Advancing De Novo Glycopeptide Sequencing with InstaNovo in Glycoproteomics

Isaac Henri Joël Houngue (hjisaach@aims.ac.za)
African Institute for Mathematical Sciences (AIMS)

Supervised by: Mr Jeroen Van Goey
InstaDeep, South Africa
Supervised by: InstaNovo Team

12 June 2025
Submitted in partial fulfillment of a structured masters degree at AIMS South Africa

Abstract

This study investigates the adaptation of InstaNovo, a transformer-based model originally designed for de novo peptide sequencing, to the more complex domain of glycoproteomics. Through a series of fine-tuning experiments on glycopeptide datasets, we observe that while the model shows some capacity to learn from glyco spectra, the overall improvements remain limited. Notably, fine-tuning on unfiltered spectra, which partially overlap with a dataset the model was trained on, results in stronger learning signals. However, across all fine-tuning settings, the model suffers from catastrophic forgetting, losing accuracy on peptide sequences it previously handled well. Even when using unfiltered spectra, which provide stronger learning signals for glycopeptides, this forgetting effect persists. PCA (Principal Component Analysis) projections of spectrum embeddings further reveal a significant domain shift, which helps explain why InstaNovo struggles on glycopeptides sequencing adaptation. To address this, we propose treating overlapping, glyco spectra not as outliers to discard, but as informative examples that help the model distinguish between glycosylated and non-glycosylated peptides. While fine-tuning strategies prove insufficient for reliable glycopeptide sequencing, our results highlight promising directions for improvement. Future work should explore richer datasets, multi-task learning, attention-guided learning mechanism, contrastive learning, and transformations that preserve the embedding of the original InstaNovo spectra while shifting glyco spectra closer to them in embedding space, helping align their distributions.

Declaration

I, the undersigned, hereby declare that the work contained in this research project is my original work, and that any work done by others or by myself previously has been acknowledged and referenced accordingly.

Isaac Henri Joël Houngue, 12 June 2025

Contents

1. Introduction

1.1 Context

Proteins are fundamental macromolecules present in all living organisms. They are long chains of amino acids and play critical roles in diverse biological processes, including metabolism, cell cycle, and cellular signaling (Gasser, 2023; Zhang and Wang, 2023). They repair and build tissues, including muscles, skin, bones, and organs, and play a central role in maintaining and restoring the body during illness, injury, or physical stress. Proteins regulate essential metabolic reactions by acting as enzymes and hormones, support immune defenses through antibodies, help transport oxygen and nutrients, maintain fluid balance, and, when needed, serve as an alternative energy source (Diker, 2022; Paidikondala and Valivarthi, 2024).

While this demonstrates the importance of proteomics (the study of proteins, including their structure, function, interactions, and modifications within the body), determining a protein's primary structure, the precise sequence of amino acids it is made of (Figure 1.1), remains the essential first step. This primary structure lays the groundwork for deeper insights into many biological processes and diseases (LaPelusa and Kaushik, 2022; Cao, 2023).

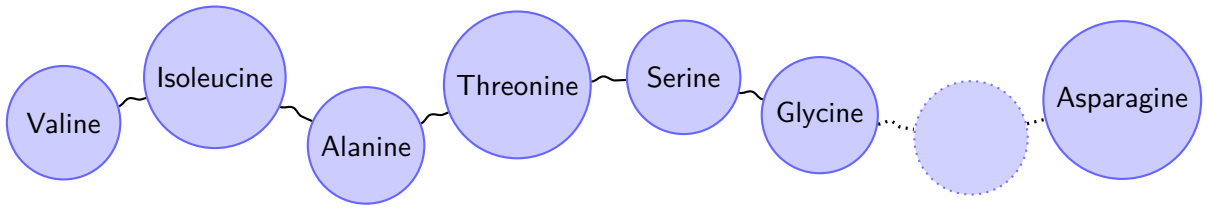


Figure 1.1: Illustration of a protein's primary structure, showing a linear sequence of amino acids. The ellipsis represents continuation of the sequence. Proteins typically range between 100-1,000 amino acids, though some are shorter than 100 amino acids and others approach 2,000 amino acids in rare cases (Nevers et al., 2023). Each amino acid has a letter code, and a sequence is usually represented as letters. For example, the sequence in the image is VIATSG...N, where ... replaces the ellipsis part of the sequence (IUPAC-IUB, 1972).

The mass spectrometry (MS) based bottom-up approach, one of the most commonly used methods in proteomics, begins by enzymatically digesting proteins into smaller fragments called **peptides**, typically using an enzyme called trypsin. These peptides are then introduced into a mass spectrometer, which measures their mass to charge ratio (m/z), and produces a first MS spectrum (also called MS1). Some of these peptides, usually the most intense, are then selected and further fragmented within the same instrument, resulting in a secondary spectrum known as MS2 (or MS/MS), where the peaks represent fragment ions with specific m/z ratios and relative intensities (Gillet et al., 2016; Dupree et al., 2020). The pattern of these peaks can be interpreted by peptide identification tools, also called proteomics search engines, to determine the peptides' amino acid sequence, providing direct evidence of the proteins' primary structures (Huang et al., 2012; Dupree et al., 2020)[1]. These search engines typically match experimental spectra against a reference proteome database. While effective in many scenarios, this reliance on known databases poses challenges when the reference is incomplete or unavailable. In such

[1]The approach is called bottom-up because it begins with small protein fragments (peptides) rather than analyzing the entire intact protein.

cases, *de novo* peptide sequencing, the process of reconstructing peptide sequences directly from spectral data without relying on a database, becomes essential.

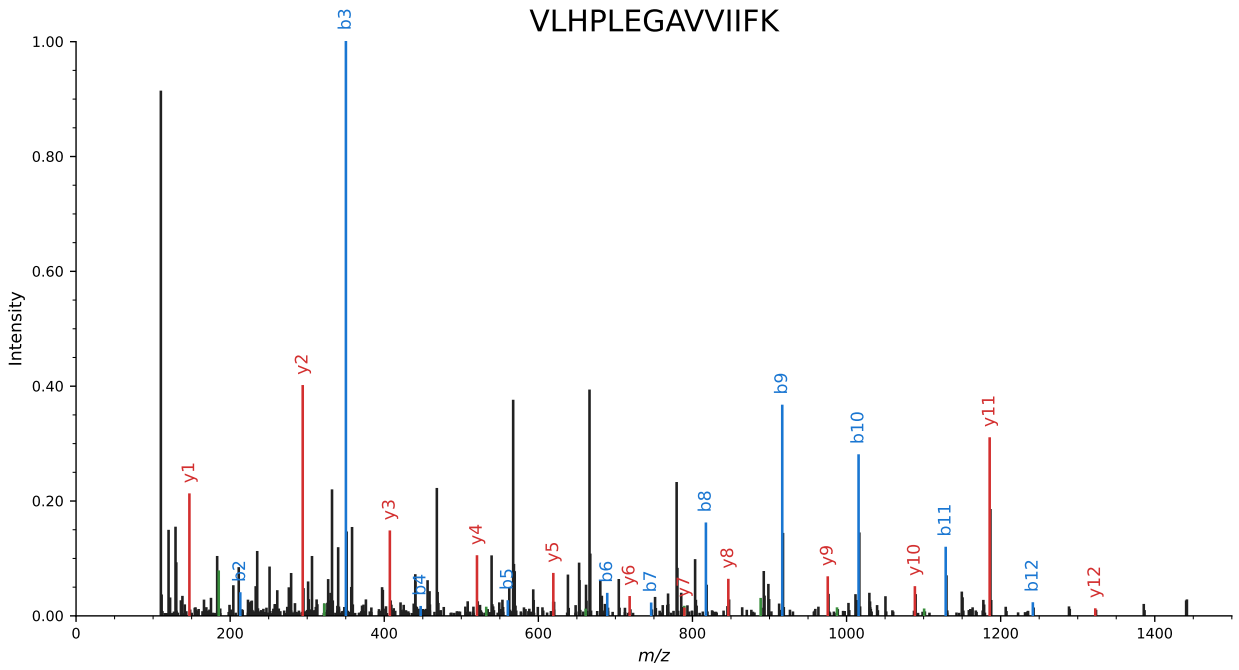


Figure 1.2: Example of MS2 spectrum displaying fragment ions for the peptide VLHPLEGAVVIIFK

Traditional *de novo* sequencing methods typically reconstruct peptide sequences from MS2 spectra by leveraging computational algorithms (Hughes et al., 2010). These range from simpler approaches that analyze consecutive mass differences between fragment ions to more advanced techniques such as subsequence matching, probabilistic models, graph-based methods and more (DiMaggio and Floudas, 2007; Frank et al., 2007). While effective in many cases, these traditional methods often struggle with noisy, incomplete, or low-quality spectral data, leading to lower accuracy compared to database-driven approaches.

Recent advances in deep learning for peptide sequencing have revolutionized mass spectrometry data analysis in proteomics (Wen et al., 2020; Tariq and Bo, 2025). A prime example is InstaNovo proposed by Eloff et al. (2025), a transformer-based neural network that accurately identifies peptide sequences directly from spectral data without relying on reference datasets.

However, another complexity in the protein space is that after their synthesis, proteins may undergo a diverse range of modifications (Bobalova et al., 2023) called post-translation modifications (PTMs), which would add or remove chemical groups to their amino acids (Goldtzvik et al., 2023). These PTMs can alter a protein's properties, influence its function, and even manifest in the mass spectra of digested peptides through mass shifts. Yet, previous computational approaches have primarily focused on modeling a limited subset of common PTMs, such as methionine oxidation and phosphorylation, leaving more complex or less abundant modifications.

Glycosylation, the enzymatic attachment of glycans (sugar molecules) to amino acids, is one of the most commonly occuring and complex PTMs. It is estimated that around 50% of all cellular proteins are glycosylated (An et al., 2009; Van den Steen et al., 1998; Sebastião et al., 2021). Moreover, aberrant glycosylation is associated with various diseases, including strong links to cancer, inflammation, and

autoimmune disorders (He et al., 2024; Oliveira et al., 2021).

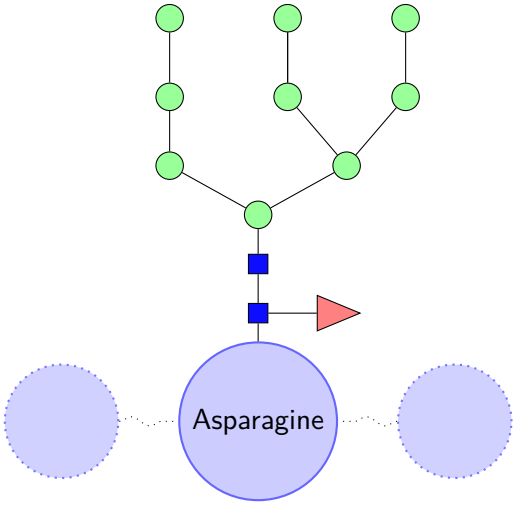


Figure 1.3: A glycan (sugar chain) attached to a protein at an asparagine amino acid through N-glycosylation (Lyons et al., 2015).

The bottom-up approach in glycoproteomics (the study of glycoproteins, including their structures, functions, interactions, and glycan modifications within biological systems) poses additional challenges, including the structural diversity of glycans, with thousands of possible variations, their characteristic mass shifts due to the large mass of these modifications, and the resulting complex patterns of fragment peaks in mass spectra.

Despite the fact that deep learning methods have advanced proteomics via de novo peptide sequencing, their adaptation for glycoproteomics is still unexplored. Fine-tuning Instanovo on glycoproteomics datasets could enable it to expand to identification of glycosylated peptides, with major biological applications.

1.2 Research Objectives

The primary goal of this re search is to explore how well the InstaNovo model can be fine-tuned to glycoproteomics data, and assess its potential for accurate de novo sequencing of glycopeptides. Specifically, this study aims to:

- Adapt InstaNovo for glycopeptide sequencing by fine-tuning it on glycoproteomics datasets to improve its ability to decode peptide sequences with attached glycans.
- Evaluate the refined model's performance in predicting glycopeptides, while also assessing whether it retains, improves, or compromises its original peptide prediction performance.
- Analyze potential reasons for any limitations in performance, using insights gained to confidently guide future refinements toward optimal results.

1.3 Scope and Limitations

The current study establishes a focused framework for glycopeptide sequencing using InstaNovo, with specific boundaries that provide a foundation for future expansion and refinement.

- The study focuses exclusively on **N-linked glycosylation**; O-linked and other forms of glycosylation are beyond the current scope.
- We do not address the challenge of distinguishing **isomeric amino acids** such as isoleucine and leucine, as they share identical masses and are indistinguishable by mass spectrometry alone.
- The **three-dimensional geometry of glycans** and their structural isomers are not considered in the current model.

1.4 Report Structure

The rest of this document is structured as follows:

Chapter 2 provides background on the InstaNovo model and its architecture, and reviews related work in glycoproteomics providing more context on how InstaNovo fits into current research in the field.

Chapter 3 outlines the methodology, including dataset description, exploratory analysis, data partitioning, and fine-tuning strategies. It also details training configurations such as learning rates and optimizers.

Chapter 4 presents the experimental setups and the results obtained, discussing the outcomes of the different configurations and approaches explored and comparing it with baseline approaches.

Chapter 5 concludes the thesis, summarizing contributions and suggesting directions for future work.

Notation and Terminology

Unless otherwise specified, the terms *peptide* and *sequence* refer to either a modified or unmodified peptide. When a distinction is necessary, we explicitly use the terms *modified peptide* or *unmodified peptide* or *glycopeptide*.

In the context of fine-tuning, all references to InstaNovo refer specifically to the pretrained InstaNovo 1.1.0 model.

2. Background

2.1 InstaNovo Overview

InstaNovo is a pretrained encoder-decoder transformer (Vaswani et al., 2017) neural network designed for direct peptide sequence prediction from MS2 spectra, without relying on external databases. The model learns spectral-to-sequence relationships based on the following features:

- *sequence*: The amino acid sequence, with potential modifications (e.g., LNGTDPIVAADSKR).
- *precursor_charge*: The charge state of the precursor after ionization (e.g., between +2 and +5).
- *precursor_mz*: The precursor ion's mass-to-charge ratio from MS1. It depends on the composition of the precursor and its charge after ionization.
- *intensity_array*: Fragment ion intensities from the MS2 spectrum, expressed as relative abundances between 0 and 1. These values indicate the strength of fragment ion signals within each spectrum.
- *mz_array*[1]: Fragment ion mass-to-charge ratios from the MS2 spectrum, which reflect the mass of the fragment ions.

The model has been trained on 56 millions samples, and has an architecture that features a (L=9)-layer encoder and (L=9)-layer decoder, utilizing sinusoidal peak embeddings, multi-head self-attention with 12 heads, and position-wise feed-forward networks to capture spectral patterns.

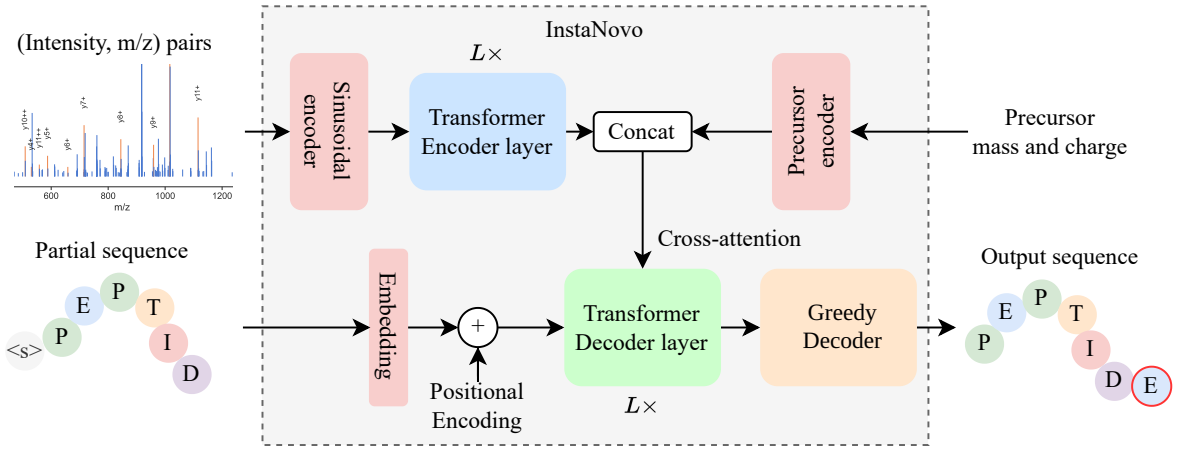


Figure 2.1: Overview of the InstaNovo architecture

During training, the MS2 spectrum is first processed by a dedicated peak embedding module that converts the raw peak intensities and masses into a high-dimensional representation. A latent spectrum token is prepended to this embedding to capture global spectral context, and the combined representation is passed through a transformer encoder that applies self-attention to capture relationships among spectral features.

Meanwhile, the peptide sequence (or a partial peptide) is first embedded using an amino acid embedding

[1]Parallel to *intensity_array* and together they represent a MS2 spectrum.

layer and then enhanced with positional information through a positional encoding module. This encoded sequence is input to the transformer decoder, which uses cross-attention to align and integrate the spectral context with the peptide predictions.

Additionally, precursor mass and charge are separately embedded and concatenated with the encoder output, thereby guiding the decoding process to enforce consistency with the precursor's properties.

Finally, at inference time, the model can employ an iterative decoding strategy that sequentially generates the peptide sequence while ensuring that the predicted sequence remains consistent with the precursor information (Algorithm 1).

Algorithm 1 InstaNovo: De Novo Peptide Sequencing from MS2 Spectra

Input:
Spectrum $MS2$ with N peaks
Precursor mass M_p, charge z_p
procedure GetPeptideSequence(PSM, $glyco\_thresh$, $general\_thresh$)
    // Spectral Encoding:
    for each peak in MS2 **do**
        Encode peak mass and intensity
    end for
    Add global context token
    // Precursor Encoding:
    Encode precursor mass and charge
    Combine with spectrum encoding
    // Transformer Processing:
    for $l \leftarrow 1$ to L_{layers} **do**
        Apply self-attention mechanism
        Apply feed-forward network
    end for
    // Simple Autoregressive Greedy Decoding: ▷ Simple mass constrained decoding
    $y_0 \leftarrow$ [SOS] ▷ Start token
    remaining_mass $\leftarrow M_p$
    for $t \leftarrow 1$ to max_length **do**
        Embed previous amino acid with position
        Apply cross-attention with spectrum
        Predict next amino acid
        Update remaining mass
        if remaining_mass ≤ 0 **or** [EOS] predicted **then**
            break
        end if
    end for
    return Complete peptide sequence
end procedure

2.2 Related Works

Most tools used in glycoproteomics today are based on fixed rules or database-driven search strategies. They often depend on known glycan structures and specific fragmentation patterns, or are computationally expensive. This can limit their ability to identify novel glycopeptides, especially when glycans are diverse or when data quality varies. The most widely cited glycopeptide identification tools in the literature are listed below, along with a brief summary of their underlying approaches.

pGlyco3 (Liu et al., 2017) follows a *glycan-first* strategy. It starts by looking for glycan specific fragment ions, especially Y-ions in the MS/MS spectrum. These are matched to a glycan database, and then possible peptide backbones are inferred. This order of operations is useful for well-characterized glycoproteins where expected glycans are already known. However, it struggles when the sample contains unexpected or rare glycan forms. While pGlyco3 includes a module (pGlycoNovo) for database-free sequencing by combining monosaccharides, this process becomes computationally expensive and less reliable with low-quality spectra. It performs best when spectra are clean and glycan compositions are typical.

MSFragger-Glyco (Polasky et al., 2020), part of the FragPipe suite, takes a different approach. It starts with peptides and searches for possible mass shifts that could indicate glycosylation. Glycan-specific fragment ions, like oxonium ions, are used to decide when a glycan search should be triggered. This makes the process efficient and scalable. MSFragger-Glyco handles both N- and O-glycosylation and supports "open" searches where unknown modifications can be detected based on their mass. However, its interpretation still depends on scoring rules, and it requires a protein sequence database. This means it's less suitable for completely novel samples or for de novo sequencing of unknown peptides.

Byonic (Bern et al., 2012) combines protein and glycan databases to search for glycopeptides. It is known for its flexibility in handling complex PTMs and mixed fragmentation types. It works well for high-resolution data and known proteins, but it performs worse when Y-ions are missing or weak. This is often the case in high-energy fragmentation modes, which leads to lower identification rates compared to tools like MSFragger-Glyco.

MetaMorpheus (Solntsev et al., 2018), particularly its O-Pair module, is designed to find O-glycosylation sites by aligning spectra against known peptide and glycan sequences. It uses electron-transfer dissociation (ETD) data to improve site localization. It does not support full de novo glycan sequencing but is useful when ETD data are available and glycans are moderately complex. Its performance depends heavily on the accuracy of database matches and the availability of high-quality spectral information.

PEAKS GlycanFinder (Sun et al., 2023) integrates de novo peptide sequencing with database-driven glycan identification. It breaks down the spectrum into parts that are then compared with theoretical glycan fragments. While it allows for some flexibility, it supports only a limited number of monosaccharide types. This makes it less useful when the sample includes rare or unexpected glycan compositions.

Glyco-Decipher (Fang et al., 2022) avoids glycan databases entirely. Instead, it uses a "stepping" method to add monosaccharide masses one at a time and test whether the resulting glycopeptide matches the spectrum. This allows it to identify novel glycoforms, but it also increases the risk of false positives. Isobaric (same-mass) glycans are difficult to distinguish this way, so the tool needs strong validation from other ions in the spectrum.

DeepGP (Zong et al., 2024) is one of the few tools to predict how glycopeptides should fragment using machine learning. It models retention time and ion intensity using neural networks. However, it still relies on database search as its main identification method. Its value lies in improving the scoring of

glycopeptide-spectrum matches when comparing candidates.

Finally, GlyCounter (Kothlow et al., 2025) is not a full search engine but a preprocessing tool. It looks for glycan-related fragment ions especially oxonium ions in spectra and extracts them for further analysis. This helps confirm that glycopeptides are present in the sample before a full identification is attempted. It's often used to enrich datasets or guide downstream searches.

In summary, these tools use various strategies to tackle glycopeptide identification. Some start from glycans, others from peptides, and a few try to bridge both. Most of them rely on known databases or scoring rules that may not generalize well to novel or low-abundance glycopeptides. This leaves an open space for approaches like InstaNovo, which aim to learn patterns directly from glycopeptide data and adapt to more complex or less predictable scenarios.

3. Methodology

3.1 Data Preparation

3.1.1 Data Analysis

The project's dataset is built from glycan-related mass spectrometry data collected from eight different projects in the PRIDE Archive (Perez-Riverol et al., 2025). These projects include samples from various species, resulting in a dataset that brings together glycosylation data from different biological sources as shown by Table 3.1 and Figure 3.1.

Project name	Samples count	Species
PXD025859	800,137	Human
PXD026629	381,457	Mouse
PXD026649	515,204	Human
PXD031025	54,684	Human
PXD031032	298,178	Mouse, Baker's yeast
PXD035158	312,275	Human, Mouse
PXD044641	43,539	Human, Domestic pig and more
PXD047898	325,901	Human
Total	2,731,375	N/A

Table 3.1: Samples count and species per project

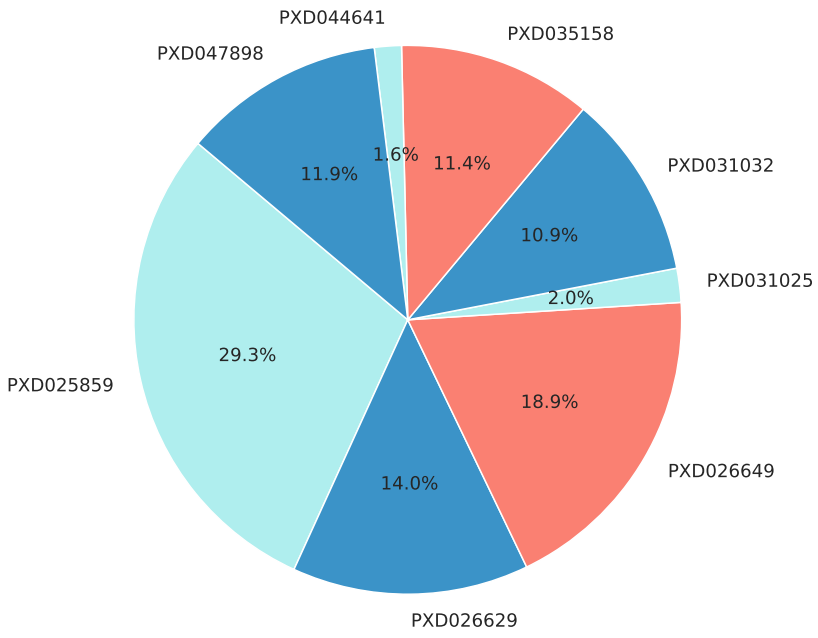


Figure 3.1: Contribution of each project to the entire glyco dataset before partitioning.

The dataset includes the five sample features needed to work with InstaNovo as mentioned in Chapter

2.1.

The distribution of modified peptides is skewed, with a small number of peptides appearing frequently while the vast majority occur only rarely (see Figure 3.2 for the top twenty most abundant). A similar pattern is observed for unmodified peptides, where most appear infrequently and only a few occur multiple times

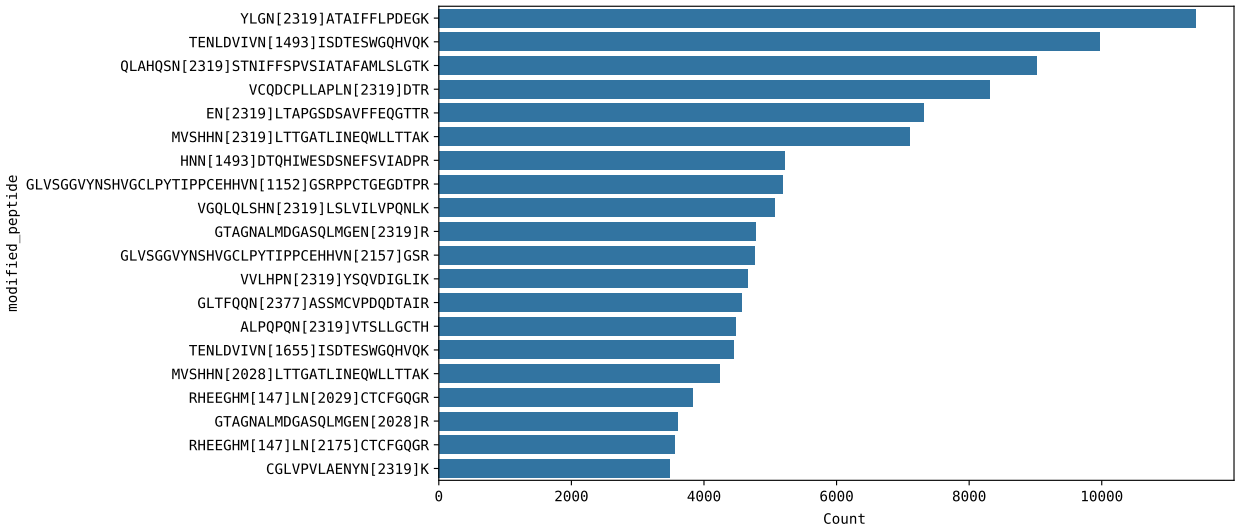


Figure 3.2: Top 20 most abundant modified peptides. As for unmodified peptides, most of these glycopeptides originate from the most abundant proteins in the dataset.

The comparison of experimental precursor m/z values with theoretical calculations revealed discrepancies, likely due to unaccounted modifications or instrumental factors (see Figure 3.3).

The analysis of the amino acid composition revealed that the dataset contains the 20 standard amino acids and Selenocysteine (U). Selenocysteine appears in only a single sample. Since the Instanovo 1.1.0 model does not support this residue, we excluded that sample from the dataset to maintain compatibility. In total, the dataset contains 247 PTMs: 245 glyco-modifications representing glycosylation with various glycans (see Figure 3.4), acetylation as a variable N-terminal modification[1], and variable methionine oxidation (see Table 3.2).

An inspection of the spectra reveals a mismatch between the experimental peaks captured by the spectrometer and the theoretical peaks that we would expect. This is likely due to the high mass of certain glyco-fragment ions, which may exceed the instrument's detection range or be excluded during precursor selection and fragmentation (see Figure 3.5).

3.1.2 Data Partitioning

Each dataset used to train Instanovo was partitioned using a group-based splitting algorithm, implemented as *GroupShuffleSplit* in Scikit-Learn (Pedregosa et al., 2011), where a group corresponds to an unmodified peptide. That means, all samples that have their peptide with modifications removed matching a given group (unmodified peptide) will belong to the same split, i.e. either train, test or valid. This approach helps prevent data leakage between splits. It also ensures that evaluation using

[1]The beginning of the protein or peptide sequence

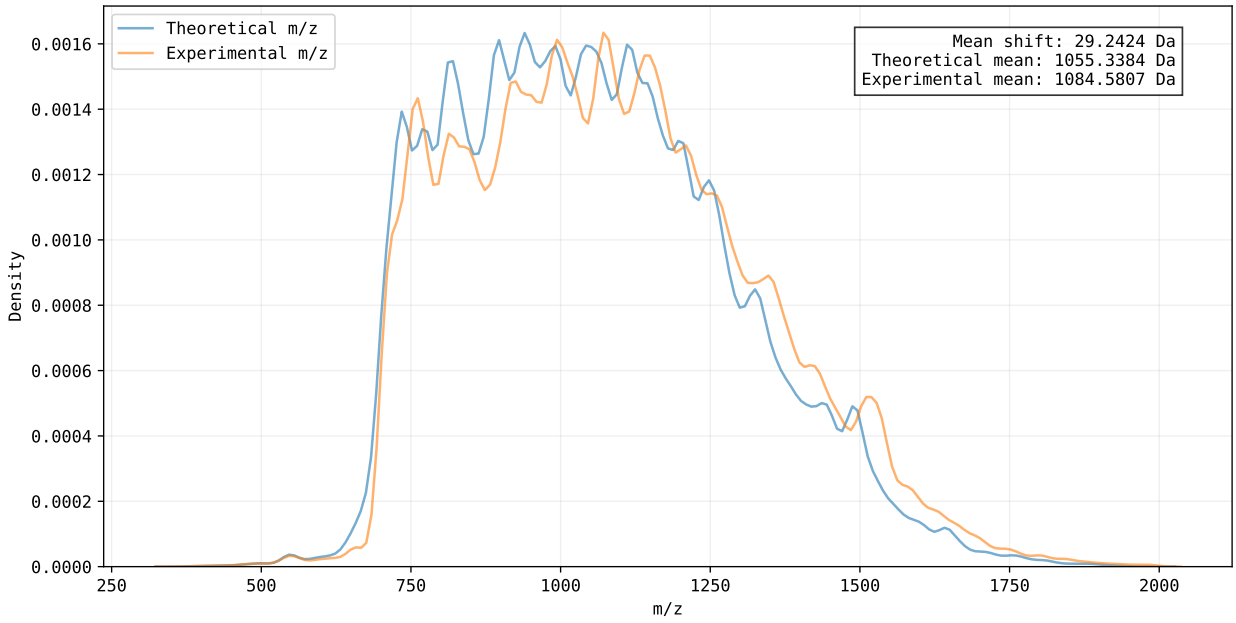


Figure 3.3: Density distributions of theoretical and experimental precursor m/z values. A noticeable rightward shift is observed in the experimental distribution compared to the theoretical one, with a mean shift of 29.24 Da.

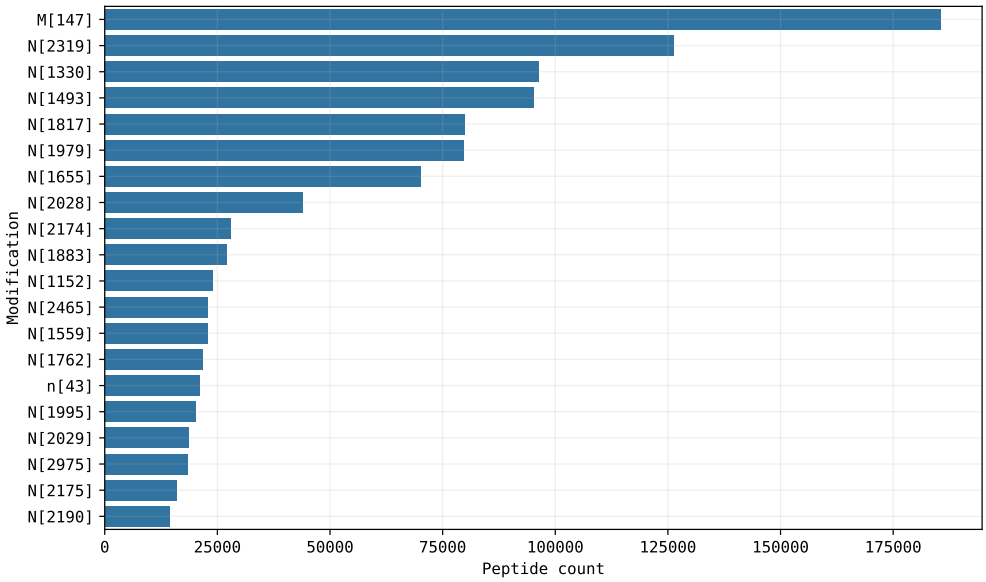


Figure 3.4: Top 20 most abundant modification sites across peptides containing at least one glycosylation. Each label represents a modified residue.

validation and test splits reflects the model's ability to generalize to new peptides, rather than memorizing sequences or modifications seen during training. To ensure consistency across Instanovo and its related projects, the mapping between each peptide and its assigned split was saved for every dataset used to train Instanovo. These split mappings, merged together, dictate a starting point for splitting any new dataset for Instanovo related projects.

Modification	Occ. 1	Occ. 2	Occ. 3	Occ. 4
Glycosylation	1,429,125	0	0	0
Oxidation	273,792	16,838	586	0
Acetylation	23,568	0	0	0
Any modification	1,424,737	208,702	10,434	392

Table 3.2: Distribution of sequences with exactly 1 to 4 occurrences of specific modification types. Each column indicates the number of sequences with the corresponding number of modifications. Glycosylation occurs only once per sequence, while oxidation can occur up to three times, and acetylation appears only once. The "Any modification" row aggregates counts across all modification types, showing that while single modifications are most common, a substantial number of sequences contain multiple modifications.

Figure 3.5: Comparison between the experimental MS2 spectrum (top, blue) and the theoretical fragment ions (bottom; green for y-ions, red for b-ions) for the glycosylated peptide AMSSN[1979]ETAAYK. All theoretical ion intensities are set to 1 for visibility.

We have created three distinct dataset splits S_{all}, S_{top6} and S_{ftop5} for experimental purposes:

1. S_{all} contains all samples from the dataset, whether or not they have glyco-modifications.
2. S_{top6} is a subset of S_{all}, containing only sequences with one of the six most abundant glyco modifications (N[2319], N[1330], N[1493], N[1817], N[1979], N[1655]). All sequences in this subset are glyco-modified.
3. S_{ftop5}[2] contains the top five glyco modifications (N[2319], N[1330], N[1493], N[1817], N[1979]), after filtering all PSMs to retain only spectra with at least one glyco peak and three non-glyco peaks (see `ng` and `n` in Algorithm 2).

S_{all} and S_{ftop5} were created starting by using the merged split mappings mentioned previously and then distributing the remaining samples using a greedy assignment algorithm (see Algorithm 3) with the standard deviation of the group sizes as the threshold to approximate the desired ratio which is 80/10/10% in terms of samples per split, rather than the number of unmodified peptides per split. It resulted in 80.01/9.99/9.99% ratio for S_{all} and 80.10/9.95/9.95% ratio for S_{ftop5}. This approach helps prevent data splits from being extremely imbalanced in terms of number of samples that would negatively impact model training and evaluation. We acknowledge that this splitting approach does not necessarily guarantee that the variation in dataset distribution is well captured across all splits which is the ultimate goal. However, our method ensures better control over the global samples distribution.

Split	Train	Valid	Test	Total
S_{all}	2,185,621	272,877	272,877	2,731,375
S_{top6}	496,199	21,185	23,395	540,779
S_{ftop5}	311,472	38,700	38,700	388,872

Table 3.3: Sample counts for each dataset split, including total per split.

3.2 Fine-tuning Strategy

In this work, we investigated several fine-tuning methods to adapt InstaNovo to our target task, taking inspiration from the successful phosphorylation fine-tuning strategy described by Lauridsen et al. (2025), while also exploring additional approaches tailored to our specific context. However, key differences such as our smaller dataset size, the low representation and high diversity of glyco-modifications (all unique per sequence), and the mass-shift impact of glycosylation on spectra meant that we did not necessarily expect to surpass the results achieved in phosphorylation adaptation. Nevertheless, we approached fine-tuning with an open mindset, aiming to understand what works, what does not, and why, in order to confidently move toward the directions of achieving strong performance.

The fine-tuning approaches considered are listed below, with the top four corresponding to those investigated during the phosphorylation adaptation of InstaNovo.

- **Fine-Tuning (Full Fine-Tuning)**: All the pre-trained model's parameters are updated from the beginning to the end of the training.
- **Head-Only (Linear Probing)**: Only the classification head (the final layer) of the pre-trained model is trained, while the rest of the model remains frozen (Alain and Bengio, 2018).

[2] ftop5 for Filtered Top 5

Algorithm 2 PSM Filtering Algorithm

Input:
  PSM (Peptide-Spectrum Match)
  $glyco\_thresh$: Minimum glyco-site matches (default: 1)
  $general\_thresh$: Minimum general ion matches (default: 3)
  tol: Ions m/z matching tolerance (default: 2)
procedure FILTERPSM(PSM, $glyco\_thresh$, $general\_thresh$, tol)
    // Preprocess spectrum
    $spectrum \leftarrow$ retain peaks with $m/z \in [50, 2500]$
    $spectrum \leftarrow$ normalize intensities$(spectrum)$
    // Generate theoretical b-ions
    $Mb\_g1 \leftarrow \{b$-ion masses containing glyco-site$\}$
    $Mb\_y \leftarrow \{b$-ion masses without glyco-site$\}$
    // Generate theoretical y-ions
    $My\_g1 \leftarrow \{y$-ion masses containing glyco-site$\}$
    $My\_y \leftarrow \{y$-ion masses without glyco-site$\}$
    // Count glyco-site matches (n_g)
    $n_g \leftarrow 0$
    for each $peak$ in $spectrum$ **do**
        if $\exists m \in Mb\_g1 \cup My\_g1 : |peak.mz - m| \leq tol$ **then**
            $n_g \leftarrow n_g + 1$
        end if
    end for
    // Count general ion matches (n)
    $n \leftarrow 0$
    for each $peak$ in $spectrum$ **do**
        if $\exists m \in Mb\_y \cup My\_y : |peak.mz - m| \leq tol$ **then**
            $n \leftarrow n + 1$
        end if
    end for
    // Filtering decision
    if $n_g \geq glyco\_thresh$ **and** $n \geq general\_thresh$ **then**
        return true ▷ Keep PSM
    else
        return false ▷ Skip PSM
    end if
end procedure

Algorithm 3 Threshold Based Greedy Assignment

Input:
  $grouped\_seqs$: Samples to distribute grouped by unmodified peptides
  $needs$: Map of number of samples needed by each split
  $splits$: List of splits that need more samples ▷ Split labels
  $thresh$: Threshold that control the error we can make while adding more samples
  $exhaust$: Boolean telling whether all the samples should be distribute despite $thresh$ or not
procedure GREEDYASSIGN($grouped\_seqs$, $needs$, $splits$, $thresh$, $exhaust$)
    $split\_to\_seqs \leftarrow$ map from each label in $splits$ to empty list
    $seq\_to\_split \leftarrow$ empty map
    for all $(seq, samples)$ in $grouped\_seqs$ **do**
        $count \leftarrow$ length of $samples$ ▷ Number of samples in the current group
        $best \leftarrow$ label in $splits$ with highest $needs[best]$
        $surplus \leftarrow count - needs[best]$ ▷ Surplus if assigned
        if $surplus \leq thresh$ **or** $exhaust$ **then** ▷ Assign if within threshold or if exhaustion enabled
            $split\_to\_seqs[best] \leftarrow split\_to\_seqs[best] + samples$ ▷ Append
            $seq\_to\_split[seq] \leftarrow best$
            $needs[best] \leftarrow needs[best] - count$
        end if
    end for
    return $(split\_to\_seqs, seq\_to\_split)$
end procedure

- **Head-Then-Full (Linear-Probing-Then-Fine-Tuning)**: This hybrid strategy first trains the classification head with the backbone frozen (linear probing), followed by full fine-tuning of all model parameters (Tomihari and Sato, 2024).
- **Gradual Unfreezing**: This technique involves progressively unfreezing layers of the pre-trained model during fine-tuning, usually starting from the top layers and moving downwards. It may allow the model to adapt more effectively to the new task without disrupting the lower-level features learned during pre-training (Wiener, 2024).
- **Full-Then-Gradual-Unfreezing**: All the model is trained for few epochs and then we freeze all the parameters leaving only one and then gradually unfreeze more parameters.
- **LoRA (Low-Rank Adaptation)**: A parameter-efficient fine-tuning method where the pre-trained model weights are frozen and small, trainable low-rank matrices are inserted into specific layers (e.g., attention or feed-forward layers). Only these additional parameters are optimized during training (Hu et al., 2021).

We distinguish between two general strategies: one where all model parameters are optimized together in a single phase for the entire training, and another where the training is divided into multiple phases, each focusing on different sets of parameters.

In each training phase, we focused on quickly reducing the loss to speed up convergence after trying to access the near-minumum value of the loss using various parameters and learning rate schedulers. At the same time we also keep an eye on the validation loss to avoid overfiting. By doing this we seek to adapt each parameters at each phase as much as possible to our task without overfitting and also cut down on future computing costs by avoiding extra, unnecessary training.

In parallel to fine-tuning Instanovo, we also investigated how fine-tuning performance would be impacted when using pre-trained models that had been initially trained on glycopeptide-enriched datasets. This has been done by training a lightweight version of Instanovo using a combination of a dataset called AC-PT (all confidence ProteomeTools, (Eloff et al., 2025)) and oversampled train set of the glyco dataset splits (Zolg et al., 2018).

4. Experiments and Results

4.1 Setup

One of the main challenges in the experiments was ensuring a fair comparison between models, as the conditions were not always consistent. To address this, we chose to focus on optimizing performance specifically for glycopeptides predictions and access effective performance on glyco validation sets. We then evaluated how the best-performing glyco model performed on AC-PT validation set, in order to assess knowledge conservation and catastrophic forgetting effect (Luo et al., 2025). This setup provides a more consistent basis for comparison. Accordingly, the models presented here were selected with that goal in mind.

4.1.1 Learning Rate Scheduler and Optimizer

We initially tested both Adam (Kingma and Ba, 2017) and AdamW (Loshchilov and Hutter, 2019) optimizers across various parameter combinations. Since their performance was nearly identical in our first experiments we chose to proceed exclusively with AdamW (see Figure 5.1 for example). We evaluated several learning rate schedulers, including StepLR, LambdaLR, LinearLR, and CosineAnnealingWarmRestarts from the PyTorch library (Paszke et al., 2019), as well as Slanted Triangular Learning Rate (STLR) (Howard and Ruder, 2018) and Linear Warmup (Ma and Yarats, 2021). STLR and Linear Warmup performed best for our experiments. Both include warm-up mechanisms, which help stabilize early training especially helpful when fine-tuning pretrained models (Kalra and Barkeshli, 2024).

4.1.2 Parameters Hand-tuning

In the search for optimal configurations, we manually adjusted key hyperparameters to fine-tune InstaNovo. This included experimenting with different values for learning rate, batch size, weight decay, and dropout rate. While we explored various settings for all four, weight decay and dropout were fixed at 1e-5 and 0.1, respectively, for the experiments presented below. Final configurations were selected based on their impact on training, validation loss and glycosylated peptide prediction metrics.

4.1.3 Dataset Selection for Learning Quality

Given the number of dataset splits we have, a key challenge was determining which split would allow the model to learn most effectively. Using too many configurations would result in an unmanageable number of experiments, so our first goal was to identify a split that provides strong training signals and overall in predicting glycopeptides.

We hypothesized three potential learning behaviors to explore this problem experimentally:

1. The model might learn better with S_{ftop5} as defined in Section 3.1.2.
2. Since filtering helps retain spectra with more (informative) peaks, the resulting filtered spectra may resemble those the model was originally trained on. In contrast, spectra that do not meet the filtering criteria could act as strong discriminative signals, helping the model to better learn the distinction. This would mean that the model will perform better on spectra that won't pass S_{ftop5} filtering criteria.
3. The model may learn effectively by revisiting and relearning unmodified sequences during training, which could help in better discrimination between glycopetides and unmodified peptides.

To evaluate these hypotheses, we ran some full fine-tuning experiments using S_{ftop5} and two different adjusted versions of S_{top6}:

1. S_{top6} with the sixth most frequent glyco modification removed: This includes the samples of S_{ftop5} plus additional spectra that don't meet the S_{ftop5} filtering criteria.
2. S_{top6} + Spectra with Unmodified Sequences from S_{all}: Combines glyco-modified sequences from S_{top6} with unmodified sequences from S_{all}.

The results show an interesting trend (see Figure 4.1): the model seems to learn to predict glycopeptides better when exposed to spectra that didn't pass the filtering criteria of S_{ftop5}. This observation encourages a focus on using one of the two last dataset splits for subsequent experiments to reduce the number of experiments.

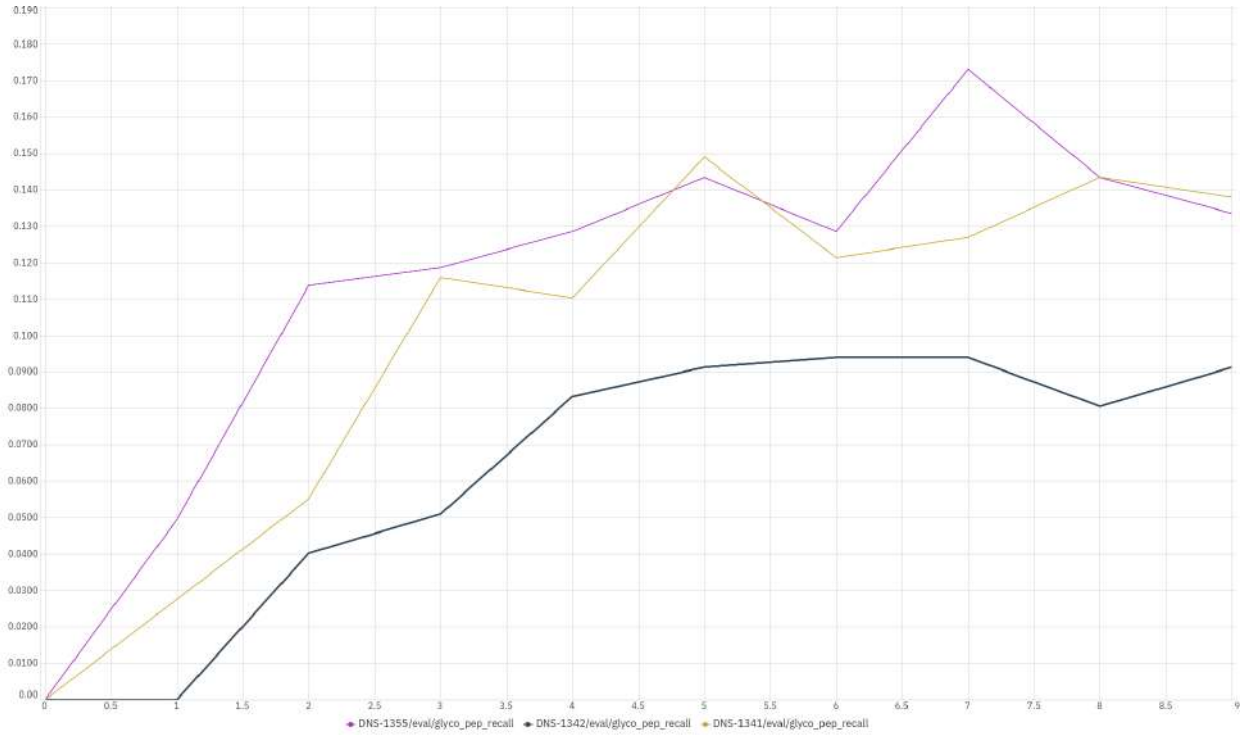


Figure 4.1: Glycopeptide recall during fine-tuning on three datasets: S_{ftop5} (dark blue), S_{top6} with the sixth most frequent modification removed (yellow), and S_{top6} + unmodified sequences (purple). Results show improved glycopeptide prediction performance with both S_{top6} variants compared to S_{ftop5} and this has been the case for two different training configurations, one using Linear Warmup and another one using STRL.

We also attempted the inverse case: fine-tuning on spectra that did not meet the filtering criteria of S_{ftop5}. However, the number of samples that failed to meet the criteria for S_{ftop5} was relatively small. To address this, we made the filtering criteria of S_{ftop5} more stringent, so that the set of spectra excluded by the filter would be larger and more suitable for training. Specifically, we set `general_thresh` to 4, `glyco_thresh` to 2, and `tol` to 0.2 (see Algorithm 2). With this stricter configuration, we observed similar learning dynamics between the original S_{top6} dataset and the new dataset built from the excluded spectra (see Figure 4.2). Since the experiments were run under mostly the same conditions, the results suggest that S_{ftop5} may not be the best choice going forward, as the filtering does not appear to provide a more favorable signal for learning glycopeptide predictions.

To further support this, we also compared the learning behavior of two distinct InstaNovo-like models trained from scratch using only the AC-PT dataset in combination with different glyco-enriched subsets. The first model was trained on AC-PT combined with an oversampled version of S_{top6}, while the second was trained on AC-PT combined with an oversampled version of S_{ftop5}. Both subsets were oversampled to the same extent to ensure a fair comparison. The model trained with S_{top6} that we named InstaNovo$_{\text{top6}}$ later in Table 4.1, consistently exhibited more favorable learning dynamics (see Figure 4.4 and 4.3), reinforcing the idea according to which the split S_{top6} gives more favorable learning signal.

We therefore chose to focus on using S_{top6} as the default dataset in our fine-tuning experiments, except in specific cases where we aimed to analyze particular behaviors of S_{ftop5}.

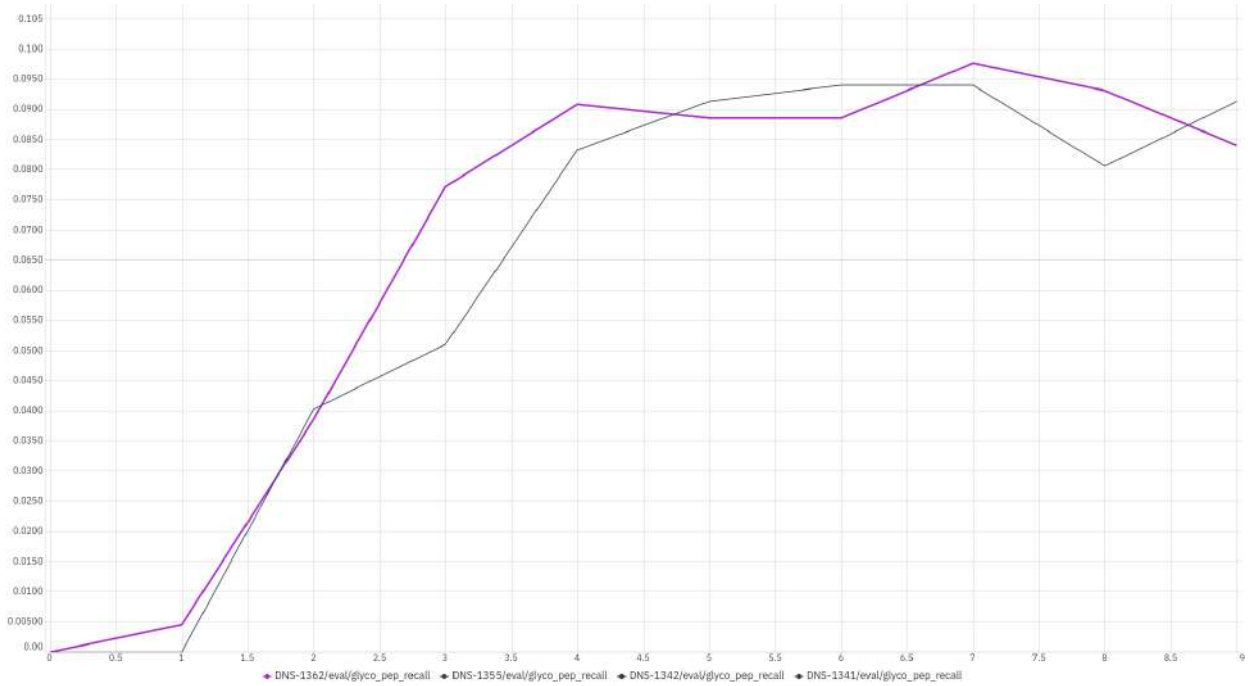


Figure 4.2: Glycopeptide recall comparison during fine-tuning between the original S_{ftop5} dataset (dark blue) and a new dataset built from spectra that did not pass a stricter Sftop5 filter (purple; `geneal_thresh` $= 4$, `glyco_thresh` $= 2$, `tol` $= 0.2$), showing that both sets lead to similar learning dynamics.

4.2 Fine-tuning Results

As mentioned in Section 3.2, here is a more detailed view of the fine-tuning approaches explored (excluding LoRA[1]):

- Full fine-tuning (Full[2]) :The entire InstaNovo model, including encoder and decoder, is fine-tuned end-to-end.
- Linear probing (Head): Only the prediction head is fine-tuned, while the encoder and decoder are kept frozen.

[1] It is discussed later in Section 4.3.2.
[2] Short name used in the results table later.

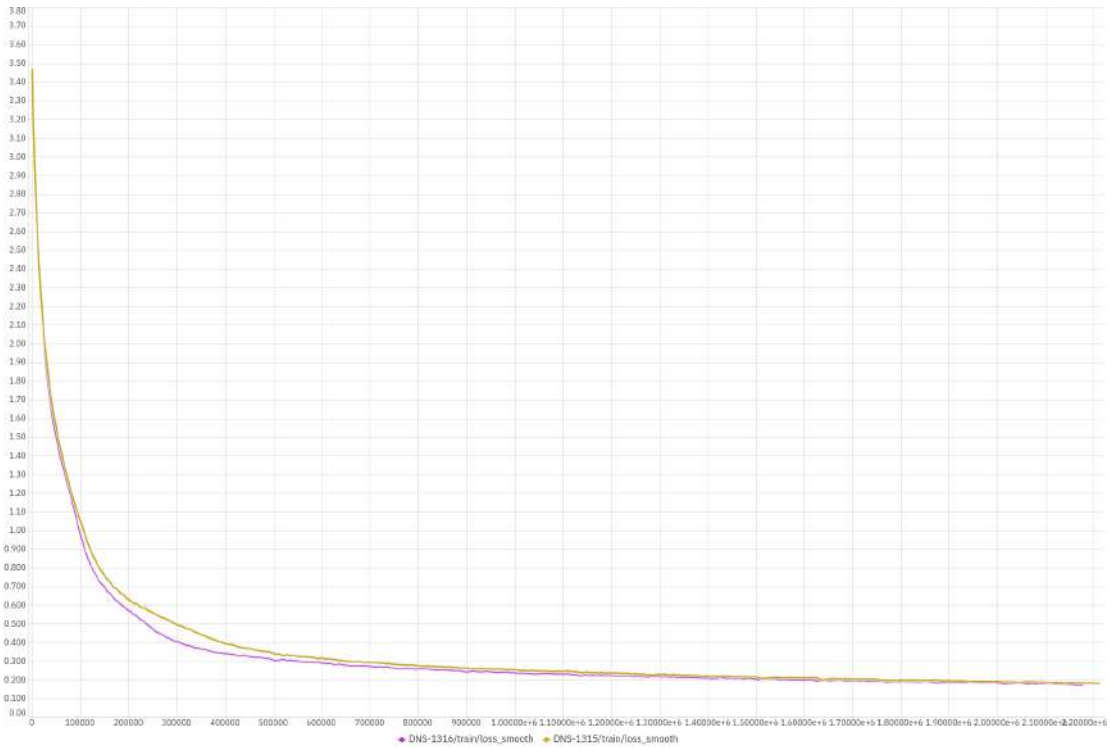


Figure 4.3: Losses during scratch-training using AC-PT + S_{ftop5} (purple) and AC-PT + S_{top6} (yellow). They seem to be learning the same way but are not learning the same thing as the peptide recall plots show.

- Full-then-Head (Full→Head): The full model is first fine-tuned, after which only the head is retrained with the rest frozen.
- Head-then-Full (Head→Full): The head is trained first, then the encoder and decoder are unfrozen for full fine-tuning.
- Decoder-first: A gradual unfreezing strategy where the head is trained first, followed by the decoder, and then the encoder.
- Encoder-first: A gradual unfreezing strategy where the head is trained first, followed by the encoder, and then the decoder.
- Full-then-Encoder (Full→Encoder-first): The model is fully fine-tuned first, then only the encoder is retrained.
- Encoder-first$_{\text{top6}}$: The Encoder-first strategy applied to InstaNovo$_{\text{top6}}$, the InstaNovo-like model that we trained from scrath using AC-PT and S_{top6} as mentined in the Section 4.1.3. This has been done to assess how prior exposure to glyco-modified sequences can affect the fine-tuning performance.

In Table 4.1, fine-tuning results are presented.

This result clearly shows that the fine-tuned models, across all approaches tested, struggle to reach production-level performance. In addition, they suffer from catastrophic forgetting, losing their ability to accurately decode sequences without glyco modification.

Figure 4.4: Glycopeptide recall during scratch-training using AC-PT + S_{ftop5} (purple) and AC-PT + S_{top6} (yellow). The results reinforce our hypothesis that S_{top6} due to spectra excluded by the S_{ftop5} filtering criteria provides a more discriminative signal.

To better understand the limitations observed during fine-tuning, we projected the embeddings of three datasets into the model's latent space. These include AC-PT (used during the pretraining of InstaNovo), the phosphorylation dataset on which fine-tuning was previously successful (Lauridsen et al., 2025), and various splits of the glyco dataset. Including the phosphorylation dataset serves as a positive control, allowing us to contrast a successful fine-tuning case against the more challenging glyco datasets. By comparing the distributions of these embeddings, we aim to assess potential domain shifts and better explain the performance drop observed in glyco-specific fine-tuning.

In Figure 4.5, we project the embeddings of AC-PT (green), Phospho[3] (blue), and S_{ftop5} (red). The red points form a tight cluster that is clearly separated from both AC-PT and Phospho in the PCA space. This strong separation visually confirms earlier observations: models fine-tuned on S_{ftop5} perform less. The representation shift visible here reinforces that the glyco spectra selected by the stricter filtering criteria are distributionally very distant from what InstaNovo was pretrained on, making them hard to learn from directly.

In Figure 4.6, we examine the spectra that were excluded by the S_{ftop5} filter; in other words, glyco spectra that failed to meet the filtering criteria of S_{ftop5}. Interestingly, these samples are still distant from the AC-PT and Phospho embeddings, but begin to show slight overlap, particularly around the edges of the distribution. This supports previous experimental findings: the model showed improved learning dynamics on these excluded spectra compared to S_{ftop5} itself, likely because these examples are less extreme in glycan signal and thus closer to the peptide domain the model was trained on.

[3] Refers to the phosphorylation dataset.

(a) Peptide-Level Metrics (%)

Model	Recall		Recall @ 5% FDR ↑	
	AC-PT	S_{top6}	AC-PT	S_{top6}
Full	18.24	9.58	0.00	0.00
Head	**59.23**	3.66	0.04	0.00
Full→Head	22.00	10.09	0.00	0.00
Head→Full	26.68	**14.74**	0.01	0.00
Decoder-first	17.65	11.12	0.00	0.00
Encoder-first	47.92	11.23	3.00	0.00
Full→Encoder-first	15.48	11.10	0.01	0.00
InstaNovo	**68.24**		**51.48**	
InstaNovo$_{\text{top6}}$	59.26	3.88	32.44	0.00
Encoder-first$_{\text{top6}}$	56.76	3.00	16.81	0.00

(b) Amino Acid-Level Metrics (%)

Model	Recall		Precision		AA Error ↓	
	AC-PT	S_{top6}	AC-PT	S_{top6}	AC-PT	S_{top6}
Full	25.94	15.10	25.43	14.88	54.16	58.12
Head	**70.00**	8.34	**69.84**	8.05	**19.24**	64.41
Full→Head	30.21	15.79	31.13	15.66	47.14	57.19
Head→Full	31.79	**19.24**	33.01	19.17	51.62	57.23
Decoder-first	23.92	16.50	24.81	16.50	55.60	57.70
Encoder-first	58.01	18.39	57.65	18.60	28.79	**54.08**
Full→Encoder-first	22.19	16.80	22.52	16.81	59.57	56.75
InstaNovo	**76.69**		**75.69**		**15.99**	
InstaNovo$_{\text{top6}}$	68.60	6.28	68.16	6.29	20.64	70.49
Encoder-first$_{\text{top6}}$	63.92	4.77	63.34	4.81	24.22	74.46

Table 4.1: Performance comparison of different models across datasets and metrics.

The Figure 4.7 focuses on glyco spectra with no glyco peak matches and having at most one non-glyco peak match i.e. as started in the Algorithm 2, we want to retain only the spectra having $\texttt{ng} \leq 1$ and $\texttt{n} = 0$. These spectra show substantial overlap with AC-PT and Phospho in the embedding space. This indicates that when the glycan signal is minimal or absent, the spectra are structurally and representationally much more compatible with the model's pretraining distribution. And such the model still find a way to adapt better compare to the case of S_{ftop5}.

Together, these projections make it clear why InstaNovo fine-tuning was successful for phospho-sequencing but not for glyco-sequencing. Phosphorylated spectra remain close to the AC-PT training distribution, so the model easily adapts. In contrast, glyco spectra in general occupy a distant region in embedding space, revealing a strong domain shift and explaining why fine-tuning failed without glyco-specific pretraining or architectural adjustments.

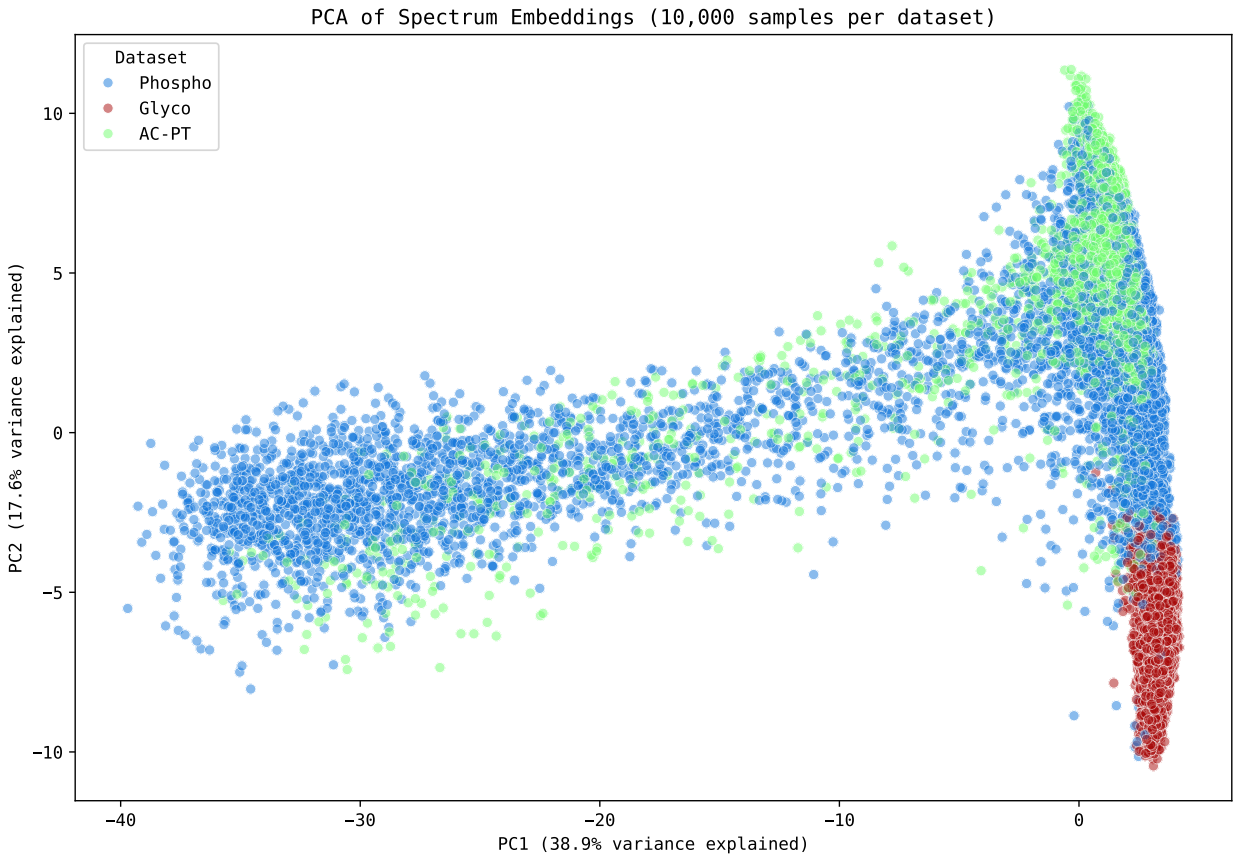


Figure 4.5: PCA comparing AC-PT and Phospho with S_{ftop5}.

4.3 Discussions

4.3.1 Insights

The fine-tuning of InstaNovo on glycoproteomics data highlights both its potential and its current limitations in adapting to more complex sequence prediction tasks. While the model shows some ability to learn from glycopeptide spectra, the improvements remain modest. This suggests that substantial challenges still need to be addressed before such models can reach production-level reliability for glycopeptide identification.

Notably, training on spectra failing the filtering criteria of S_{ftop5} yielded stronger learning signals than training on the filtered S_{ftop5} dataset. One explanation for this, even without PCA insights, is that the filtered spectra may closely resemble the spectra InstaNovo was originally trained on. In this case, the model might already be overconfident in associating such spectra with unmodified peptide sequences, making adaptation to glycosylated sequences more difficult.

In contrast, many of the spectra failing the filtering criteria are less similar to the model's training distribution. This dissimilarity may provide stronger discriminative signals that help the model better differentiate between glycopeptides and non-glycopeptides.

A more tangible explanation comes from the PCA projections shown in Figures 4.5,4.6 and 4.7. These figures clearly illustrate a domain shift between AC-PT (used to train InstaNovo), the Phospho dataset,

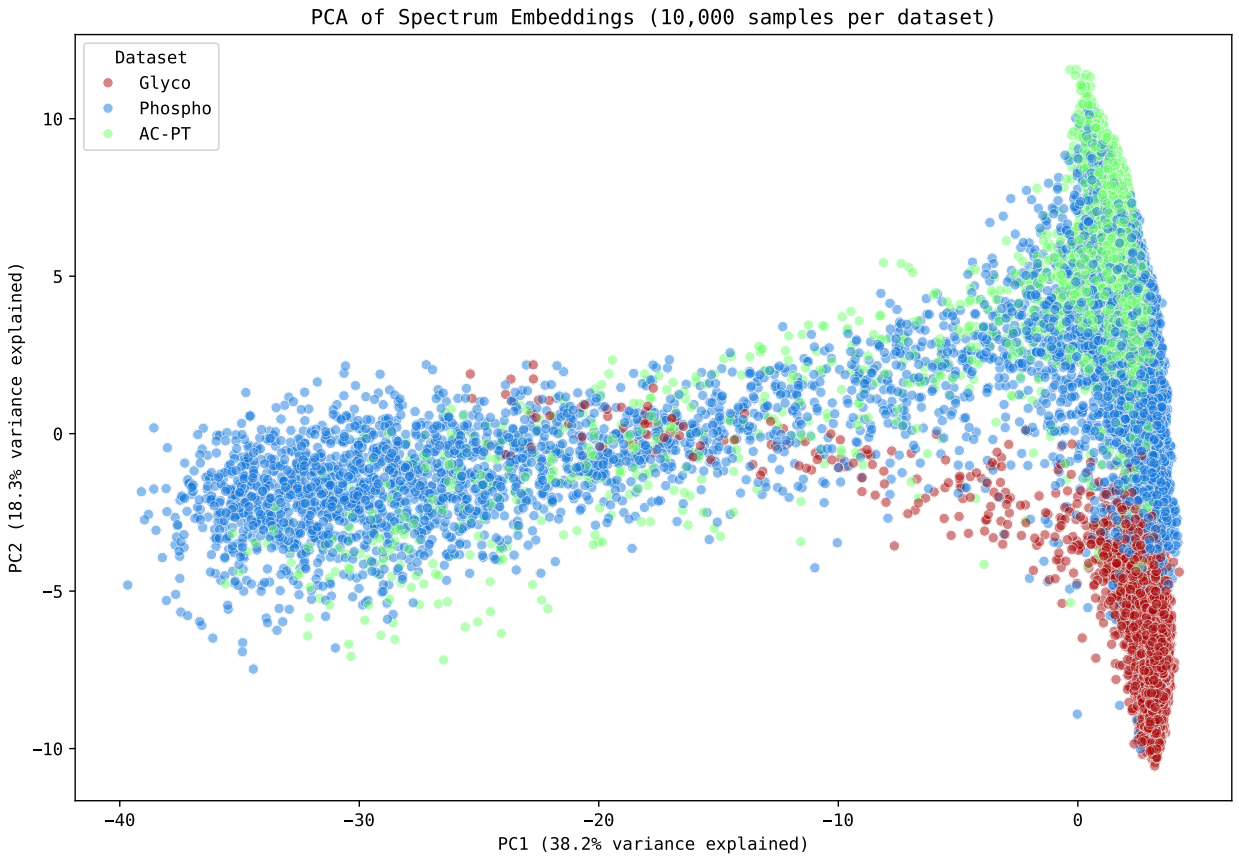


Figure 4.6: PCA comparing AC-PT and Phospho with spectra of S_{top6} failing S_{ftop5} filtering criteria.

and the glyco datasets. This separation in the embedding space highlights why fine-tuning succeeds on Phospho data but struggles on glyco data. The domain shift is likely driven by the heavier nature of glycopeptides (see Figure 4.8) and their characteristic mass shifts, which set them apart from the model's original representation space.

Overall, these results show that while InstaNovo can begin to learn from glycoproteomics data, fine-tuning a peptide-focused model is not sufficient. Moving forward, progress will likely require dedicated model designs, more targeted training strategies, or loss functions tailored to glycan-related features.

4.3.2 Unsuccessful Fine-Tuning Approaches

Vocabulary Transfer with Embedding Adjustments Mosin et al. (2023b) The large increase in vocabulary size due to glycan modifications led us to test vocabulary transfer methods to help InstaNovo learn glycopeptide patterns faster using its existing peptide sequencing knowledge. We tried two approaches: (1) using expected embeddings from the pre-trained model and (2) setting embeddings of all glycan-modified amino acids to match the embedding of unmodified asparagine, as modified and unmodified asparagine share chemical properties. While the expected embedding approach slightly sped up early training in some particular configuration, both methods failed to improve overall glycopeptide prediction accuracy.

Cross-Entropy Loss for Imbalanced Token Distribution The addition of 245 glycan-modified tokens created an imbalanced dataset, with glyco-tokens appearing less frequently than standard

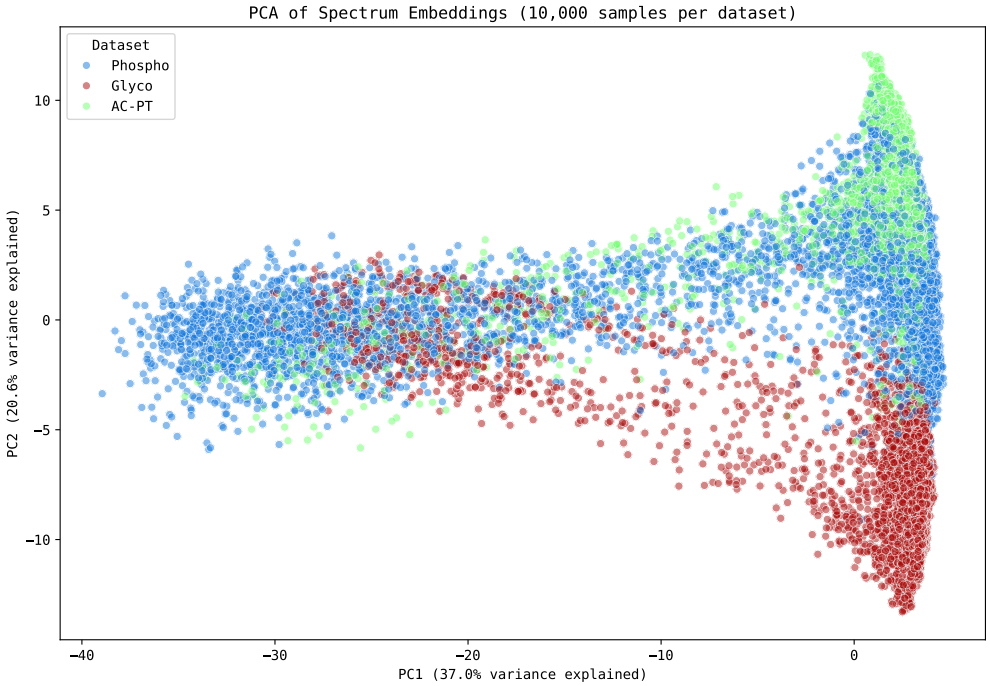


Figure 4.7: PCA comparing AC-PT and Phospho with spectra with 0 glyco peak and at most 1 non glyco peak match.

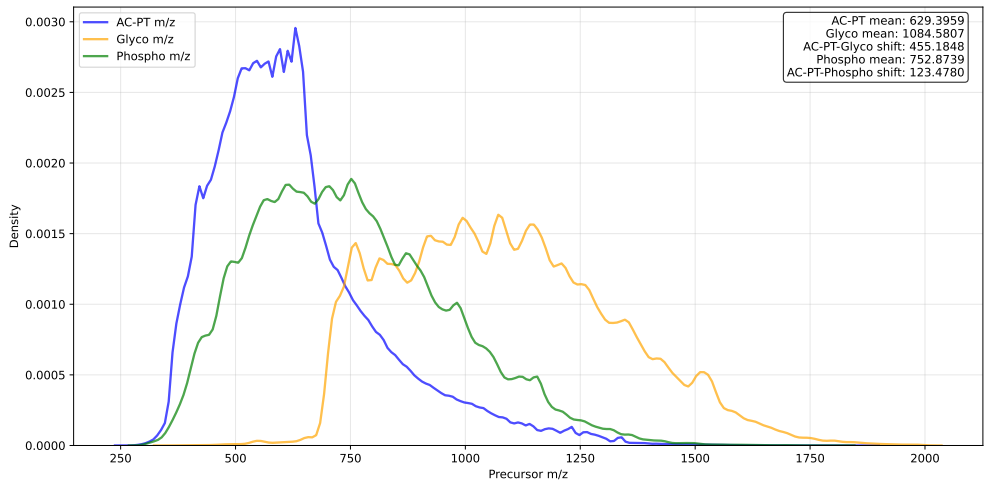


Figure 4.8: Experimental m/z distribution of the AC-PT, Phospho, and Glyco datasets, highlighting the heavier nature of glycopeptides.

amino acid tokens. To address this, we tested a cross-entropy loss function to encourage the model to prioritize predicting glyco-tokens. This approach performed poorly, resulting in one of the worst experiments. The failure likely stems from the loss function overemphasizing rare tokens, causing the model to lose accuracy on unmodified peptide sequences without significantly improving glycopeptide predictions. This highlighted the need for a balanced training approach.

LoRA (Low-Rank Adaptation) We experimented with LoRA, a parameter-efficient fine-tuning method,

to adapt InstaNovo to glycopeptides while preserving its pre-trained weights. However, the results were poor, with no significant improvement in glycopeptide sequencing accuracy. LoRA's limited capacity to adjust the model for the large, complex vocabulary of glycan modifications likely prevented it from learning the intricate patterns needed for accurate predictions, indicating that more extensive fine-tuning was necessary.

Consideration of Other Loss Functions Recognizing the imbalanced dataset, we considered experimenting with alternative loss functions designed for imbalanced data, such as focal loss. However, the poor performance of the cross-entropy loss experiment discouraged further exploration in this direction. The failure of cross-entropy loss suggested that modifying the loss function alone may not be enough to address the challenges of spectral variability.

Mass Constrained Decoding We started with mass-constrained decoding using InstaNovo's greedy search to decode glycopeptides, aiming to use mass constraints to improve accuracy. This approach did not work, failing to accurately decode glycopeptides. Our investigation into why was shallow, but early findings suggest the greedy search method struggles with the complex mass shifts from glycan modifications. This needs proper investigation as understanding the limitations of mass-constrained decoding could unlock improvements in glycopeptide sequencing.

5. Conclusion and Future Work

This study demonstrates that while InstaNovo can begin to adapt to glycoproteomics through fine-tuning, the improvements remain modest. Glycopeptide sequencing presents unique challenges due to domain shifts and structural complexity not encountered in peptide-only training. Experiments show that fine-tuning on glyco datasets leads to limited performance gains and suffers from catastrophic forgetting of previously learned peptide sequences. And also training a small version with AC-PT with glyco spectra under the default InstaNovo training settings does not help the model to sequence glycopeptides better.

Interestingly, unfiltered spectra offer stronger learning signals, revealing the potential of using "noisy" (spectra with missing peak or peaks not matching the theoretical peaks) data as a teaching signal rather than discarding it. This suggests that exposing the model to more "noisy" spectra may help it develop better representations for glycosylation. PCA projections of spectrum embeddings reinforce this idea: glyco spectra occupy regions far from the original training distribution, while phosphorylation data lies closer. This separation helps explain why InstaNovo fine-tunes successfully on phosphorylation data but struggles on glycopeptides.

To improve glyco-aware modeling, future efforts should go beyond fine-tuning. Promising directions include:

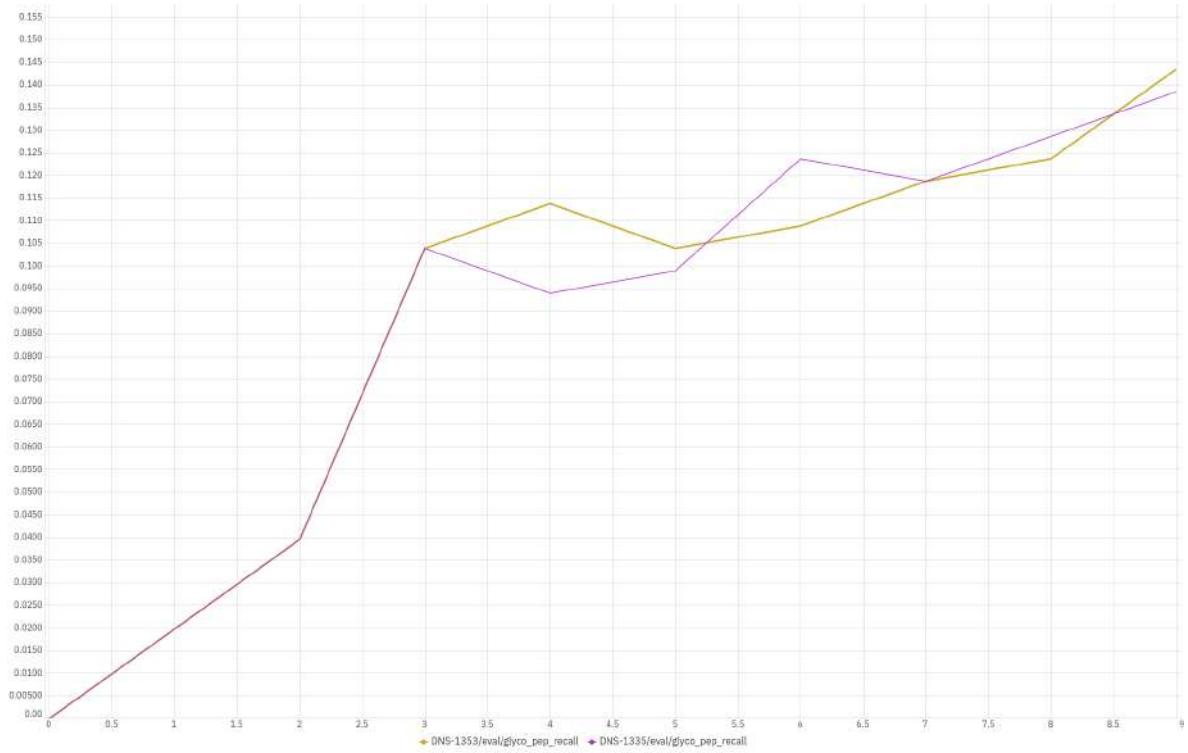
- **Richer data:** Collect more glycopeptide spectra that overlap with the AC-PT dataset in the embedding space, as these examples may help bridge the domain gap and improve generalization.
- **Invariant transformation design:** Develop transformations that leave AC-PT spectral embeddings unchanged in the embedding space while shifting glycopeptide spectra closer to that region. This could reduce domain mismatch and facilitate more effective fine-tuning.
- **Multi-task loss (Gong et al., 2019):** Incorporating auxiliary objectives for glyco token prediction to help the model learn both peptide and glycan simultaneously: basically, add another classification head.
- **Attention-guided loss (Wang et al., 2025):** Leveraging attention mechanisms to focus learning on glycan-relevant regions of the spectrum.
- **Contrastive learning (Rethmeier and Augenstein, 2021):** Structuring the embedding space so that glycosylated and non-glycosylated spectra form separable clusters.

Appendix

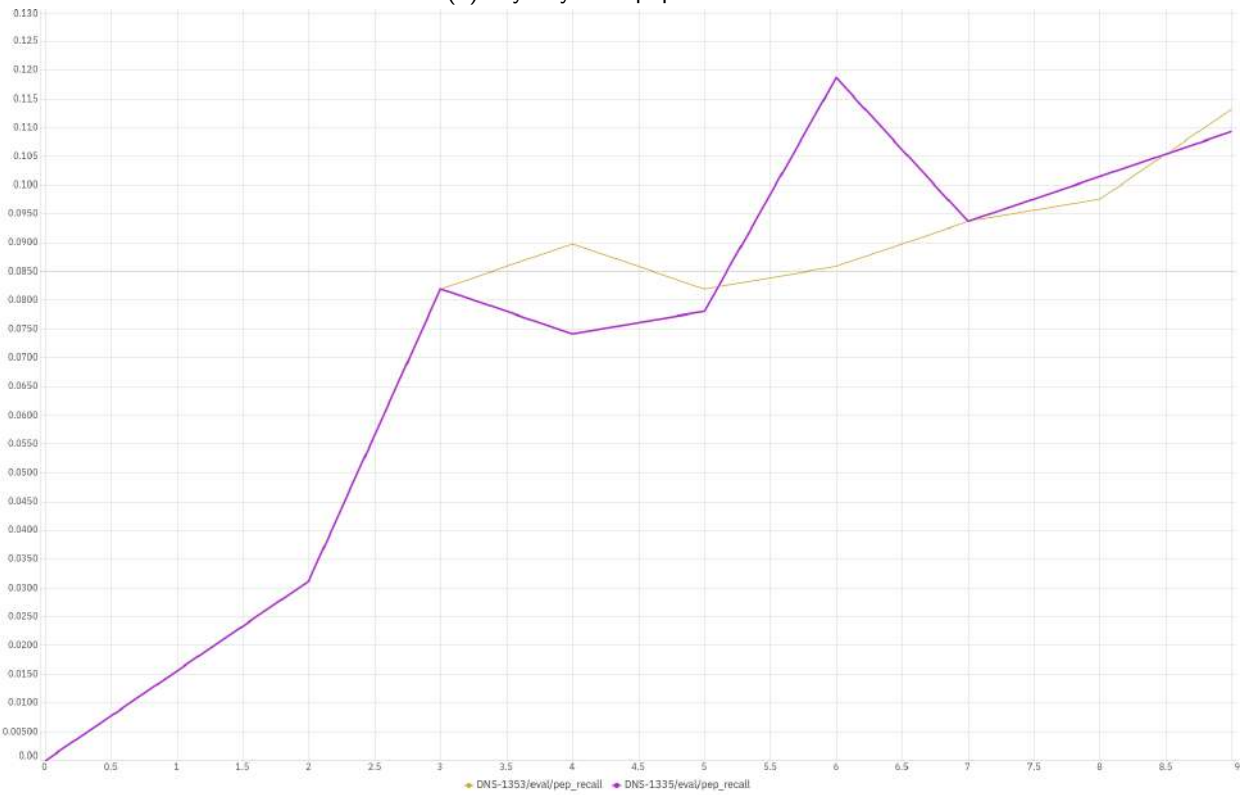
5.1 Other Figures

5.2 Repository

Some early analyses of the dataset are available in this GitHub repository, while later analyses were conducted on AIchor (InstaDeep, 2025).



(a) Glycosylated peptide recall



(b) Peptide recall

Figure 5.1: Comparison of glycopeptide recall and peptide recall between Adam (Yellow) and AdamW (Purple) optimizers during InstaNovo training, showing similar performance under identical configurations.

Acknowledgements

I would like to begin by acknowledging AIMS, Google DeepMind, all funding sponsors, Ulrich Paquet – the AIMS South Africa Director, and all the teachers who have taught us from around the world. Special thanks to Claire David, the Academic Director of the AI stream, for ensuring everything ran smoothly. In particular, I want to thank Arnu Pretorius, my reinforcement learning instructor, through whom I found my project.

A big thank you to Jeroen Van Goey, my principal supervisor, for always making time for me, even when I wasn't making much progress due to coursework. Thank you for your patience, steady guidance, and for believing in me while I was still finding my footing. I also appreciate how responsive you've been and how helpful it was during times where lectures and assignments took over everything.

To Kevin Michael Eloff, my supervisor, thank you for your insightful feedback and perspective.

To Konstantinos Kalogeropoulos, thank you for your quick replies, responsiveness, and helpful insights. I will never forget our first meeting. Maybe that is why *imposter syndrome* still follows me around, making me forget all the English I know in every team meeting.

To Jesper Lauridsen, thank you for always being available and for helping by highlighting important aspects of the project to keep an eye on, especially knowing how limited my time was. I truly appreciate your help and insights.

I also want to thank my other supervisors, Amandla Mabona, Rachel Catzel, and Jemma Daniel. Due to tight timing, I haven't had as many opportunities to interact as I would have liked, but I hope to get to know you better in the future. I'm also grateful to the entire InstaNovo team and everyone who has contributed to this work in one way or another.

I would also like to thank my family, especially my mother, for their unwavering support throughout this journey.

Finally, I want to express my appreciation to Florian Stimberg for taking the time to check in and talk through how things were going at AIMS: the pressure, the stress, the ups and downs.

References

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes, 2018. URL https://arxiv.org/abs/1610.01644.

An, H. J., Froehlich, J. W., and Lebrilla, C. B. Determination of Glycosylation Sites and Site-specific Heterogeneity in Glycoproteins. *Curr. Opin. Chem. Biol.*, 13(4):421, Aug. 2009. doi: 10.1016/j.cbpa.2009.07.022.

Bern, M., Kil, Y. J., and Becker, C. Byonic: Advanced Peptide and Protein Identification Software. *Current protocols in bioinformatics / editoral board, Andreas D. Baxevanis ... [et al.]*, CHAPTER: Unit13.20, Dec. 2012. doi: 10.1002/0471250953.bi1320s40.

Bobalova, J., Strouhalova, D., and Bobal, P. Common post-translational modifications (ptms) of proteins: Analysis by up-to-date analytical techniques with an emphasis on barley. *Journal of Agricultural and Food Chemistry*, 71(41):14825–14837, 2023. doi: 10.1021/acs.jafc.3c00886. URL https://doi.org/10.1021/acs.jafc.3c00886. PMID: 37792446.

Cao, B. Cellular components: Proteins and their crucial role in cell function. *Journal of Biochemistry Research*, 6(3):55–57–, Jun 2023. doi: https://doi.org/10.37532/oabr.2023.6(3).55-57. URL https://www.openaccessjournals.com/articles/cellular-components-proteins-and-their-crucial-role-in-cell-function-16465.html#:~:text=Proteins%20are%20essential%20cellular%20components.

Diker, A. Functions of Protein, dec 1 2022. [Online; accessed 2025-05-01].

DiMaggio, P. A., Jr. and Floudas, C. A. De novo peptide identification via tandem mass spectrometry and integer linear optimization. *Anal. Chem.*, 79(4):1433, Feb 2007. doi: 10.1021/ac0618425.

Dupree, E. J., Jayathirtha, M., Yorkey, H., Mihasan, M., Petre, B. A., and Darie, C. C. A Critical Review of Bottom-Up Proteomics: The Good, the Bad, and the Future of This Field. *Proteomes*, 8 (3):14, July 2020. ISSN 2227-7382. doi: 10.3390/proteomes8030014.

Eloff, K., Kalogeropoulos, K., Mabona, A., Morell, O., Catzel, R., Rivera-de Torre, E., Berg Jespersen, J., Williams, W., van Beljouw, S. P. B., Skwark, M. J., Laustsen, A. H., Brouns, S. J. J., Ljungars, A., Schoof, E. M., Van Goey, J., Auf dem Keller, U., Beguir, K., Lopez Carranza, N., and Jenkins, T. P. InstaNovo enables diffusion-powered de novo peptide sequencing in large-scale proteomics experiments. *Nat. Mach. Intell.*, 7:565–579, Apr. 2025. ISSN 2522-5839. doi: 10.1038/s42256-025-01019-5.

Fang, Z., Qin, H., Mao, J., Wang, Z., Zhang, N., Wang, Y., Liu, L., Nie, Y., Dong, M., and Ye, M. Glyco-Decipher enables glycan database-independent peptide matching and in-depth characterization of site-specific N-glycosylation. *Nat. Commun.*, 13(1900):1–15, Apr. 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-29530-y.

Frank, A. M., Savitski, M. M., Nielsen, M. N., Zubarev, R. A., and Pevzner, P. A. De Novo Peptide Sequencing and Identification with Precision Mass Spectrometry. *J. Proteome Res.*, 6(1):114, Jan. 2007. doi: 10.1021/pr060271u.

Gasser, J. Role of proteins: Complex molecules present in the human body. *Enz Eng.*, 12:207, 2023.

Gillet, L. C., Leitner, A., and Aebersold, R. Mass Spectrometry Applied to Bottom-Up Proteomics: Entering the High-Throughput Era for Hypothesis Testing. *Annu. Rev. Anal. Chem.*, (Volume 9, 2016):449–472, June 2016. doi: 10.1146/annurev-anchem-071015-041535.

Goldtzvik, Y., Sen, N., Lam, S. D., and Orengo, C. Protein diversification through post-translational modifications, alternative splicing, and gene duplication. *Current Opinion in Structural Biology*, 81:102640, 2023. ISSN 0959-440X. doi: https://doi.org/10.1016/j.sbi.2023.102640. URL https://www.sciencedirect.com/science/article/pii/S0959440X23001148.

Gong, T., Lee, T., Stephenson, C., Renduchintala, V., Padhy, S., Ndirango, A., Keskin, G., and Elibol, O. A comparison of loss weighting strategies for multi task learning in deep neural networks. *IEEE Access*, PP:1–1, 09 2019. doi: 10.1109/ACCESS.2019.2943604.

He, M., Zhou, X., and Wang, X. Glycosylation: mechanisms, biological functions and clinical implications. *Sig. Transduct. Target. Ther.*, 9(194):1–33, Aug. 2024. ISSN 2059-3635. doi: 10.1038/s41392-024-01886-1.

Howard, J. and Ruder, S. Universal Language Model Fine-tuning for Text Classification. *arXiv*, Jan. 2018. doi: 10.48550/arXiv.1801.06146.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Huang, T., Wang, J., Yu, W., and He, Z. Protein inference: a review. *Briefings Bioinf.*, 13(5):586–614, Sept. 2012. ISSN 1467-5463. doi: 10.1093/bib/bbs004.

Hughes, C., Ma, B., and Lajoie, G. A. De Novo Sequencing Methods in Proteomics. In *Proteome Bioinformatics*, pages 105–121. Humana Press, 2010. ISBN 978-1-60761-444-9. doi: 10.1007/978-1-60761-444-9_8.

InstaDeep. Homepage - AIchor, Mar. 2025. URL https://aichor.ai. [Online; accessed 25. Jun. 2025].

IUPAC-IUB. A one-letter notation for amino acid sequences (definitive rules), 1972. URL https://old.iupac.org/publications/pac/1972/pdf/3104x0639.pdf.

Kalra, D. S. and Barkeshli, M. Why warmup the learning rate? underlying mechanisms and improvements, 2024. URL https://arxiv.org/abs/2406.09405.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Kothlow, K., Schramm, H. M., Markuson, K. A., Russell, J. H., Sutherland, E., Veth, T. S., Zhang, R., Duboff, A. G., Tejus, V. R., McDermott, L. E., Dräger, L. S., and Riley, N. M. Extracting informative glycan-specific ions from glycopeptide MS/MS spectra with GlyCounter. *bioRxiv*, page 2025.03.24.645139, Mar. 2025. URL https://doi.org/10.1101/2025.03.24.645139.

LaPelusa, A. and Kaushik, R. Physiology, proteins, November 2022. Updated 2022 Nov 14. In: StatPearls [Internet]. Available from: https://www.ncbi.nlm.nih.gov/books/NBK555990/.

Lauridsen, J., Ramasamy, P., Catzel, R., Canbay, V., Mabona, A., Eloff, K., Fullwood, P., Ferguson, J., Kirketerp-Møller, A., Goldschmidt, I. S., Claeys, T., van Puyenbroeck, S., Lopez Carranza, N., Schoof, E. M., Martens, L., Van Goey, J., Francavilla, C., Jenkins, T. P., and Kalogeropoulos, K. Instanovo-p: A de novo peptide sequencing model for phosphoproteomics. *bioRxiv*, 2025. doi: 10.1101/2025.05.14.654049. URL https://www.biorxiv.org/content/early/2025/05/18/2025.05.14.654049.

Liu, M.-Q., Zeng, W.-F., Fang, P., Cao, W.-Q., Liu, C., Yan, G.-Q., Zhang, Y., Peng, C., Wu, J.-Q., Zhang, X.-J., Tu, H.-J., Chi, H., Sun, R.-X., Cao, Y., Dong, M.-Q., Jiang, B.-Y., Huang, J.-M., Shen, H.-L., Wong, C. C. L., He, S.-M., and Yang, P.-Y. pGlyco 2.0 enables precision N-glycoproteomics with comprehensive quality control and one-step mass spectrometry for intact glycopeptide identification. *Nat. Commun.*, 8(438):1–14, Sept. 2017. ISSN 2041-1723. doi: 10.1038/s41467-017-00535-2.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

Luo, Y., Yang, Z., Meng, F., Li, Y., Zhou, J., and Zhang, Y. An empirical study of catastrophic forgetting in large language models during continual fine-tuning, 2025. URL https://arxiv.org/abs/2308.08747.

Lyons, J., Milner, J., and Rosenzweig, S. Glycans instructing immunity: The emerging role of altered glycosylation in clinical immunology. *Frontiers in Pediatrics*, 3, 06 2015. doi: 10.3389/fped.2015.00054.

Ma, J. and Yarats, D. On the adequacy of untuned warmup for adaptive optimization, 2021. URL https://arxiv.org/abs/1910.04209.

Maliepaard, J. C., Damen, J. M. A., Boons, G.-J. P., and Reiding, K. R. Glycoproteomics-Compatible MS/MS-Based Quantification of Glycopeptide Isomers. *Analytical Chemistry*, 95(25):9605–9614, June 2023. ISSN 0003-2700, 1520-6882. doi: 10.1021/acs.analchem.3c01319. URL https://pubs.acs.org/doi/10.1021/acs.analchem.3c01319.

Mosin, V., Samenko, I., Kozlovskii, B., Tikhonov, A., and Yamshchikov, I. P. Fine-tuning transformers: Vocabulary transfer. *Artif. Intell.*, 317:103860, Apr. 2023a. ISSN 0004-3702. doi: 10.1016/j.artint.2023.103860.

Mosin, V., Samenko, I., Kozlovskii, B., Tikhonov, A., and Yamshchikov, I. P. Fine-tuning transformers: Vocabulary transfer. *Artificial Intelligence*, 317:103860, Apr. 2023b. ISSN 0004-3702. doi: 10.1016/j.artint.2023.103860. URL http://dx.doi.org/10.1016/j.artint.2023.103860.

Nevers, Y., Glover, N. M., Dessimoz, C., and Lecompte, O. Protein length distribution is remarkably uniform across the tree of life. *Genome Biol.*, 24(1):1–20, Dec. 2023. ISSN 1474-760X. doi: 10.1186/s13059-023-02973-2.

Oliveira, T., Thaysen-Andersen, M., Packer, N. H., and Kolarich, D. The Hitchhiker's guide to glycoproteomics. *Biochemical Society Transactions*, 49(4):1643–1662, Aug. 2021. ISSN 0300-5127, 1470-8752. doi: 10.1042/BST20200879. URL https://portlandpress.com/biochemsoctrans/article/49/4/1643/229276/The-Hitchhiker-s-guide-to-glycoproteomics.

Paidikondala, K. K. and Valivarthi, N. K. Protein: It's importance in human nutrition. *International Research Journal of Modernization in Engineering Technology and Science*, 6(4), apr 2024. URL https://www.irjmets.com/uploadedfiles/paper//issue_4_april_2024/53719/final/fin_irjmets1714026509.pdf.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library, 2019. URL https://arxiv.org/abs/1912.01703.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Perez-Riverol, Y., Bandla, C., Kundu, D. J., Kamatchinathan, S., Bai, J., Hewapathirana, S., John, N. S., Prakash, A., Walzer, M., Wang, S., and Vizcaíno, J. A. The pride database at 20 years: 2025 update. *Nucleic Acids Research*, 53(D1):D543–D553, 2025. doi: 10.1093/nar/gkae1011.

Polasky, D., Yu, F., Teo, G., and Nesvizhskii, A. Fast and comprehensive n- and o-glycoproteomics analysis with msfragger-glyco. *Nature Methods*, 17:1–8, 11 2020. doi: 10.1038/s41592-020-0967-9.

Rethmeier, N. and Augenstein, I. A Primer on Contrastive Pretraining in Language Processing: Methods, Lessons Learned and Perspectives. *arXiv*, Feb. 2021. doi: 10.48550/arXiv.2102.12982.

Sebastião, M. J., Marcos-Silva, L., Gomes-Alves, P., and Alves, P. M. Proteomic and Glyco(proteo)mic tools in the profiling of cardiac progenitors and pluripotent stem cell derived cardiomyocytes: Accelerating translation into therapy. *Biotechnol. Adv.*, 49:107755, July 2021. ISSN 0734-9750. doi: 10.1016/j.biotechadv.2021.107755.

Solntsev, S. K., Shortreed, M. R., Frey, B. L., and Smith, L. M. Enhanced Global Post-translational Modification Discovery with MetaMorpheus. *J. Proteome Res.*, 17(5):1844–1851, May 2018. ISSN 1535-3893. doi: 10.1021/acs.jproteome.7b00873.

Sun, W., Zhang, Q., Zhang, X., Tran, N. H., Ziaur Rahman, M., Chen, Z., Peng, C., Ma, J., Li, M., Xin, L., and Shan, B. Glycopeptide database search and de novo sequencing with PEAKS GlycanFinder enable highly sensitive glycoproteomics. *Nat. Commun.*, 14(4046):1–15, July 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-39699-5.

Tariq, U. and Bo. deep_learning_in_proteomics, June 2025. URL https://github.com/bzhanglab/deep_learning_in_proteomics. [Online; accessed 10. Jun. 2025].

Thaysen-Andersen, M. and Packer, N. H. Advances in LC–MS/MS-based glycoproteomics: Getting closer to system-wide site-specific mapping of the N- and O-glycoproteome. *Biochimica et Biophysica Acta (BBA) - Proteins and Proteomics*, 1844(9):1437–1452, Sept. 2014. ISSN 15709639. doi: 10.1016/j.bbapap.2014.05.002. URL https://linkinghub.elsevier.com/retrieve/pii/S1570963914001174.

Tomihari, A. and Sato, I. Understanding linear probing then fine-tuning language models from ntk perspective, 2024. URL https://arxiv.org/abs/2405.16747.

Van den Steen, P., Rudd, P. M., Dwek, R. A., and Opdenakker, G. Concepts and principles of o-linked glycosylation. *Crit. Rev. Biochem. Mol. Biol.*, 33(3):151–208, 1998.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention Is All You Need. *arXiv*, June 2017. doi: 10.48550/arXiv.1706.03762.

Wang, R., Shi, X., Pang, S., Chen, Y., Zhu, X., Wang, W., Cai, J., Song, D., and Li, K. Cross-attention guided loss-based deep dual-branch fusion network for liver tumor classification. *Information Fusion*, 114:102713, Feb. 2025. ISSN 1566-2535. doi: 10.1016/j.inffus.2024.102713.

Wen, B., Zeng, W.-F., Liao, Y., Shi, Z., Savage, S. R., Jiang, W., and Zhang, B. Deep Learning in Proteomics. *Proteomics*, 20(21-22):1900335, Oct. 2020. doi: 10.1002/pmic.201900335.

Wiener, E. Gradual Unfreezing, Feb. 2024. URL https://ericwiener.github.io/ai-notes/AI-Notes/Definitions/Gradual-Unfreezing. [Online; accessed 11. Jun. 2025].

Zhang, W. and Wang, T. Understanding Protein Functions in the Biological Context. *Protein Pept. Lett.*, 30(6):449–458, 2023. ISSN 1875-5305. doi: 10.2174/0929866530666230507212638.

Zolg, D. P., Wilhelm, M., Schmidt, T., Médard, G., Zerweck, J., Knaute, T., Wenschuh, H., Reimer, U., Schnatbaum, K., and Kuster, B. ProteomeTools: Systematic Characterization of 21 Post-translational Protein Modifications by Liquid Chromatography Tandem Mass Spectrometry (LC-MS/MS) Using Synthetic Peptides. *Mol. Cell. Proteomics*, 17(9):1850–1863, Sept. 2018. ISSN 1535-9484. doi: 10.1074/mcp.TIR118.000783.

Zong, Y., Wang, Y., Qiu, X., Huang, X., and Qiao, L. Deep learning prediction of glycopeptide tandem mass spectra powers glycoproteomics. *Nat. Mach. Intell.*, 6(8):1–12, July 2024. ISSN 2522-5839. doi: 10.1038/s42256-024-00875-x.