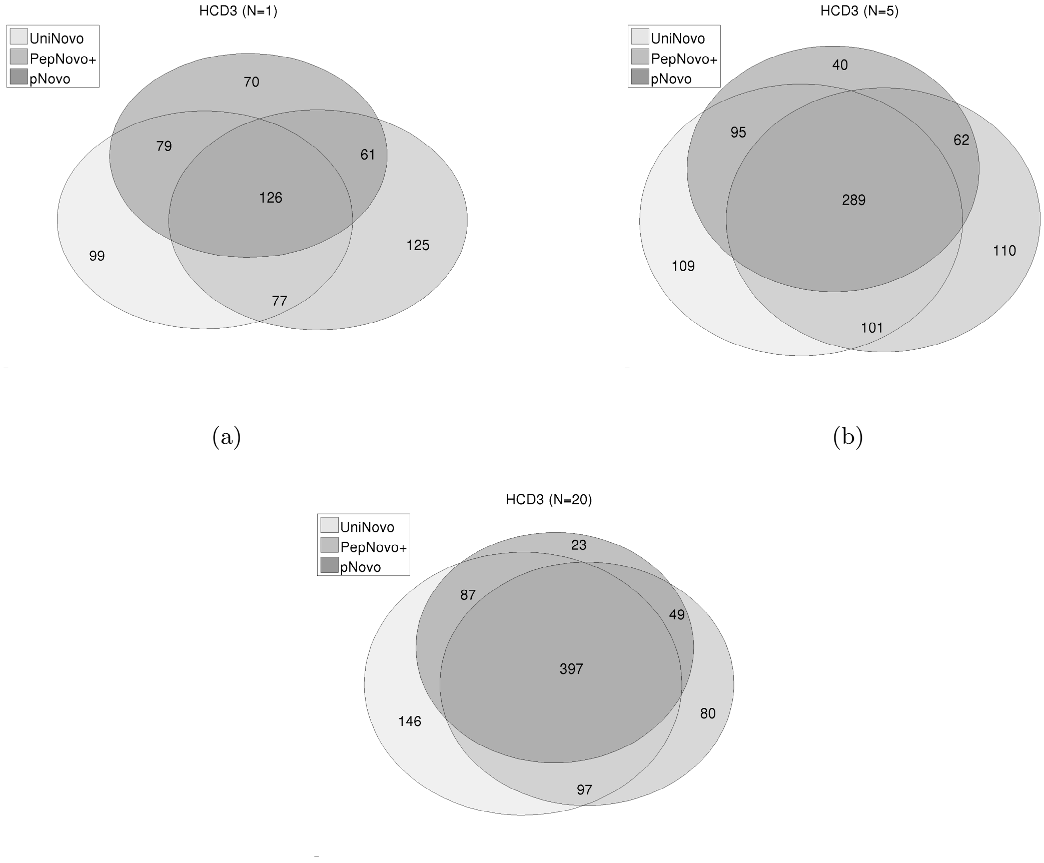


(a)

(b)

(c)

**Fig. A10.** The Venn diagrams of the number of correctly sequenced spectra for ETDA4 dataset. (a) $N = 1$ (b) $N = 5$ (c) $N = 20$



(a)

(b)

(c)

**Fig. A11.** The Venn diagrams of the number of correctly sequenced spectra for HCD3 dataset. (a) $N = 1$ (b) $N = 5$ (c) $N = 20$

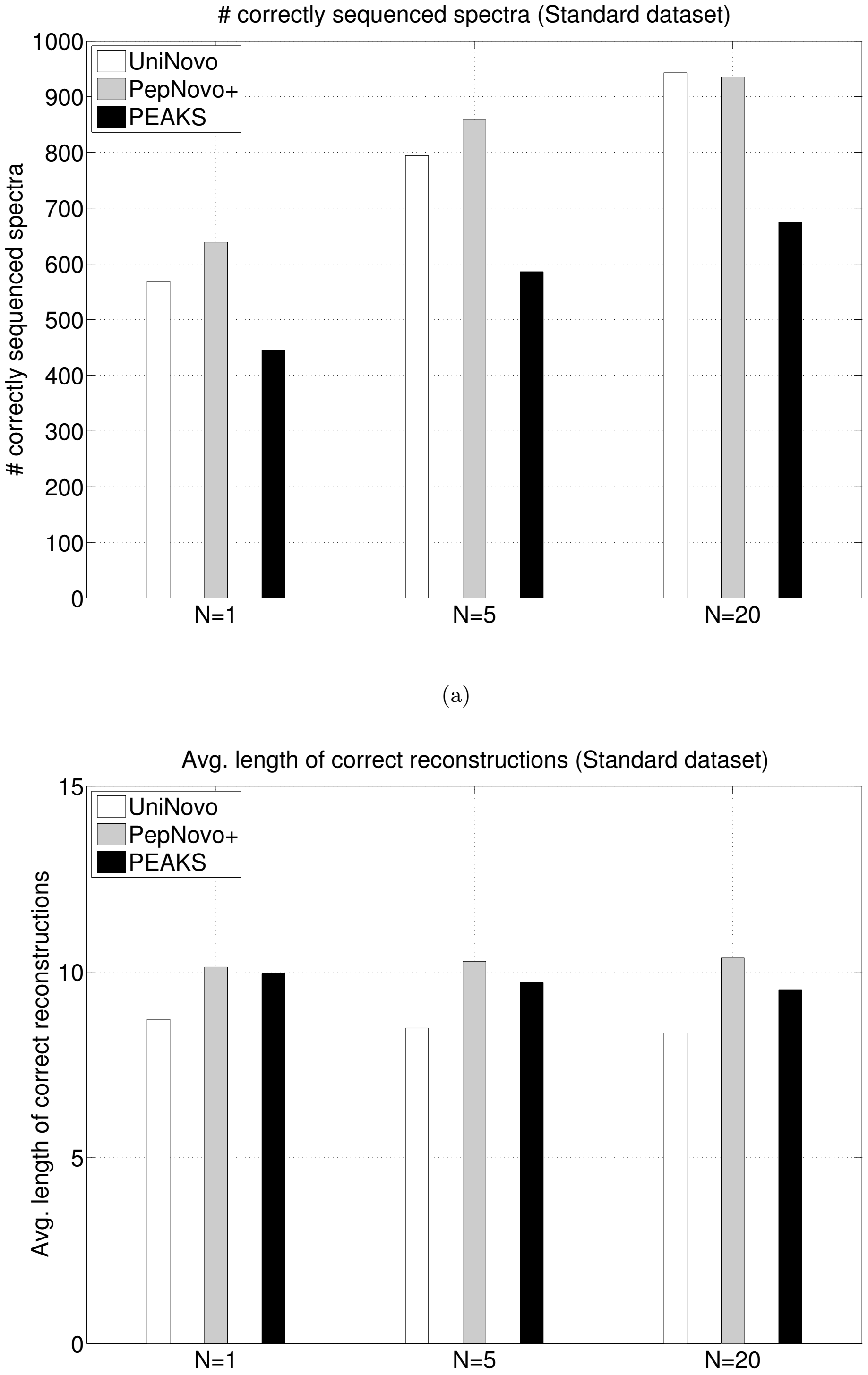
## A12 Analysis of Standard dataset

### A12.1 Standard dataset

Since all the spectra in the datasets in Table 1 are identified by a single search engine (i.e., MS-GFDB), the experimental results from those datasets may be positively/negatively biased toward specific tools tested. Thus, we benchmarked UniNovo, PepNovo+, and PEAKS using an additional dataset (named as *Standard* dataset) reported in [7] containing spectra identified by Sequest [2] and PeptideProphet [6]. Standard dataset contains 1,388 doubly charged CID spectra (generated by Thermo Electron LTQ) of distinct peptides collected from the Standard Protein Mix database [9]. As mentioned in [7], the parent masses of the spectra in Standard dataset were corrected according to the Sequest identifications; the spectra have high resolution MS1 and low resolution MS2.

### A12.2 Results

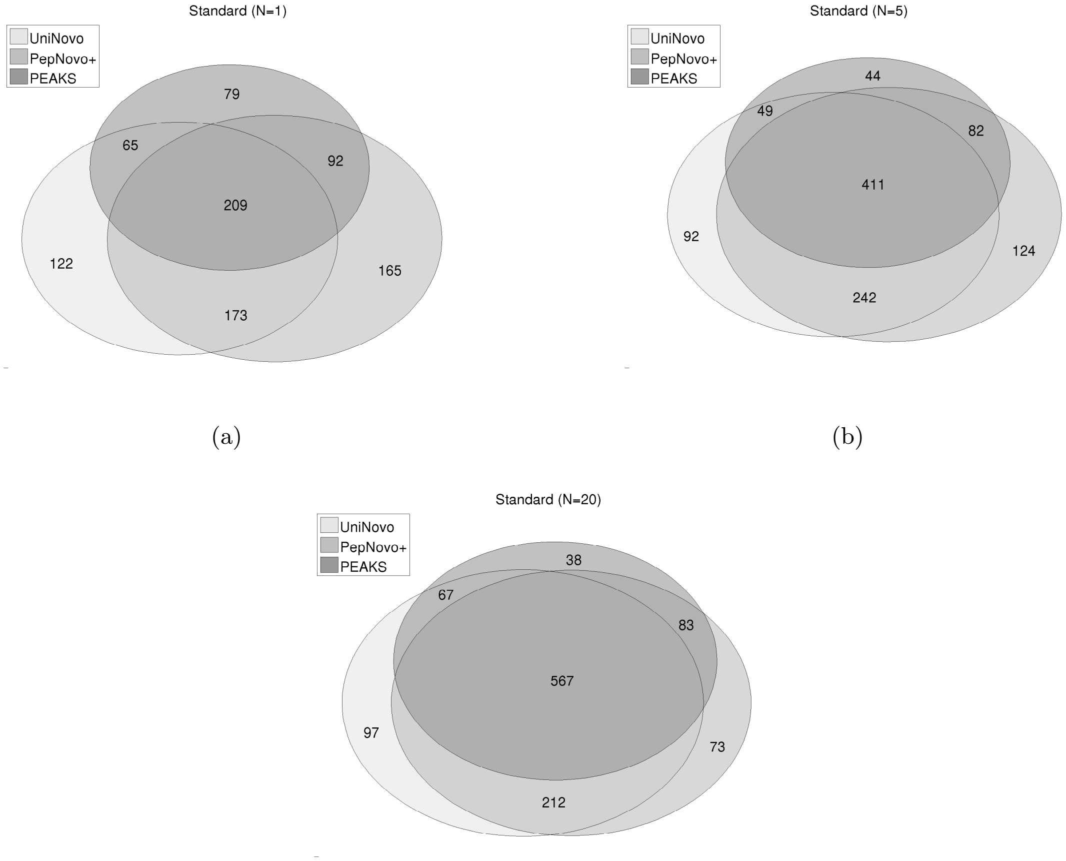
For Standard dataset, we measured the number of correctly sequenced spectra and the average length of the correct reconstructions for $N = 1, 5$, and 20. All experimental parameters for Standard dataset were the same as for CID2 dataset, except that non-enzyme specificity was used for PEAKS (because many of spectra in Standard dataset are from peptides with non-enzymatic cleavages). The results obtained from Standard dataset were compared with those from CID2 dataset because both datasets contain doubly charged CID spectra. Figure A12 shows the results from Standard dataset. Similar to the results from CID2 dataset, UniNovo found slightly more correctly sequenced spectra when $N = 20$ and slightly less when $N = 1, 5$ than PepNovo+ (Figure A12 (a)). PEAKS found the smallest number of correctly sequenced spectra. Figure A12 (b) shows that the length of correct reconstructions for PepNovo+ was longer than UniNovo or PEAKS as in CID2 dataset. We also drew the Venn diagrams of the correctly sequenced spectra for Standard dataset in Figure A13. The Venn diagrams for Standard dataset and CID2 dataset (Figure 3 (a)-(c)) had similar percentages of overlaps of spectra ($20.3\%, 36.5\%, 44.3\%$ vs. $23.2\%, 45.2\%, 54.3\%$ for $N = 1, 5, 20$) even if the percentages for Standard dataset were smaller than for CID2 by 3-10%. Overall, essentially similar results were obtained for Standard and CID2 datasets, which suggests that no significant bias toward specific tools was introduced in the experiments using the datasets in Table 1.



(a)

(b)

**Fig. A12.** The results from Standard dataset (a) the number of correctly sequenced spectra (b) the average length of correct reconstructions



(a)

(b)

(c)

**Fig. A13.** The Venn diagrams of the number of correctly sequenced spectra for Standard dataset. (a) $N = 1$ (b) $N = 5$ (c) $N = 20$

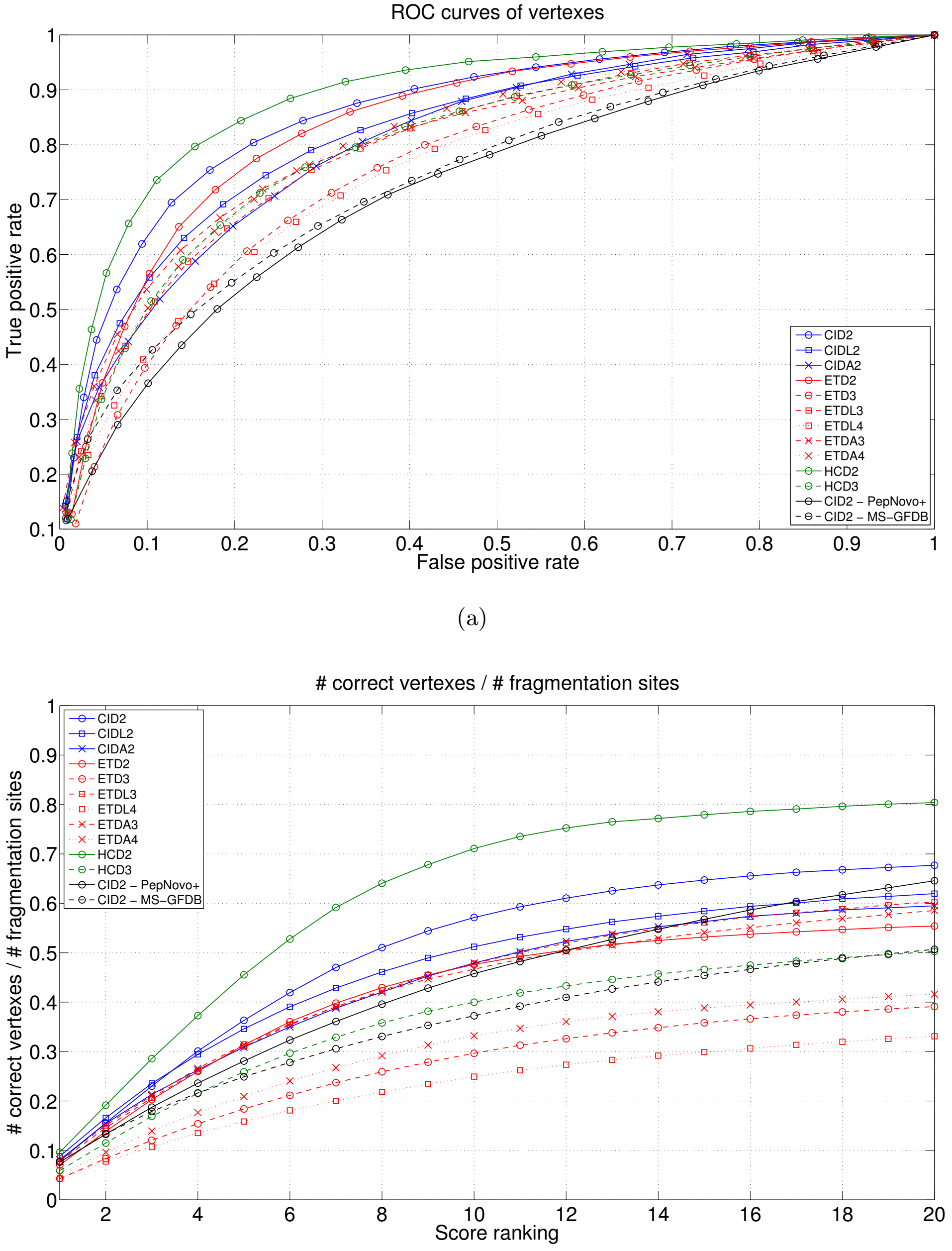
## A13 Evaluation of the spectrum graph for different datasets/tools

To evaluate the spectrum graph of UniNovo for different spectrum types, we plotted ROC (*Receiver Operating Characteristic*) curves of vertices (i.e., plausible fragmentation sites) in the spectrum graphs from each dataset. Given a spectrum graph, we first ranked all vertices in such a way that the $x$th highest scoring vertex has the rank $x$. Then we chose 20 top ranking vertices (excluding source and sink vertices), and calculated true positive rates and false positive rates at various rank thresholds. For a rank threshold $x$, the true positive rate was calculated by # of correct vertices of rank less than $x$ divided by # of correct vertices, and the false positive rate was by # of incorrect vertices of rank less than $x$ divided by # of incorrect vertices.[8]

Figure A14(a) shows the ROC curves for the datasets in Table 1, except the ones of spectral pairs. For CID2 dataset, the ROC curves of spectrum graphs generated by MS-GFDB and PepNovo+ were also drawn. The ROC curve of UniNovo for CID2 dataset (blue circled line) is significantly better than those of PepNovo+ and MS-GFDB (black and black dashed lines). For instance, at the false positive rate of 0.1, the true positive rate of UniNovo was 0.7 while both MS-GFDB's and PepNovo+'s were about 0.4. As the ROC curves suggest, HCD2 (ETDL4) datasets represents the most (the least) suitable datasets for *de novo* sequencing. Other datasets can be ranked as: HCD2 (the best)$\rightarrow$CID2 $\rightarrow$ ETD2 $\rightarrow$ CIDL2 $\approx$ CIDA2 $\approx$ ETDL3 $\approx$ ETDA3 $\approx$ ETDA4$\approx$ HCD3 $\rightarrow$ ETD3 $\approx$ ETDL4 (the worst).

The above ROC curve evaluates the sensitivity/specificity of the scoring functions with 20 highest ranking vertices in the spectrum graph. However, if only few of the 20 vertices are correct - in other words, most fragmentation sites are not selected for 20 vertices - such an evaluation may be pointless. Thus, we also measured the fraction of all fragmentation sites that are actually included in the correct vertices of rank less than $x$ (i.e., the number of correct vertices of rank less than $x$ divided by the number of all fragmentation sites). The same measurement was done for CID2 dataset by PepNovo+ and MS-GFDB. Figure A14 (b) shows that UniNovo (blue circled line) correctly detected 20% and 40% more fragmentation sites within top 20 vertices than PepNovo+ and MS-GFDB (black and black dashed lines), respectively. Together with the ROC comparison, one can deduce that UniNovo detects more fragmentation sites and scores them more specifically than PepNovo+ or MS-GFDB. Also one can infer that the good performance of PepNovo+ shown in Figure 2 is obtained by reranking of the reconstructions using the sequence specific features. From Figure A14 (b), we can evaluate each dataset in terms of the fraction of correctly predicted fragmentation sites as: HCD2 (the best)$\rightarrow$CID2 $\rightarrow$ CIDL2 $\rightarrow$ CIDA2 $\approx$ ETD2 $\approx$ ETDL3 $\approx$ ETDA3 $\rightarrow$ HCD3 $\rightarrow$ ETDA4 $\rightarrow$ ETD3 $\rightarrow$ ETDL4 (the worst).

[8] The error tolerance for vertices was set to 0.5 Da except for HCD2 and HCD3 datasets. For HCD2 and HCD3 datasets, the error tolerance was set to 20 ppm.



(a)

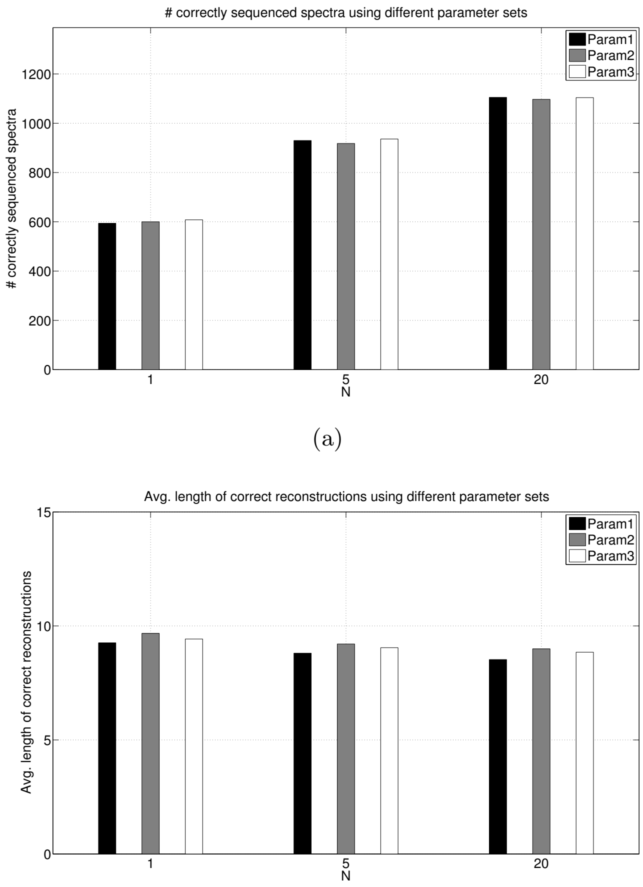
(b)

**Fig. A14.** (a) ROC curves of vertices (i.e., plausible fragmentation sites) in the spectrum graphs. Per each spectrum graph, the vertices are ranked by their scores so that the $x$th highest scoring vertex has the rank $x$. We took 20 highest ranking vertices per each spectrum graph, and calculated the true positive rate and the false positive rate. Given a rank threshold $x$, the true (false) positive rate is given by # of correct (incorrect) vertices of rank less than $x$ divided by # of correct (incorrect) vertexes. Using UniNovo, ROC curves for the datasets in Table 1 (except the ones of spectral pairs) were generated. We also generated ROC curves using PepNovo+ (black line) and MS-GFDB (black dashed line) for CID2 dataset.

(b) The fraction of correctly predicted fragmentation sites. Given a rank threshold $x$, we measured what fraction of all fragmentation sites are included in the correct vertices of rank less than $x$.

## A14 How many PSMs (i.e., identified spectra) are required to train UniNovo

Even if a relatively small number of PSMs are required to train UniNovo, still there should be a sufficient number of PSMs in the training dataset to avoid possible overfitting. To see how many PSMs are required for training of UniNovo, we generated three different training datasets, each of which was formed by randomly selecting 5,000 PSMs in the training dataset (consisting of about 1,9000 PSMs) used to train the parameters for CID tryptic doubly charged spectra. From the generated training datasets, we learned three different parameter sets. Then for CID2 dataset, we repeated the experiments to measure the number of correctly sequenced spectra and the average length of correct reconstructions using each parameter set. Figure A15 shows the results for $N = 1, 5$, and 20. Figure A15 (a) illustrates that the number of correctly sequenced spectra for the different parameter sets were almost the same each other, with the maximum difference of 14. The maximum difference of the average length of correct reconstructions was only 0.5, suggesting that overfitting did not occur. When we use a smaller number of PSMs (about 4,000), the results started to diverge throughout different parameter sets (data not shown). Therefore, we recommend to use at least 5,000 PSMs (per charge state) in the training dataset to avoid overfitting.



(a)

(b)

**Fig. A15.** The number of correctly sequenced spectra (a) and the average length of correct reconstructions (b) for CID2 dataset using differnt parameter sets learned from different training datasets consisting of 5,000 PSMs each. Even though a small number of PSMs were used to train each parameter set, the results are consistent each other suggesting that overfitting did not occur.

## References

1. V. Dancik, T. A. Addona, K. R. Clauser, J. E. Vath, and P. A. Pevzner. De novo peptide sequencing via tandem mass spectrometry. *Journal of Computational Biology*, 6(3-4):327–342, 1999.
2. J. K. Eng, A. L. McCormack, and J. R. Yates. An approach to correlate tandem mass spectral data of peptides with amino acid sequences in a protein database. *Journal of the American Society for Mass Spectrometry*, 5(11):976–989, 1994.
3. C. K. Frese, A. F. M. Altelaar, M. L. Hennrich, D. Nolting, M. Zeller, J. Griep-Raming, A. J. R. Heck, and S. Mohammed. Improved peptide identification by targeted fragmentation using CID, HCD and ETD on an LTQ-Orbitrap velos. *J. Proteome Res.*, 10(5):2377–2388, 2011.
4. Y. Huang, J. M. Triscari, G. C. Tseng, L. Pasa-Tolic, M. S. Lipton, R. D. Smith, and V. H. Wysocki. Statistical characterization of the charge state and residue dependence of low-energy CID peptide dissociation patterns. *Analytical Chemistry*, 77(18):5800–5813, 2005.
5. D. Hunter. An upper bound for the probability of a union. *Journal of Applied Probability*, 13(3):597–603, 1976.
6. A. Keller, A. Nesvizhskii, E. Kolker, and R. Aebersold. Empirical statistical model to estimate the accuracy of peptide identifications made by ms/ms and database search. *Anal. Chem.*, 74:5383–92, Jan 2002.
7. S. Kim, N. Bandeira, and P. A. Pevzner. Spectral profiles, a novel representation of tandem mass spectra and their applications for de novo peptide sequencing and identification. *Molecular & Cellular Proteomics*, 8(6):1391–1400, 2009.
8. S. Kim, N. Mischerikow, N. Bandeira, J. D. Navarro, L. Wich, S. Mohammed, A. J. R. Heck, and P. A. Pevzner. The generating function of CID, ETD, and CID/ETD pairs of tandem mass spectra: Applications to database search. *Molecular & Cellular Proteomics*, 9(12):2840 –2852, 2010.
9. J. Klimek, J. S. Eddes, L. Hohmann, J. Jackson, A. Peterson, S. Letarte, P. R. Gafken, J. E. Katz, P. Mallick, H. Lee, A. Schmidt, R. Ossola, J. K. Eng, R. Aebersold, and D. B. Martin. The standard protein mix database: A diverse data set to assist in the production of improved peptide and protein identification software tools. *Journal of Proteome Research*, 7(1):96–103, 2008.
10. D. L. Swaney, C. D. Wenger, and J. J. Coon. Value of using multiple proteases for Large-Scale mass Spectrometry-Based proteomics. *J. Proteome Res.*, 9(3):1323–1329, 2010.
11. V. H. Wysocki, G. Tsaprailis, L. L. Smith, and L. A. Breci. Mobile and localized protons: a framework for understanding peptide dissociation. *Journal of Mass Spectrometry*, 35(12):1399–1406, 2000.