Diffusion-based Foundation Model For Mass Spectrometry Via Self-Supervised Learning on Unlabelled Data

Vicent Mwanda (vicent@aims.ac.za)
African Institute for Mathematical Sciences (AIMS)

Supervised by: Divanisha Patel, Jeroen Van Goey, Jemma Daniel, and Konstantinos Kalogeropoulos
InstaDeep, South Africa

11 June 2026

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

Abstract

The identification of mass spectra is key to proteomics, which is a technology used in a wide range of biological and medical applications. With the surge of machine learning applications in proteomics, this data needs to be processed into representations suitable for AI models to learn from. However, mass spectrum data contains noise from different sources and instrument configurations, which can reduce its utility in downstream machine learning models. Current approaches, such as the InstaNovo and Casanovo models, that use encoders to provide these representations in the form of embeddings are also limited, as they do not fully reduce the impact of noise in the data and originate from models optimized for the singular task of de novo peptide sequencing. In addition, some of these methods require supervised learning, which relies on labelled data, yet a large amount of mass spectrum data currently remains unlabelled. In this study, a diffusion-based encoder is trained using a self-supervised masked spectrum prediction task to learn robust representations of mass spectra from unlabelled data. The resulting embeddings should provide general-purpose spectrum representations that can be transferred to downstream proteomics tasks, reducing reliance on task-specific supervised models and enabling more effective use of large-scale mass spectrometry datasets. Overall, our model achieves an R^2 of 0.784 for precursor m/z at large-scale compared to the baseline (0.923), while learning compact and organised latent representations. This indicates a trade-off between representation compactness and predictive accuracy. For classification, the foundation model achieves a precursor charge accuracy of 0.647 on the full dataset, which is below the majority-class baseline of approximately 0.734 given the strong class imbalance (charge 2 dominates 73.4% of spectra), while the InstaDeep baseline achieves 0.942, indicating strong performance far above trivial baselines. For fragmentation type, the model achieves 0.624 compared to a majority-class baseline of 0.547 and a random baseline of 0.250, indicating marginal but non-trivial performance. Based on the small-scale ablations, the results suggest that the diffusion-based encoder using masking and the multi-loss objective supports learning organised and compact latent representations, however, additional mechanisms,such as parameter tuning at scale, are required to improve predictive performance of the current model, particularly under imbalanced classification settings and at a large dataset scale. In addition, our findings show that the centroid loss requires mechanisms, such as AdaLN modulation, to reduce its performance penalty on learned embeddings.

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.

Vicent Mwanda, 11 June 2026

Contents

1. Introduction

1.1 Context and Justification

Mass spectrometry is one of the key technologies in proteomics (Cho, 2007). As a science, proteomics focuses on the study of proteins and cellular function at large scale (Aebersold and Mann, 2003). Applications of this technology enable a wide range of biological and medical applications, such as disease biomarker discovery, and structural biology, which rely on protein identification as the first step (Eloff et al., 2025). However, the success of proteomics relies on the availability of analytical data, such as mass spectra, for proteins (Domon and Aebersold, 2006). This is why advances in mass spectrometry, such as improved peptide ionization methods and the development of more sensitive instrumentation, have been carried out to improve the availability of proteomics data (Aebersold and Mann, 2003).

The data provided by mass spectrometry is important as it supports downstream proteomic technologies, including machine learning applications, where it is used to train models for downstream tasks, such as de novo peptide sequencing in order to decode unknown amino acid sequences directly from observed spectral peaks (Bittremieux et al., 2026). Machine learning models, such as InstaNovo and Casanovo, have demonstrated promising results on this sequencing task (Eloff et al., 2025; Yilmaz et al., 2023). However, these models are still limited by the availability of both large datasets and computational resources. In terms of datasets, advances in mass spectrometry have led to growth in the volume of mass spectra–based proteomic data. However, a large portion of this data remains unlabelled, with studies reporting that an average of 75% of analysed spectra remain unidentified, which limits its use for tasks requiring labelled data (Griss et al., 2016; Ning et al., 2010). This limitation is especially evident in de novo peptide sequencing, where models such as InstaNovo must predict peptide sequences directly from spectra without relying on database matching (Eloff et al., 2025). The scarcity of labelled training data restricts these supervised models from learning from the growing amount of spectral data (Chai et al., 2024).

However, the abundance of unlabelled mass spectrometry data presents an opportunity to enhance de novo peptide sequencing tasks. Even without labels, spectral data contains rich structural and contextual information that can be exploited as shown by Bushuiev et al. (2025). Specifically, representation learning and foundation models have demonstrated that large collections of unlabelled data can be used to learn meaningful embeddings that capture the underlying patterns in the data (Syed, 2025).

Mass spectrum data is, however, affected by various sources of noise, including measurement variability, instrument configuration, and sample preparation processes (Du et al., 2008; Urban and Štys, 2015). As a result, raw spectral data may contain distortions that introduce undesirable signals for downstream analysis (Aebersold and Mann, 2003). One solution to this problem, therefore, is to learn robust representations from this data that account for the present noise. This is where diffusion-based models show promise for learning such representations (Jiang and Cai, 2025). These models learn through a process of gradually adding noise and then de-noising data in order to recover the structured signals from noisy inputs (Hoogeboom et al., 2021). This makes them well suited for refining spectral representations and generating robust embeddings from noisy data.

In this work, I present the development and evaluation of a diffusion-based foundation model for mass spectrometry using self-supervised learning on unlabelled data. The model is trained through self-supervised learning and masked prediction on large collections of unlabelled proteomic spectral data to learn contextual embeddings that capture the underlying patterns of proteomic spectra. These learned embeddings aim to improve the quality of spectral representations used in downstream proteomic tasks.

1.2 Objectives

The main objective of this study is to investigate whether a diffusion-based foundational encoder model trained through self-supervised learning and masked prediction on unlabelled mass spectrometry data can learn robust semantic representations that improve downstream proteomic tasks such as de novo peptide sequencing, retention time prediction, and post-translational modification detection.

The specific objectives are:

1. To develop a diffusion-based self-supervised framework for learning meaningful representations from noisy and unlabelled mass spectrometry spectral data.
2. To investigate the role of diffusion-based modelling in learning noise-robust and structure-aware spectral embeddings.
3. To evaluate the quality and robustness of the learned spectral representations using intrinsic and extrinsic analysis methods, including clustering structure, and linear probing for downstream tasks.

1.3 Contributions

This study makes the following contributions:

- **Automated representation learning for mass spectra using Masked Diffusion and Tokenization:** The framework used in this work was inspired by Jiang and Cai (2025), whose framework focused on automated representation learning on image data in a continuous setting with Gaussian noise applied in the forward process. In this work, we adapt this framework to mass spectra data through masked diffusion with spectra represented as sequences of peak tokens rather than image pixels. Jiang and Cai (2025) train their model using a single loss objective, L_{simple}. We replace this loss with a multi-objective function (section 3.3) because L_{simple} alone allows the diffusion model to prioritise reconstruction fidelity without enforcing structure in the learnt representations, which is a challenge for embeddings intended for general purpose downstream tasks.
- **Investigation of the impact of centroid-based regularisation on representation learning for mass spectra:** The centroid of the mass spectrum encodes a summary of its global distribution (Marty, 2022). In this work, we propose the use of the centroid loss (section 3.3 and 4.2), based on the spectral centroid, as a physics-informed constraint on the representation learning for mass spectra, and evaluate its effect on the learned latent space through ablation experiments; Ablation-10:Full+CENT and Ablation-11:Full. The results show that adding the centroid loss introduces a performance penalty across both classification and regression tasks, compared to the multi-loss objective model(Ablation-9: Full) without it.
- **Investigation of the impact of multi-loss objectives on representation learning for mass spectra:** We compare single-loss and multi-loss objectives configurations across eleven ablations (section 3.5.3 and 4.2) and show that the harder multi-loss objectives generally produce more organised latent representations and stronger downstream linear probe performance than simple reconstruction objectives. For example, Ablation-9:Full achieves an R^2 score of 0.880 for precursor m/z and an accuracy of 0.910 for precursor charge, compared to 0.734 and 0.874 respectively, for the simplest single-loss reconstruction objective, Ablation-1:DF. We also show that the quality of the semantic structure of the learned embeddings is mainly driven by the contrastive loss component (based on Ablation-5:CL-only), although the addition of the variance and contrastive losses can still reduce the linear predictability of the semantic properties.

1.4 Thesis Outline

This study is structured into four main chapters: Literature Review (Chapter 2), Methodology (Chapter 3), Results and Discussion (Chapter 4), and Conclusion (Chapter 5). The literature review chapter provides an overview of mass spectrometry in proteomics, diffusion-based generative models, and representation learning approaches, and with emphasis on their application to learning from unlabelled mass spectra. The methodology chapter describes the proposed diffusion-based self-supervised framework, including the dataset, preprocessing pipeline, masking strategies, model architecture, and the representation learning and embedding extraction process. The results and discussion chapter presents the evaluation of the model and learned representations, including sanity checks, ablation studies, UMAP visualisations, and linear probe analysis, as well as a comparison with baseline methods. Finally, the conclusion chapter summarises the key findings of the study, discusses limitations, and outlines directions for future work.

2. Literature Review

In this section, the proposed method is reviewed in relation to study domain and existing work. As diffusion models belong to the broader class of deep generative methods, a review of existing Deep Generative Models (DGM) is necessary to evaluate their suitability for mass spectra prediction and embedding extraction. These methods can be categorised into variational models (Denoising Diffusion Probabilistic Model(DDPM) and Denoising Diffusion Implicit Model(DDIM)), energy-based models (Score-based diffusion), and normalising flows (Flow matching and categorical flows).

2.1 Study Domain

This section focuses on the key relationships for mass spectrometry data relevant to machine learning problems. Mass spectrometry data is generated through ionisation, separation, and detection processes that produce a high dimensional spectrum where ions are represented by their mass-to-charge ratio (m/z) and corresponding intensities (Levner, 2005). Understanding the structural characteristics of spectral data is important for incorporating prior physical knowledge into machine learning models, for example through physics-informed regularisation of mass spectra prediction (Tbahriti et al., 2026).

One important property of a mass spectrum is its centroided peaks, denoted by m_c, which represent the mean position of spectral peaks. According to Marty (2022), the centroid is defined as:

$$m_c = \frac{\sum_i m_i I_i}{\sum_i I_i}, \tag{2.1.1}$$

where I_i denotes the intensity at index i, and m_i denotes the corresponding mass-to-charge ratio at index i. Since the centroid captures the overall mass distribution of a measured peak, it can serve as a physics-informed constraint as well as evaluation metric during training. In particular, a centroid-based loss function can be used to assess a model's ability to produce physically consistent spectral reconstructions, as inspired by Tbahriti et al. (2026).

2.2 Overview of Deep Generative Models (DGM)

The fundamental aim of a DGM is to optimally learn the distribution of a finite dataset of observations, x, within polynomial time (Lai et al., 2025). These models operate on the assumption that observations are sampled from an unknown complex true distribution, p_{data}, for which no analytical form is known. The DGM must therefore approximate p_{data} using a parametrised model p_ϕ, where ϕ represents the learnable weights of a deep neural network. To optimise p_ϕ, the model minimises the discrepancy measure, $D(p_{\text{data}}, p_\phi)$:

$$\phi^* \in \arg\min_\phi D(p_{\text{data}}, p_\phi), \tag{2.2.1}$$

such that $p_{\phi^*}(x) \approx p_{\text{data}}(x)$. This discrepancy measure varies by architecture. For instance, standard variational approaches utilise Kullback–Leibler (KL) divergence, while score-based methods often rely on Fisher divergence depending on the practical requirements of the modelling task as highlighted by Lai et al. (2025) and Liang et al. (2025).

According to Lai et al. (2025), to model the true data distribution, the parametric model, p_ϕ must constitute a valid probability density function by satisfying the following fundamental conditions:

- **Positivity**: $p_\phi(x) \geq 0$ for all x, often ensured by applying an exponential function to the model's raw output.
- **Normalisation**: The integral over the domain is required to be equal to one, which is achieved by dividing the original function by its integral over the entire space. It is important to make careful and practical considerations of this normalisation factor because in many methods such as energy based methods, it can be intractable.

In the context of mass spectral data, the DGM aims to approximate the complex distribution implicitly represented by the data used during training. Once this distribution has been learned, the model should be capable of accurately reconstructing peaks from noisy data. Predictions or novel samples can then be generated by drawing from the learned distribution p_ϕ, for example through Monte Carlo sampling, as discussed by Lai et al. (2025).

2.3 Diffusion Models

2.3.1 Denoising Diffusion Probabilistic Models (DDPM)

Similar to Variational Autoencoders (VAEs), which use latent variables to represent data and are optimised using the lower bound of the log likelihood, DDPMs are rooted in variational frameworks,but are distinguished by two distinct stochastic processes (Lai et al., 2025; Nakkiran et al., 2024). The first process is a forward process, which can be seen as a fixed encoder that corrupts the original input or data distribution by progressively adding noise until the distribution becomes a simple prior distribution, $p_{\text{prior}} = \mathcal{N}(0, \mathbf{I})$. At each step, the forward process is governed by the following equation.

$$q(x_t \mid x_{t-1}) \coloneqq \mathcal{N}\Big(x_t;\, \sqrt{1-\beta_t}\, x_{t-1},\, \beta_t \mathbf{I}\Big)\,, \tag{2.3.1}$$

where $x_t \in \mathbb{R}^D$, D is the dimension of input into the model, t is the diffusion step, $\beta_t \in (0, 1)$ controls the noise schedule and $t \in \{1, \ldots, T\}$, T represents the number of diffusion steps required to reach p_{prior}, and $\mathbf{I}$ is an identity matrix. According to Lai et al. (2025) and Liang et al. (2025), when the noising process is applied recursively, the distribution of the noisy samples, x_t conditioned on the original data, x_0 has the closed-form expression

$$q(x_t \mid x_0) = \mathcal{N}\Big(x_t; \sqrt{\bar{\alpha}_t}\, x_0, (1-\bar{\alpha}_t)\mathbf{I}\Big)\,, \tag{2.3.2}$$

where $\bar{\alpha}_t \coloneqq \prod_{k=1}^{t}(1-\beta_k)$. Consequently, x_t can be sampled directly from this distribution as:

$$x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, \epsilon, \tag{2.3.3}$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$. According to Liang et al. (2025), the choice for the form of β_t used in above equations is made such that as $\bar{\alpha}_t \to 0$ then x_t asymptotically approaches the standard Gaussian distribution, p_{prior}.

The second process then involves a neural network (learnable decoder) that reverses the noise corruption through a parametrised distribution. This component progressively denoises the data representation until a coherent and meaningful data representation is obtained (Lai et al., 2025; Nakkiran et al., 2024).

In practice, the learnable decoder is treated as a parametric model, and is trained to minimise the KL divergence

$$\mathbb{E}_{q(x_t)}\left[D_{\mathrm{KL}}(q(x_{t-1} \mid x_t) \,\|\, p_\phi(x_{t-1} \mid x_t))\right]. \tag{2.3.4}$$

As shown in equation 2.3.4 above, the training objective encourages the reverse process, $p_\phi(x_{t-1} \mid x_t)$ to approximate the true reverse diffusion dynamics. However, computing the transition distribution, $q(x_{t-1} \mid x_t)$ directly is often computationally challenging because it involves evaluating an unknown marginal data distribution. To address this issue, the reverse diffusion process is reformulated by conditioning on the clean data sample x_0, yielding a tractable posterior distribution (Lai et al., 2025; Nichol and Dhariwal, 2021):

$$q(x_{t-1} \mid x_t, x_0) = \frac{q(x_t \mid x_{t-1})\, q(x_{t-1} \mid x_0)}{q(x_t \mid x_0)}. \tag{2.3.5}$$

As shown by Ho et al. (2020) and Lai et al. (2025), equation 2.3.5 leads to the simplified training objective

$$\mathcal{L}_{\text{simple}}(\phi) = \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon_\phi(x_t, t) - \epsilon\|_2^2\right], \tag{2.3.6}$$

where $\epsilon_\phi(x_t, t)$ is the predicted noise. After training the learnable decoder, sampling for new data is performed sequentially. In particular, the denoised estimate of the clean sample, x_0 can be recovered from the predicted noise, $\epsilon_\phi(x_t, t)$ as follows:

$$x_0 = \frac{x_t - \sqrt{1-\bar{\alpha}_t}\,\epsilon_\phi(x_t, t)}{\sqrt{\bar{\alpha}_t}}. \tag{2.3.7}$$

Relating DDPMs to the mass spectra prediction problem, the traditional DDPM is suitable for continuous spectral data. However, a key limitation of DDPM is that its sampling process is inherently slow (Lai et al., 2025). According to Nichol and Dhariwal (2021), the diffusion models trained on image generation task required 4000 steps which led to generating a single sample taking several minutes. In the case of foundation models, the training data is typically on the order of millions of samples (Sanders et al., 2025). The direct use of DDPMs is therefore constrained by their slow sampling rate in the mass spectrum prediction task. Some of the solutions to this limitation explored include reduction of sampling steps to fewer evenly spaced steps with output quality considerations as well as using Denoising Diffusion Implicit Models (DDIM) as discussed in the following section 2.3.2.

2.3.2 Denoising Diffusion Implicit Models (DDIM)

As mentioned,a significant limitation of the DDPM is its inference speed because the Markovian nature of its process requires numerous steps, leading to costly sampling operations. To address this, DDIM was proposed to improve the sampling rate in (Song et al., 2022). By utilising a non-Markovian forward process, DDIM allows for deterministic sampling and fewer steps, making it more practical for large-scale spectral prediction tasks where computational efficiency is a priority (Lai et al., 2025).

According to Lai et al. (2025), DDIM is an ordinary differential equations (ODE) based solver. To understand how DDIM works, there is a need to consider the principles of a common framework for designing numerical solvers using the semi-linear structure for a general drift function, $f(x(t), t) \coloneqq f(t)x$ for $f : \mathbb{R} \to \mathbb{R}$, which is induces the following Probabilistic Flow ODE (PF ODE):

$$\frac{dx(t)}{dt} = \underbrace{f(t)x(t)}_{\text{linear part}} + \underbrace{\frac{1}{2}\frac{g^2(t)}{\sigma_t}\,\epsilon_{\phi^\times}(x(t), t)}_{\text{non-linear part}}, \tag{2.3.8}$$

where:

$$g^2(t) = 2\sigma_t\sigma_t' - 2\frac{\alpha_t'}{\alpha_t}\sigma_t^2,$$

$$f(t) = \frac{\alpha_t'}{\alpha_t},$$

$$\alpha_t' := \frac{d\alpha_t}{dt} \quad \text{and} \quad \sigma_t' := \frac{d\sigma_t}{dt},$$

α_t and σ_t are diffusion coefficients,

$\epsilon_{\phi^\times}(x(t), t)$ is the predicted noise.

This linear to non-linear split structure in x provides for accuracy and stability for the model, which aids in the derivation of the trajectory necessary for step reduction as shown in Fig. 2.1. To achieve the reduction step approximation using DDIM, two time steps t and s are used where $s > t$ such that integrand for noise, $\epsilon_{\phi^\times}(x_\tau, \tau)$ for all $\tau \in [t, s]$ in the trajectory equation is approximated by $\epsilon_{\phi^\times}(x_s, s)$. Finally, with the integrand fixed at s, the trajectory becomes an Euler-style update of the form

$$x_t = \varepsilon(s \to t)\,\tilde{x}_s + \left(\frac{1}{2}\int_s^t \frac{g^2(\tau)}{\sigma_\tau}\,\varepsilon(\tau \to t)\,d\tau\right)\epsilon_{\phi^\times}(\tilde{x}_s, s), \tag{2.3.9}$$

where the exponential integrator, $\varepsilon(s \to t) := \exp\left(\int_s^t f(u)\,du\right)$. When this equation is simplified and parametrised with deterministic oracles such that $\epsilon_{\phi^\times} \approx \epsilon^*$ and $x_{\phi^\times} \approx x^*$, a single Euler step yields the closed-form DDIM update (Lai et al., 2025):

$$x_t = \alpha_t\,\tilde{x}^*(\tilde{x}_s, s) + \sigma_t\,\tilde{\epsilon}^*(\tilde{x}_s, s). \tag{2.3.10}$$

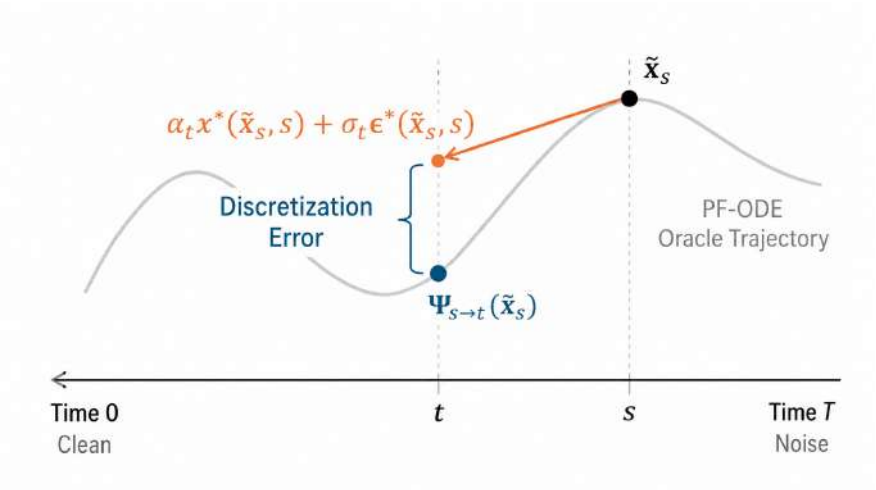


Figure 2.1: **Illustration of DDIM as an Euler discretization of the PF-ODE.** The true PF-ODE oracle (grey curve) trajectory evolves from the state, $\tilde{x}_s$ to $\Psi_{s\to t}(\tilde{x}_s)$, while the DDIM update (orange curve) maps $\tilde{x}_s$ directly to $\alpha_t x^*(\tilde{x}_s, s) + \sigma_t \epsilon^*(\tilde{x}_s, s)$. The two trajectories have a discrepancy that leads to a discretization error, and if t is distant from s, then the quality of samples generated drops (Lai et al., 2025).

While the DDIM helps to mitigate the inference speed limits of DDPM, DDIM from equation 2.3.10 utilises first-order exponential–Euler discretization, which has a global error of $O(h)$, as highlighted

in (Song et al., 2022; Lai et al., 2025). While this error can be reduced by considering higher-order schemes $O(h^k)$ with $k \geq 2$, such improvements do not necessarily guarantee a reduction in the number of required sampling steps (Lai et al., 2025). For the mass spectra problem, this highlights a critical trade-off between the numerical precision of the reconstructed peaks and the computational budget required for high-throughput spectral analysis.

2.3.3 Score-based Diffusion Methods

Score-based diffusion methods are derived from energy-based methods (EBM), which rely on a probability distribution based on an energy function, $E_\phi(x)$, that maps lower energy to the most probable data points (Lai et al., 2025). In particular, the score-based approach is unique in that it solves the problem of the intractable partition function, Z_ϕ inherent in the energy-based model

$$p_\phi(x) := \frac{\exp\left(-E_\phi(x)\right)}{Z_\phi}, \tag{2.3.11}$$

where $Z_\phi := \int \exp\left(-E_\phi(x)\right) dx$. Score-based methods achieve this by introducing the score function

$$s(x) := \nabla_x \log p(x), \tag{2.3.12}$$

which is defined as the gradient of the log-density of $p(x)$. This is possible because the gradient of the log-partition function is zero with respect to x. The score function can be modelled directly using a deep neural network without requiring normalisation. In this way, the scored-based approaches also have an added advantage of providing a complete representation as they fully characterises the underlying distribution in the data. In practice, a parametric score-based model, $s_\phi(x)$, is used to approximate this distribution.

To train $s_\phi(x)$, the score matching technique is used by attempting to approximate the score function, $s(x) = \nabla_x \log p_{\text{data}}(x)$ for the samples of p_{data}, by minimising the mean squared error

$$\mathcal{L}_{\text{SM}}(\phi) := \frac{1}{2}\,\mathbb{E}_{x\sim p_{\text{data}}}\left[\|s_\phi(x) - s(x)\|_2^2\right]. \tag{2.3.13}$$

However,this equation is intractable due to the unknown true score, $s(x)$ which is the target. According to Lai et al. (2025), the solution to this problem is a tractable objective function

$$\tilde{\mathcal{L}}_{\text{SM}}(\phi) := \mathbb{E}_{x\sim p_{\text{data}}(x)}\left[\text{Tr}(\nabla_x s_\phi(x)) + \frac{1}{2}\,\|s_\phi(x)\|_2^2\right], \tag{2.3.14}$$

derived through integration by parts. Once the trained score model, $s_{\phi^\times}(x)$ is obtained, it can used to replace the oracle score in the Langevin dynamics

$$x_{n+1} = x_n + \eta\, s_{\phi^\times}(x_n) + \sqrt{2\eta}\,\epsilon_n, \tag{2.3.15}$$

where $\epsilon_n \sim \mathcal{N}(0, \mathbf{I})$, $n = 0, 1, 2, \ldots$, initialised at x_0 and η is the step size. This expression can then be used for sampling the new data , x_{n+1} (Lai et al., 2025; Song et al., 2021).

Applied to mass spectrum prediction, score-based methods are more likely to understand the patterns in the data compared to DDPM because they particularly directly focus on the "shape" of the data distribution (Nichol and Dhariwal, 2021; Song and Ermon, 2019). Specifically, the score-based models directly learn the score function, $\nabla_x \log p(x)$ which points to regions of high data likelihood. In contrast, DDPMs reconstruct the data gradually from noisy data without explicitly considering the gradient of the true distribution leading to less precise restoring. However, score-based methods are also very slow because they require several evaluations of score network by the numerical solver (Jolicoeur-Martineau et al., 2021).

2.3.4 Discrete diffusion and Categorical diffusion

Discrete Diffusion Models are another class of diffusion models intended to address the limitations of standard diffusion approaches on discrete data tasks such as text generation, molecule design and categorical data generation. In particular, standard diffusion models and their guiding mechanisms rely on computing gradients, which is not feasible in a discrete setup (Schiff et al., 2025). Even discrete extensions of these models are not ideal for controllable generation. Furthermore, the perplexity performance of standard diffusion models still lags behind that of autoregressive models (Schiff et al., 2025; Feng et al., 2025).

Discrete Diffusion Models can be categorised into two primary architectures according to the nature of the reverse process. The first includes Masked Diffusion Models (MDMs), which begin with masked sequences, while the second uses a reverse process that starts from sequences of tokens sampled randomly from a vocabulary or discrete state space (Feng et al., 2025). In this study, the focus is placed on MDMs because they are more closely related to the masked spectrum prediction problem. It is worth noting that Discrete Diffusion Models become particularly relevant when binning is applied to mass spectra, thereby discretizing the spectra into bins (de Jonge et al., 2025).

MDMs can be formulated within the Discrete Denoising Diffusion Probabilistic Models (D3PMs) framework, which defines both forward and reverse processes based on earlier works (Yu et al., 2025; Feng et al., 2025). To better understand MDMs, consider a diffusion process over categorical one-hot vectors, y with K classes such that, $y \in \{0,1\}^K$ with $\sum_{i=1}^{K} y_i = 1$ (Schiff et al., 2025; Yu et al., 2025; Li et al., 2025). The forward process then relies on a time-dependent stochastic transition matrix, $Q_{t|s} \in \mathbf{R}^{K \times K}$, whose (i,j)-th entry is the probability of the transition to the i-th state at time t, given the j-th state at time s (Schiff et al., 2025; Yu et al., 2025). A Markov corruption process is induced over this matrix as follows:

$$q(y_t \mid y_s) = \mathrm{Cat}(y_t;\, \pi = Q_{t|s}\, y_s), \tag{2.3.16}$$

where $\mathrm{Cat}(\cdot;\pi)$ denotes the categorical distribution with probability vector $\pi \in \Delta^K$, and Δ^K is a probability simplex such that its categorical space, $V = \{y \in \{0,1\}^K\} \subset \Delta^K$. Using the D3PM framework then involves a forward Markov process

$$q(y_{1:T} \mid y_0) = \prod_{t=1}^{T} q(y_t \mid y_{t-1}), \tag{2.3.17}$$

that gradually corrupts the data into noise, and a parametrised reverse process

$$p_\theta(y_{0:T}) = p(y_T) \prod_{t=1}^{T} p_\theta(y_{t-1} \mid y_t), \tag{2.3.18}$$

that learns to denoise the data (Yu et al., 2025; Schiff et al., 2025). In order to simplify the processes in equations 2.3.17 and 2.3.18, a simplified Mask diffusion Model is used in (Yu et al., 2025). In this approach, the forward process replaces the input tokens with a special ([MASK]) token, and any masked token remains throughout this process. The forward transition $q(y_t \mid y_0)$ for any token, y at each time step, t can then be defined as shown below.

$$q(y_t \mid y_0) = \mathrm{Cat}(y_t;\, \alpha_t y_0 + (1-\alpha_t)m), \tag{2.3.19}$$

where $m \in \{0,1\}^K$ is the one-hot vector corresponding to the [MASK] token, and $\alpha_t \in [0,1]$ is the monotonically decreasing schedule such that $\alpha_0 \approx 1$ and $\alpha_T = 0$. The reverse process then aims to denoise the masked sequence by replacing the [MASK] tokens with predicted ones (Yu et al., 2025). In

addition, the unmasked tokens also remain unchanged during this reverse process, such that the reverse posterior at a previous time, s is then conditioned on y_t and y_0 as shown below:

$$q(y_s \mid y_t, y_0) = \begin{cases} \mathrm{Cat}(y_s;\, y_t), & \text{if } y_t \neq m, \\ \mathrm{Cat}\left(y_s;\, \dfrac{(1-\alpha_s)m + (\alpha_s - \alpha_t)y_0}{1-\alpha_t}\right), & \text{if } y_t = m \end{cases} \tag{2.3.20}$$

According to Yu et al. (2025), when a neural network, $f_\phi(y_t)$ is used to predict the original data, y_0 from the noisy data, y_t, this transition equation can rewritten as shown below.

$$p_\phi(y_s \mid y_t) = \begin{cases} \mathrm{Cat}(y_s;\, y_t), & \text{if } y_t \neq m, \\ \mathrm{Cat}\left(y_s;\, \dfrac{(1-\alpha_s)m + (\alpha_s - \alpha_t)f_\phi(y_t)}{1-\alpha_t}\right), & \text{if } y_t = m \end{cases} \tag{2.3.21}$$

Using this equation, the training loss, L over the masked tokens can then be written as:

$$\mathcal{L} = \sum_{t=2}^{T} \mathbb{E}_{y_0, y_{1:T}} \left[-\frac{\alpha_{t-1} - \alpha_t}{1-\alpha_t} \sum_{n=1}^{N} \delta_m(y_{t,n})\, y_{0,n}\, \log \left[f_\phi(y_t)\right]_n \right]. \tag{2.3.22}$$

where T is total number of time steps, $\delta_m(y_t, n)$ denotes the indicator function, $\delta_m(y_t, n) = 1$, if the n-th token of y_t is a masked token, otherwise, $\delta_m(y_t, n) = 0$.

For the masked mass spectrum prediction problem, Masked Diffusion Models (MDMs) are suitable because the data is transformed into a fixed-length representation through binning, which enables token-based modelling of spectra. For the mass spectrum data, the binning of the spectra is a necessary preprocessing step because it creates fixed-length vectors that are suitable for machine learning models (de Jonge et al., 2025). It is important to understand this process in relation to masked diffusion problem. The typical binning process involves selecting the number of bins, D, which determines the width of each bin

$$\Delta = \left(\frac{m_{\max} - m_{\min}}{D} \right), \tag{2.3.23}$$

where m_i is mass to charge ratio(m/z), $m_{\min}$ and $m_{\max}$ are the minimum and maximum mass-to-charge ratios respectively. Next, the peaks are assigned to bins, and the corresponding value, x_k for each bin is obtained by aggregating the peaks assigned to it, as shown below:

$$x_k = \sum_{i:\, m_i \in B_k} I_i \,. \tag{2.3.24}$$

where x_k is the value for bin k, I_i is intensity at m_i, and B_k denotes the k-th bin interval. From the above steps, it is clear that the binning process converts each spectrum into a fixed-length representation $x \in \mathbb{R}^D$. This representation, therefore, induces a discrete structure over which masked diffusion models are well-suited, since each bin can be treated as a token in a structured sequence under a masking-based corruption process. In addition, masking enables the transformer based model to learn generalizable representations of mass spectra in the presence of missing peak values resulting from stochastic instrument sampling and varying peptide concentrations (Zhou et al., 2026; Fröhlich et al., 2024).

However, the masked discrete diffusion model (MDM) framework, while well-suited for categorical binned spectra, can be affected by the choice of bin size, which influences the information content of the

model input and, in turn, the reconstruction fidelity (Madiona et al., 2019). Additionally, capturing dependencies across distant bins can be more challenging due to the computational cost of processing high-dimensional joint distributions, particularly in discrete settings where relationships between elements must be learned explicitly, similar to diffusion models in other domains such as images and language (Hayakawa et al., 2025).

2.4 Flow matching and Categorical flow matching

An alternative generative Markov framework to diffusion is Flow Matching (FM), which relies on deterministic dynamics (Patel et al., 2024). Flow Matching offers a regression-based approach to estimate the underlying vector of a continuous normalising flow directly over the time interval $t \in [0, 1]$ (Eijkelboom et al., 2025; Zhao et al., 2026). FM exploits the fact that the true underlying vector, u_t, and the probability path, p_t are generally unknown, however, it is possible to compute the gradient using a conditional flow formulation per example, as shown below.

$$u_t(x) = \int u_t(x \mid x_1) \frac{p_t(x \mid x_1) p_{\text{data}}(x_1)}{p_t(x)} \, dx_1 = \int u_t(x \mid x_1) \, p_t(x_1|x) \, dx_1 \tag{2.4.1}$$

Compared to diffusion models, FM is simulation-free, making it a more scalable approach for continuous normalising flows (Lipman et al., 2022). Furthermore, if $p_t(x_1|x)$ is replaced with a variational distribution q_t^θ parametrised by θ, the setup becomes a faster variational flow, as shown in equation 2.4.2 below, which relies on that fact if $q_t^\theta(x_1 \mid x)$ and $p_t(x_1|x)$ are identical then $v_t^\theta(x)$ is equal to $u_t(x)$ (Eijkelboom et al., 2025).

$$v_t^\theta(x) := \int u_t(x \mid x_1) \, q_t^\theta(x_1 \mid x) \, dx_1 \tag{2.4.2}$$

While traditional flow matching is suitable for continuous data, it is challenged by masked spectrum prediction task as the task presents discrete categorical data as highlighted in section 2.3.4 above. According to Eijkelboom et al. (2025), one approach to address it limitation is categorical flow matching (CatFlow). With CatFlow, the model learns a categorical simplex distribution over the possible outcomes of each discrete component, rather than regressing directly in a continuous space. The variational distribution for CatFlow also becomes a categorical distribution:

$$q_t^\theta(x_1^d \mid x) = \text{Cat}\left(x_1^d | \boldsymbol{\theta}_t^d(x)\right). \tag{2.4.3}$$

Each data point also is represented as a one-hot vector on the categorical simplex, and the flow is trained to map a simple base distribution, $q_t^\theta(x_1^d \mid x)$ to the target categorical distribution while respecting the discrete structure:

$$\text{L}_{\text{CatFlow}}(\theta) = -\mathbb{E}_{t,x,x_1}\left[\sum_{d=1}^{D_x}\sum_{k=1}^{K_d} \mathbb{I}[x_1^d = k] \log \mu_t^{d,k}(x)\right], \tag{2.4.4}$$

where k refers to a specific category, d represents a component of the learned vector field, and K is the total number of categories, $\mu_t^{d,k}(x) = \mathbb{E}_{q_t^\theta(x_1|x)}\left[\mathbb{I}[x_1^d = k]\right]$ describes the parameters of the categorical distribution. It is important to note that while FM and CatFlow utilise deterministic paths that result in faster convergence, they are less resistant to error accumulation compared to diffusion models as a result of using these deterministic paths (Yuan et al., 2026).

2.5 Diffusion on structured/sequential data

Diffusion on structured or sequential data is important for tasks working on data with correlated elements such as time-series or mass spectra (Cao et al., 2025). In this setting, the input data consists of ordered elements where dependencies exist across positions, and the model is required to capture both local and global structure.

However, this limitation can be overcome using attention mechanisms. According to Wang et al. (2025), using multi-head attention mechanism on spectra enabled a Diffusion Molecule Transformer (DMT) to capture spatial dependencies in input data. Using attention, the reverse diffusion process can, therefore, capture both local spectral structure and long-range dependencies, which is important in structured domains such as mass spectrometry data.

2.6 Representation extraction from Diffusion Models

As mentioned, representation extraction is key to the solution required for masked spectrum prediction. While representation learning for tandem mass spectrometry has already been explored using methods like the transformer-based network of Bushuiev et al. (2025), which were pre-trained in a self-supervised manner on millions of unlabelled spectra, diffusion approaches offer a better way to generate refined representations. As highlighted by Jiang and Cai (2025), diffusion provides powerful representations and a theoretical foundation capable of understanding semantic content.

An existing approach for representation extraction is through latent diffusion models, where an encoder–decoder architecture is used to map the input spectrum x into a latent vector z. The diffusion process is then applied in this latent space, and the latent variable z serves directly as the embedding, providing a compact and structured representation of the data (Jiang and Cai, 2025; Chen et al., 2024). According to Jiang and Cai (2025), a more dynamic representation can be obtained by extending Diffusion Transformers (DiT) to encode a time-step dependent vector, v_t^s for each noise level using a single neural network, $W_\mu(x_0, t)$ with parameters μ. This approach enables the compression of multi-level semantic information at different levels into the directional vectors, v_t^s. These vectors act as explicit self-conditions that guide the denoising DiT core, assisting it in recovering the data, x_0 directly from the corrupted state, yielding an estimate, $\hat{x}_t$ of the original sample, as shown in Fig. 2.2.

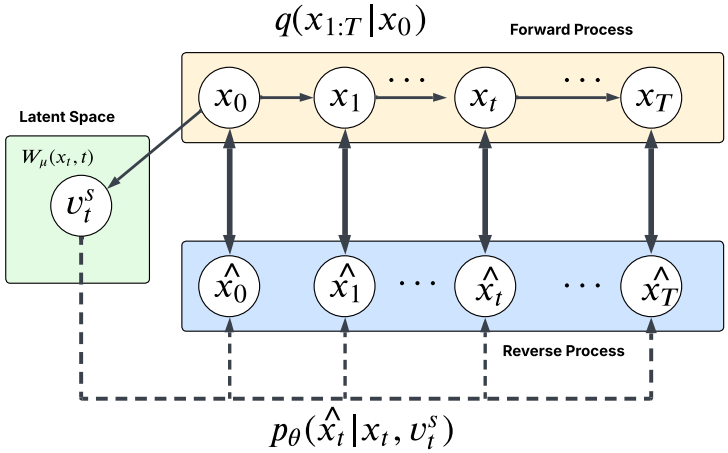


Figure 2.2: Illustration of the representation learning process (Jiang and Cai, 2025).

2.7 Comparative Summary of DGMs for Masked Spectrum Prediction

In this section, a comparative summary of generative models in relation to the masked spectrum prediction task is presented. The aim is to guide the rationale for selecting an appropriate DGM for this study. The key strengths and limitations of each model are summarised in Table 2.1.

Table 2.1: Comparative summary of generative models for masked spectrum prediction

Model	Data Type	Key Strengths	Limitations
DDPM	Continuous	Strong generative capability (Song et al., 2022; Lai et al., 2025); Good for continuous spectra	Slow sampling (requires many steps) (Song et al., 2022; Lai et al., 2025).
DDIM	Continuous	Faster inference than DDPM (Song et al., 2022).	First-order discretisation (accuracy vs step trade-off (Lai et al., 2025).
Score-Based Diffusion	Continuous	Captures data distribution gradients which is good for masked spectrum restoration(Lai et al., 2025).	Step size tuning required; sampling still costly (Jolicoeur-Martineau et al., 2021; Lai et al., 2025).
MDM	Discrete	Handles binned/categorical spectra suitable for masked spectrum (Schiff et al., 2025); More robust to noise/ missing data than CatFlow (Yuan et al., 2026).	Sequence error rate lags behind autoregressive models (Feng et al., 2025);
FM	Continuous	Very fast training and sampling (Lipman et al., 2022).	Requires careful parametrisation; Not suitable for discrete data (Gat et al., 2024).
CatFlow	Discrete	Efficient discrete data modelling; Converges faster than FM (Eijkelboom et al., 2025).	Can suffer error accumulation due to deterministic paths. (Yuan et al., 2026)

Based on the comparison of methods, **Masked Discrete Diffusion Model (MDM)** is the most appropriate choice for the masked spectrum prediction problem due to its direct compatibility with the discretised representation of binned MS/MS spectra. In particular, MDM naturally operates on categorical or token-based inputs, making it well-suited for modelling structured peak sequences under masking and reconstruction objectives. Furthermore, this kind of model provides robustness to inherent noise and missing peak information in the data. According to Feng et al. (2025), MDMs can achieve a near-optimal Token Error Rate (TER) using a fixed number of sampling steps, regardless of sequence length. Compared to CatFlow, MDMs benefit from stochastic corruption and denoising processes, which reduce the risk of error accumulation associated with deterministic generation paths (Yuan et al., 2026). While MDMs typically incur higher sampling costs compared to some alternative discrete methods, this is an acceptable trade-off given their improved adaptability to different masking strategies and their stability in noisy spectral settings.

3. Materials and Methods

3.1 Dataset

The dataset used in this study consists of tandem mass spectrometry (MS/MS) spectra obtained from large-scale proteomics experiments (Wen and Noble, 2024). Each spectrum in the dataset represents the fragmentation pattern of a peptide precursor ion in the form of a collection of detected peaks described by their mass-to-charge ratio (m/z) and intensity. These two quantities are important to the machine learning task because they define the structure of the mass spectrum sample as illustrated by the average spectrum shown in Fig. 3.1.

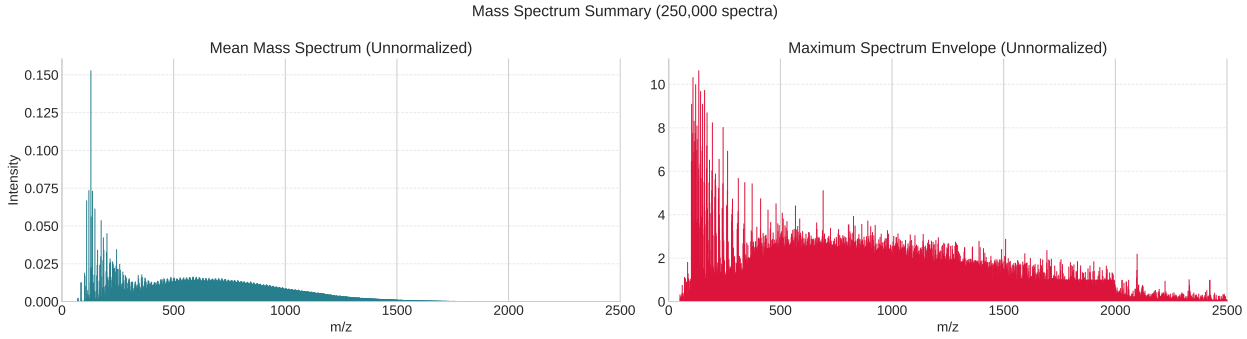


Figure 3.1: Average and maximum spectra based on 250,000 randomly sampled spectra from the training dataset.

In addition to spectral peaks, the dataset also contains several metadata fields associated with each spectrum, including precursor mass, precursor charge state, precursor intensity, retention time, collision energy, hyperscore, peptide sequence, peptide length, and fragmentation method such as higher-energy C-trap dissociation (HCD), and collision-induced dissociation (CID). These metadata fields are useful during downstream evaluation of the learned embeddings because they provide contextual information about the spectra and can help assess the type of structural information captured by the model representations. The metadata also exhibits meaningful correlations, as shown in Fig. 3.2, such as between precursor charge and sequence length, and precursor m/z and sequence length. These relationships reflect known physical and biochemical properties of peptide fragmentation, indicating that the dataset contains structured dependencies that are expected to be captured in the learned embeddings by the diffusion-based encoder.

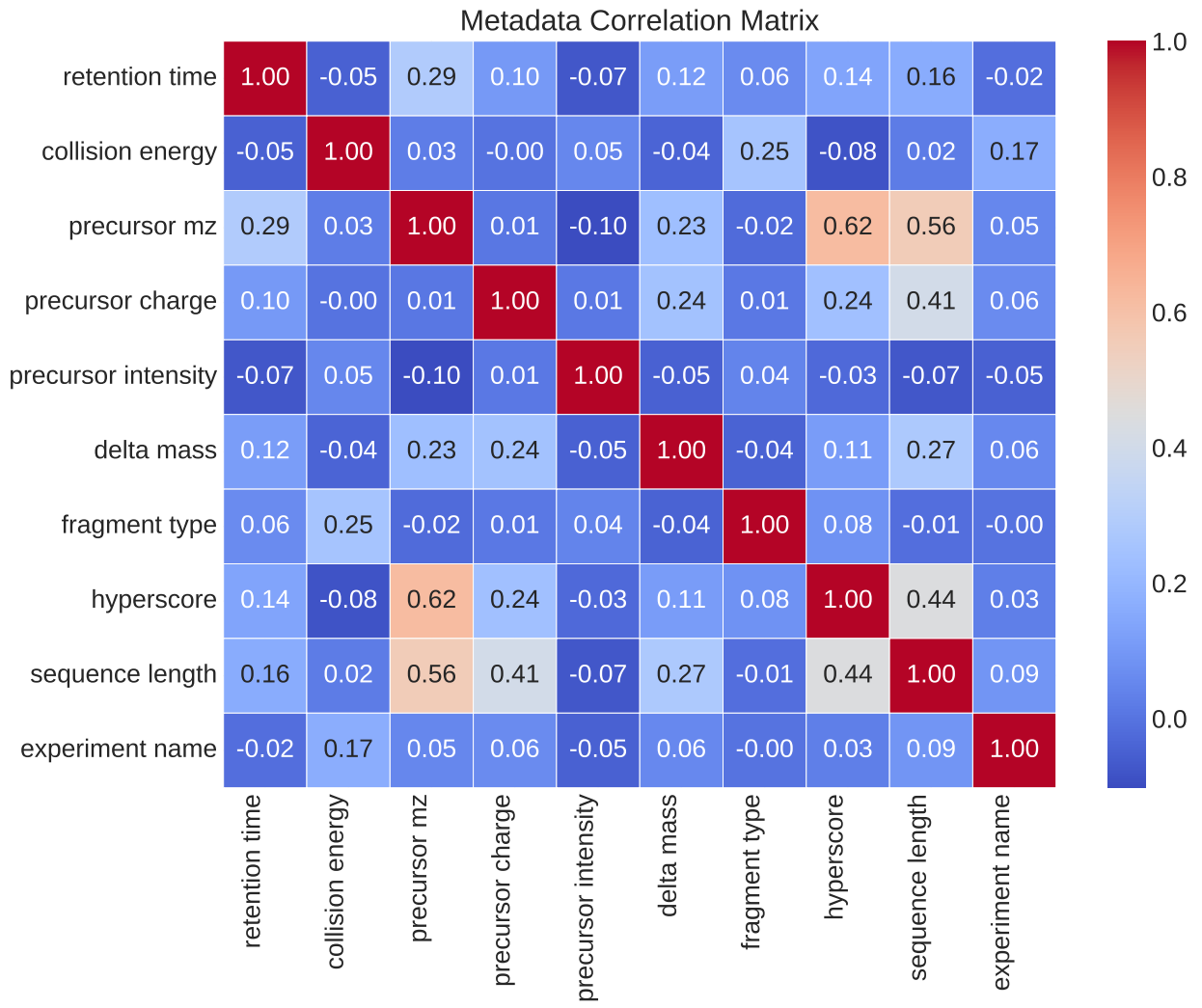


Figure 3.2: Correlation matrix of meta data fields based on 250,000 randomly sampled spectra from the training dataset.

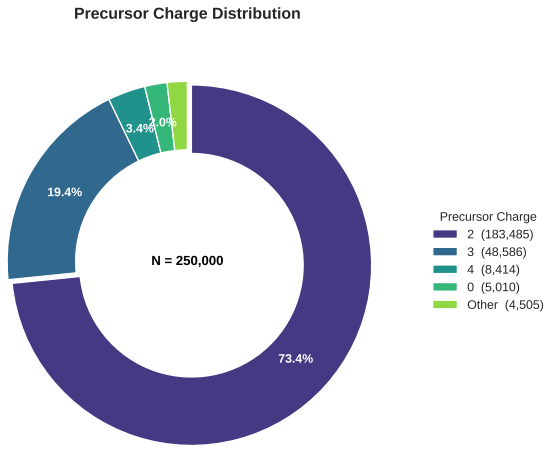
3.1.1 Key features

Each spectrum in the data is a variable-length sequence of pairs of mass-to-charge ratio and intensity, $(m/z_i, I_i)$ ordered in ascending according to mass to charge ratio. The number of peaks varies significantly across the spectra depending on the nature of the precursor, instrument sensitivity and fragmentation efficiency.

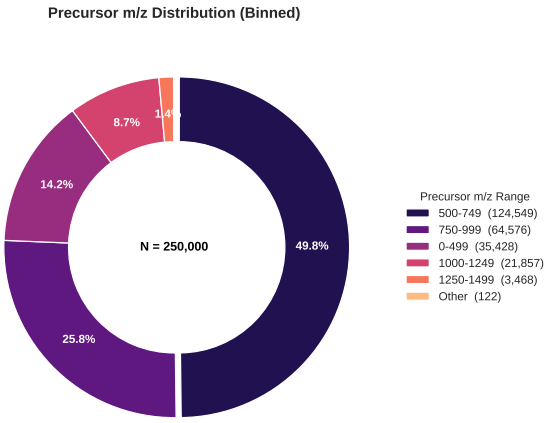
Noise in the MS/MS data is a key feature because it leads to spectra containing both informative peaks and uninformative noise peaks. The noise comes from both instrument and acquisition processes. Because raw spectra can vary in length and contains noise from measurement, preprocessing is done to prepare it for the machine learning task. This preprocessing includes a number of tasks such as peak filtering and normalisation as discussed in section 3.2.

3.1.2 Dataset Statistics

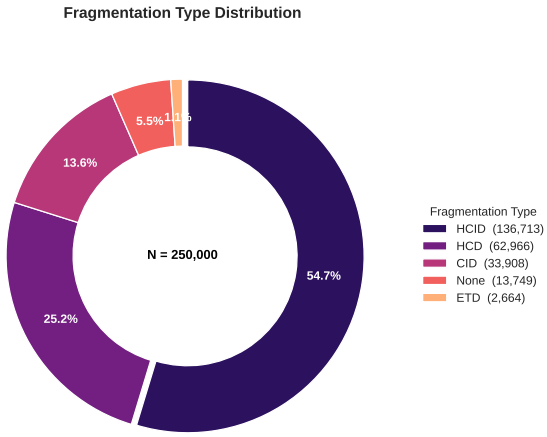
For this study, I utilize two datasets; small experimental dataset for ablation studies, and a large evaluation dataset for comparison with a baseline. For the large-scale evaluation dataset, we consider 18,255,265 spectra samples from 82 PRoteomics IDEntifications (PRIDE) database projects. This dataset contains 1,805,830 unique sequences and 1,493,218 unique unmodified peptides. It was also split into 11,703,040 training samples, 790,417 validation samples and 5,761,808 test samples. The experimental dataset contains 180,000 samples, partitioned as 130,000 samples for training, 10,000 samples for validation, and 40,000 samples for test set. The distribution of samples across different metadata categories also reveals varying levels of class imbalance, which may influence the type of spectral patterns learned during training as shown in Fig. 3.3. As an example, based on the sequence length distribution, the model is training on more spectra with medium length compared to very short and very long sequences which is likely to affect its performance on the under-represented shorter or longer sequences in the data.



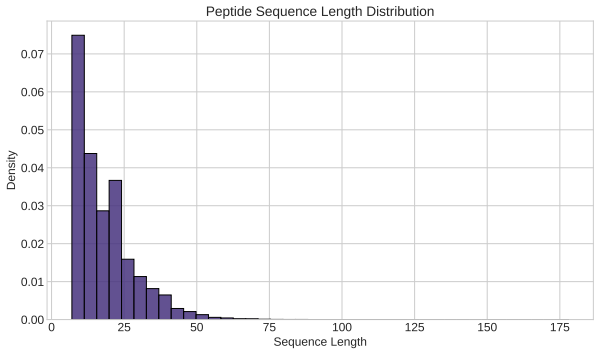
(a) Distribution of training spectra based precursor charge.



(b) Distribution of training spectra based precursor m/z.



(c) Distribution of training spectra based fragmentation type.



(d) Distribution of training spectra based sequence length.

Figure 3.3: Overview of key distributions in the training spectra dataset.

3.2 Data Preprocessing

Because raw MS/MS spectra vary in length, intensity scale, and noise level, preprocessing is required before the data can be used for model training. The preprocessing pipeline consists of spectrum filtering and cleaning, normalisation, fixed-length padding, peak ordering, and masking operations. These steps transform the spectra into consistent input suitable for the diffusion-based learning framework in this study.

3.2.1 Spectrum Filtering and Cleaning

The first preprocessing step involves filtering and cleaning the raw spectra to reduce the impact of noise on the machine learning process as well as remove unnecessary peaks. Peaks outside the selected mass-

to-charge ratio range are discarded because they often contain unreliable or low-quality measurements. Low-intensity peaks are also filtered to suppress instrument noise and background chemical signals that do not contribute useful information to the learning task. In addition, precursor ion peaks are removed from the fragment spectrum to prevent the model from relying on simple reconstruction cues during training.

3.2.2 Normalisation

Normalisation is applied to reduce variations between spectra caused by differences in acquisition conditions, instrument sensitivity, and total ion current. This process ensures that the model focuses on the structural relationships between peaks rather than differences in scale across experiments. Since the input data is two dimensional, two sets of normalisations are implemented for spectra intensity and m/z separately as discussed below:

i) Intensity Normalisation: This normalisation is performed to stabilise the distribution of fragment intensities across spectra. A square-root transformation is first applied to reduce the effect of extreme intensity values and improve numerical stability during training. The transformed intensities are L_2 normalised to ensure consistent scaling across the spectra while preserving relative peak relationships.

ii) m/z Normalisation: The m/z values are normalised using maximum normalisation to a fixed numerical range to ensure consistency across spectra. This ensures that the positional information of peaks is represented on a common scale that can be processed effectively by the transformer-based model architecture.

3.2.3 Peak Ordering

Before padding, peaks are ordered according to ascending order of the mass-to-charge values. This preserves the natural physical structure of the spectrum and ensures that neighbouring peaks maintain their relative positional relationships along the mass axis.

3.2.4 Fixed-Length Padding

Since the number of peaks present in each spectrum varies significantly across samples, each spectrum is padded to a fixed vector length after preprocessing. This ensures that the spectra are processed efficiently in batches. In addition, padding values are added only when necessary, and padding masks are used during training to ensure that padded positions do not contribute to the learning process.

3.2.5 Masking

Since the study involves masked peak prediction for self-supervised learning objective, masking is applied such that during training, a subset (40%) of spectral peaks is hidden from the model, and the diffusion-based encoder is required to reconstruct the masked peaks using the surrounding spectral context.

In particular, a combination of Thompson sampling and span masking is used. This masking strategy is termed *Thompson span masking*. It begins by assigning each spectral peak a sampling score based on its normalized intensity using a Thompson sampling-inspired approach (Russo et al., 2020). We consider $I_i \in [0, 1]$ to denote the normalized intensity of peak i. A sampling score, s_i, is then obtained from a Beta distribution,

$$s_i \sim \text{Beta}(\alpha + \kappa I_i^\gamma, \beta + \kappa(1 - I_i^\gamma)),$$

where α, β, κ, and γ are hyperparameters controlling the shape of the distribution. Peaks with higher sampled scores are selected as anchor peaks, around which contiguous spans of neighbouring peaks are then masked (Joshi et al., 2020). The masking also scheduled according to the diffusion time step during the training process.

3.3 Model Architecture and Experimental Environment

3.3.1 Model Architecture

The diffusion architecture of the model used in this study inspired by Jiang and Cai (2025). The proposed foundation diffusion model consists of a latent encoder transformer with a Diffusion Transformer (DiT) designed to operate on preprocessed spectra represented as token sequences made of up to 200 peak tokens generated from a peak embedder. Based on the design, the final spectral embeddings are extracted as latent tokens from the latent encoder as shown in Fig. 3.4. The system consists of several components including: a peak embedder, a time embedder, a latent encoder transformer and a DiT as discussed below.

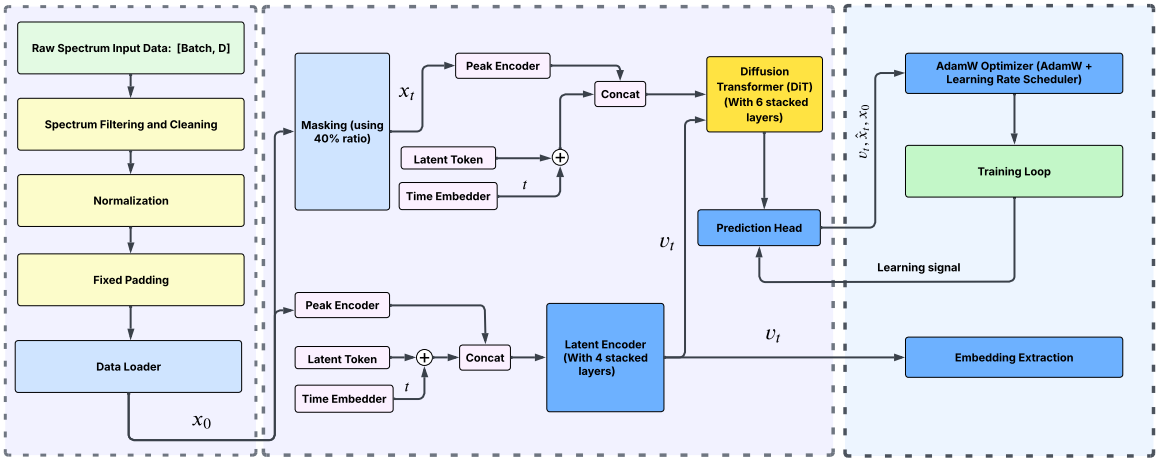


Figure 3.4: Flow Chart of Model Architecture

i) Peak Embedder:
Before the spectral peaks are processed by the main transformer architecture, they are converted into token embeddings using a Multi-scale Peak Embedding module to ensure a compatible and uniform learnable representation of the spectra. To achieve this, the normalised m/z values are first encoded via a set of learnable sinusoidal frequencies, which are initialised to logarithmically-spaced periods spanning the domain of the normalised m/z. This produces a 768-dimensional sinusoidal representation of the peaks,as inspired by Voronov et al. (2022). This representation is further concatenated with the intensity values, and processed by a two-layer Multi-Layer Perceptron (MLP) with a ReLU activation, before projecting the output into a second two-layer MLP designed to yield a 768-dimensional peak-level embedding.

ii) Time Embedding:
The system also utilises a Time Embedder Module that conditions the latent encoder and DiT on the

current stage of the diffusion process. The time-steps are processed as normalised steps ($t \in [0, 1]$) scaled to a maximum period of 10,000 to ensure multi-scale representation of diffusion time, before being projected onto exponentially-spaced frequencies and encoded using corresponding sine and cosine functions, yielding a fixed 768-dimensional sinusoidal representation. This fixed sinusoidal representation is then processed through a two-layer MLP with a SiLU activation function to project into a 768-dimensional time embedding that serves a conditioning signal for the model.

iii) Latent Encoder:
The latent encoder is implemented as a unified transformer encoder consisting of four stacked layers, each configured with a hidden dimension of 768, 12 attention heads, a feed-forward dimension of 1024 and a 0.1 dropout rate. The feed-forward and hidden dimensions are inspired by Eloff et al. (2025). Our encoder is, however, time-dependent, as inspired by Jiang and Cai (2025). However, unlike(Jiang and Cai, 2025), our encoder operates on the clean (unmasked) spectrum, x_0 while incorporating time-step information via a learnable time-step embedding injected into the encoder's attention blocks (specifically to the [LATENT] token modulation) in order to produce a time-dependent global latent representation used for conditioning the diffusion process. We hypothesize that the learned time-step embedding, t helps the encoder produce a more meaningful latent vector v_t, which is passed to the DiT during the diffusion process. This time-step embedding is also used during evaluation to obtain predictions from the encoder.

The encoder also includes positional encoding in form of Rotary Position Embeddings (RoPE) based the m/z-sorted peak sequence with a rotary fraction of 0.25, which is applied to the encoder's attention module. Each encoder layer utilises Flash Multi-Head Attention (FlashMHA) for efficient kernel-based self-attention computation. In addition, each layer follows a transformer block architecture comprising self-attention, residual connections, layer normalisation, and a position-wise feed-forward network as shown in Fig. 3.5.

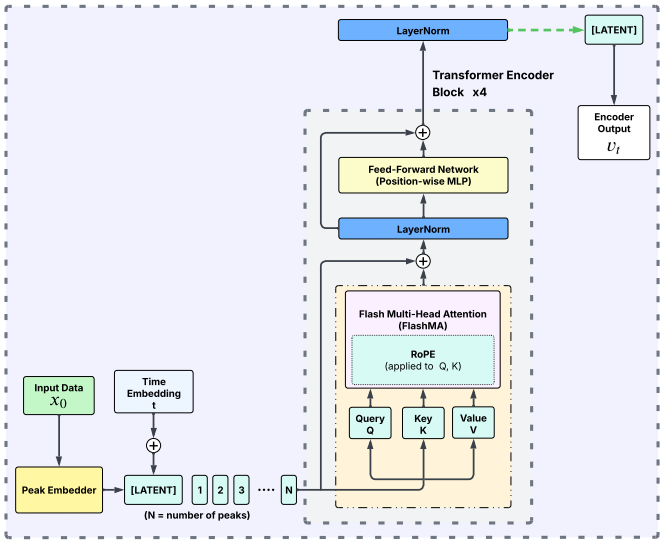


Figure 3.5: Flow Chart of Latent Encoder Architecture

During the forward pass, the clean spectrum, x_0 is first embedded using the peak embedder into a

sequence of tokens, to which a special [LATENT] token is prepended. The temporal conditioning need for conditioning on diffusion time-step, t is injected directly into this latent via a learned time embedding.

Finally, the encoder outputs a sequence, whose first token ([LATENT] token) serves a global summary of the spectrum. This latent token is extracted as the time-dependent conditioning vector, v_t, and projected into the DiT decoder's embedding space through a linear transformation. In addition, this latent token is also retained as the spectral embedding used for downstream evaluation tasks following training. It is also worth noting that the remaining tokens from encoder are discarded during downstream processing to ensure a strict information bottleneck in which the entire spectral structure is compressed into a single learned latent representation, v_t used to guide the reconstruction process by the DiT decoder.

iv) De-noising Transformer:
The de-noising transformer is implemented as a Diffusion Transformer (DiT) consisting of six stacked layers, each using the same hidden dimensionality, number of attention heads, and dropout rate as the latent encoder. Structurally, the DiT is closely aligned with the latent encoder in terms of architectural scale and transformer block design. However, while the encoder operates exclusively on clean spectra x_0, this decoder operates on corrupted (masked) spectral inputs, x_t and performs conditional reconstruction of the original signal.

Similar to the latent encoder, the decoder employs RoPE position embeddings, and FlashMHA to efficiently model interactions between spectral peaks. Each decoder layer consists of a self-attention block, a cross-attention block, and a position-wise feed-forward network as shown in Fig. 3.6, with each sub-layer preceded by layer normalisation and connected through residual pathways. The primary architectural distinction from the encoder is the inclusion of cross-attention and Adaptive Layer Normalisation (AdaLN), both of which enable the decoder to incorporate information from the latent representation, v_t from the encoder.

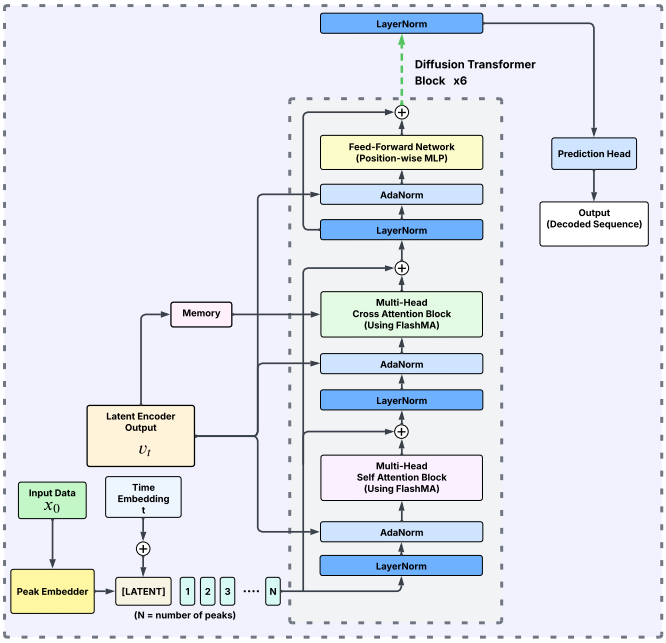


Figure 3.6: Flow Chart of De-noising Architecture

During the forward pass, a corrupted spectrum, x_t is generated through the masking process described in section 3.2.5. The corrupted spectrum is subsequently embedded using a dedicated peak embedder, producing a sequence of token representations. A special decoder `[LATENT]` token is prepended to this sequence, to which the time-step embedding is directly added. As a result, the decoder receives explicit diffusion time-step conditioning through this latent token while preserving the physical representation of the spectral tokens themselves.

The decoder is further conditioned on the latent representation, v_t generated by the encoder. First, v_t is supplied as the memory input to the cross-attention module in every decoder layer. Because the memory consists of a single latent token, each corrupted spectral token attends only to a compact global summary of the clean spectrum. This bottleneck encourages the latent representation to capture high-level spectral structure while preventing direct token-level information transfer between the encoder and the DiT.

Secondly, the same latent representation, v_t is used to modulate internal activations through a variant of Adaptive Layer Normalisation (AdaLN) without gating, as inspired by Perez et al. (2017) and Jiang and Cai (2025). Each decoder layer contains three independent AdaLN modules applied immediately before the self-attention, cross-attention, and feed-forward sub-blocks. Given a normalised activation, x_a, AdaLN applies a transformation of the form

$$\mathrm{AdaLN}(x_a, v_t) = x_a \odot (1 + \gamma(v_t)) + \beta(v_t),$$

where $\odot$ denotes element-wise multiplication, $\gamma(v_t)$ and $\beta(v_t)$ are the scale and shift respectively. The scale and shift parameters are generated directly from the latent conditioning vector, v using a linear projection applied to a SiLU-activated embedding of v_t, producing a $2d$-dimensional vector that is equally split into γ and β where d is the hidden dimension of 768.

Within each decoder layer, the self-attention is also first applied over the corrupted spectral tokens, allowing information exchange between observed and masked positions. The resulting representations are then refined through cross-attention using v_t, followed by a position-wise feed-forward network that transforms each peak token independently. This sequence of operations is repeated across all six decoder layers before a final layer normalisation is applied.

Finally, the decoder's output consists of reconstructed token representations corresponding to each spectral peak. These representations are then passed to a trainable prediction head for masked peak reconstruction, m/z prediction, auxiliary tasks, and spectral representation learning objectives.

iv) Learning Objective:
Unlike previous work by Jiang and Cai (2025) that utilises a variant of the simple objective function, L_{simple} , as highlighted in section 2.3.1, our model is trained using a multi-objective loss function comprising a masked reconstruction loss $\mathcal{L}_{\text{recon}}$, a contrastive loss $\mathcal{L}_{\text{contrast}}$, a variance regularisation term $\mathcal{L}_{\text{var}}$, a full reconstruction loss $\mathcal{L}_{\text{recon\_full}}$, an m/z prediction loss $\mathcal{L}_{\text{mz}}$, and an auxiliary intensity regression loss $\mathcal{L}_{\text{aux}}$. The overall learning objective is expressed as:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{recon}}\mathcal{L}_{\text{recon}} + \lambda_{\text{recon\_full}}\mathcal{L}_{\text{recon\_full}} + \lambda_{\text{contrast}}\mathcal{L}_{\text{contrast}} + \lambda_{\text{mz}}\mathcal{L}_{\text{mz}} + \lambda_{\text{aux}}\mathcal{L}_{\text{aux}} + \lambda_{\text{var}}\mathcal{L}_{\text{var}},$$

where the λ-terms are weighting coefficients that control the relative contribution of each loss component. This objective function is selected based on empirical observations to prevent embedding collapse, which is possible when using L_{simple}, where the diffusion model tends to prioritise reconstruction fidelity without enforcing structure in the learned representation. The components of the multi-objective loss function ensure that the model learns embeddings that are robust and carry meaningful information, as

discussed below. As part of the experiments, we also investigate a variant of the multi-objective loss that includes the centroid regularisation highlighted in section 2.1.

a) Reconstruction Loss

The reconstruction objective follows masked modelling approaches inspired by Masked Autoencoders and denoising diffusion models (Ho et al., 2020; Lai et al., 2025), recovering missing spectral peak embeddings from partial observations through a masked mean squared error objective:

$$\mathcal{L}_{\text{recon}} = \frac{\sum_{b,l} m_{b,l} \, \|\mathbf{y}_{b,l} - \hat{\mathbf{y}}_{b,l}\|_2^2}{D \sum_{b,l} m_{b,l} + \varepsilon}, \tag{3.3.1}$$

where $\mathbf{y}_{b,l} \in \mathbb{R}^D$ denotes the clean DiT peak token at position l in batch element b, $\hat{\mathbf{y}}_{b,l} \in \mathbb{R}^D$ denotes the corresponding decoder prediction, $m_{b,l} \in \{0, 1\}$ is a binary masking indicator, and $\varepsilon = 10^{-8}$ ensures numerical stability when the number of masked positions is small.

b) Full Reconstruction Loss

In addition to masked prediction, a global reconstruction, $\mathcal{L}_{\text{recon\_full}}$ term enforces consistency across all peak-token embeddings:

$$\mathcal{L}_{\text{recon\_full}} = \frac{1}{BLD} \sum_{b=1}^{B} \sum_{l=1}^{L} \|\mathbf{y}_{b,l} - \hat{\mathbf{y}}_{b,l}\|_2^2 \,, \tag{3.3.2}$$

where L is the length of the spectral token sequence. Unlike $\mathcal{L}_{\text{recon}}$, which is restricted to masked positions, this term penalises reconstruction error across the entire sequence, encouraging consistency in both masked and unmasked regions and stabilising training.

c) Contrastive Loss

The contrastive objective follows the InfoNCE formulation used in self-supervised representation learning (Chen et al., 2020; van den Oord et al., 2018). Given a batch of B spectra, each spectrum latent embedding,$\mathbf{v}_t^i$ is paired with an augmented view, $\mathbf{v}_{\text{aug}}^i$ producing embedding pairs $(\mathbf{v}_t^i, \mathbf{v}_{\text{aug}}^i)$ for $i = 1, \ldots, B$. The pairwise similarity logits are computed as:

$$S_{ij} = \frac{\bar{\mathbf{v}}_t^i \cdot \bar{\mathbf{v}}_{\text{aug}}^j}{\tau} \tag{3.3.3}$$

where $\bar{\mathbf{v}}$ is L2 normalized latent embedding and τ is a temperature hyperparameter controlling the sharpness of the distribution. The contrastive loss is then obtained as the cross-entropy loss

$$\mathcal{L}_{\text{contrast}} = -\frac{1}{B} \sum_{i=1}^{B} \log \frac{\exp(S_{ii})}{\sum_{j=1}^{B} \exp(S_{ij})}, \tag{3.3.4}$$

where the diagonal entries S_{ii} correspond to positive pairs and off-diagonal entries $S_{ij,\ i \neq j}$ are treated as negatives. This encourages the encoder to produce augmentation-invariant representations by pulling together embeddings of the same spectrum while pushing apart embeddings of different spectra.

d) m/z Prediction Loss

The m/z prediction objective supervises the model to predict the m/z value of each masked spectral peak. Predictions are made using a two-stage hierarchical classification head, where the m/z range $[m/z_{\min},\ m/z_{\max}]$ is partitioned into G groups of K bins each, giving a bin resolution of δ Da. For

each masked position i, the head produces group logits $\mathbf{g}_i \in \mathbb{R}^G$ and offset logits $\mathbf{o}_i \in \mathbb{R}^K$. The loss objective is then:

$$\mathcal{L}_{mz} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \Big[w_g \cdot \mathrm{CE}(\mathbf{g}_i,\ g_i^*) + w_o \cdot \mathrm{CE}(\mathbf{o}_i,\ o_i^*) \Big], \tag{3.3.5}$$

where $\mathrm{CE}(\mathbf{p}, y) = -\log \mathrm{softmax}(\mathbf{p})_y$ denotes the cross-entropy loss between logits $\mathbf{p}$ and ground-truth class y, $\mathcal{M}$ denotes the set of masked positions, g_i^* and o_i^* are the ground-truth group and offset bin indices for the true m/z value at position i, and weighting coefficients are w_g and w_o. This loss directly supervises fragment ion m/z recovery at masked positions, forming a strong indicative bias for the model.

e) Auxiliary Intensity Loss

An auxiliary supervision term is introduced for peak-level intensity prediction using the Huber loss, as inspired by Kharyuk et al. (2018):

$$\mathcal{L}_{\text{intensity}} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \mathrm{Huber}(\hat{I}_i, I_i^*; \delta), \tag{3.3.6}$$

where $\hat{I}_i$ and I_i^* denote the predicted and ground-truth intensities at masked position i, respectively, $\mathcal{M}$ is the set of masked positions, and $\delta = 0.1$ is the transition parameter that determines the boundary between the quadratic and linear regions of the loss function. The Huber loss is defined as

$$\mathrm{Huber}(\hat{I}, I^*; \delta) = \begin{cases} \dfrac{(\hat{I} - I^*)^2}{2\delta}, & \text{if } |\hat{I} - I^*| < \delta, \\ |\hat{I} - I^*| - \dfrac{\delta}{2}, & \text{otherwise.} \end{cases} \tag{3.3.7}$$

This loss is computed only over masked positions and improves robustness to outlier intensity values arising from noise and sparsity in MS/MS spectra.

f) Variance Regularisation

To prevent embedding collapse, a VICReg-style regularisation is applied to the un-normalised Latent token embeddings, $\mathbf{z} \in \mathbb{R}^{B \times D}$ (Bardes et al., 2022). It combines a variance term and a covariance term:

$$\mathcal{L}_{\text{var}} = \frac{\lambda_v}{D} \sum_{j=1}^{D} \max(0,\ \gamma - \sigma_j) + \frac{\lambda_c}{D} \sum_{i \neq j} [\mathbf{C}]_{ij}^2, \tag{3.3.8}$$

where $\sigma_j = \sqrt{\mathrm{Var}_b(z_{b,j}) + \varepsilon}$ is the standard deviation of the j-th embedding dimension across the batch, γ is the target minimum standard deviation, $\mathbf{C} \in \mathbb{R}^{D \times D}$ is the batch covariance matrix with diagonal zeroed out, and weighting coefficients being λ_v and λ_c. The variance term prevents dimensional collapse by penalising dimensions whose variance falls below γ, while the covariance term decorrelates embedding dimensions to maximise the information content of the representation.

g) Centroid Loss

The centroid regularisation term, $\mathcal{L}_{\text{cent}}$ defined below provides physics-informed supervision over the spectral distribution using the spectral centroid defined in Section 2.1.

$$\mathcal{L}_{\text{cent}} = \frac{1}{B} \sum_{b=1}^{B} \Big(\hat{m}_{c,b} - m_{c,b} \Big)^2, \tag{3.3.9}$$

where $m_{c,b}$ and $\hat{m}_{c,b}$ denote the spectral centroid of the ground-truth and predicted peak embeddings for batch element b, respectively. This term penalises shifts in the spectral centre of mass between predicted and true embeddings, encouraging the model to preserve the global mass distribution of the latent representation.

v) Optimiser:
During training, the model is optimised using the AdamW optimiser with a learning rate of 5×10^{-4}, and a weight decay of 0.1. To ensure stable convergence, a cosine warmup-hold learning rate schedule is employed. The learning rate is linearly increased over the first 10% of training steps, followed by a holding phase at the maximum learning rate for the subsequent 5% steps, before being smoothly decayed following a cosine schedule to a minimum value equal to 10% of the base learning rate.

The training is performed with an effective batch size of 256, combined with gradient accumulation over 4 steps to improve gradient stability and ensure robust optimisation. Hyperparameter tuning was performed using manual search. Although manual search can be less efficient than automated optimization methods, as highlighted by Franceschi et al. (2025), it was considered appropriate in this study due to limited computational resources.

3.3.2 Experimental Environment

The foundational diffusion model was implemented using PyTorch and the Hugging Face Accelerate Python Libraries. Training and evaluation were executed on the AIchor machine learning platform using a containerised Docker environment.

Training was performed on a single compute node with four H100 NVIDIA GPUs, each providing 80 GB of High Bandwidth Memory, and 96 CPU cores. Accelerate was used for distributed multi-GPU training. TensorBoard and MLflow were used to track training progress, model performance, dataset metadata, and system resource utilisation throughout training. However, for the sanity check and ablation studies, a smaller compute setup, consisting of one H100 Colab runtime with 80GB of memory, was used due to limited access to the bigger compute resources.

The foundation model was trained for $30,000$ optimisation steps using an effective batch size of 256. Checkpoints were saved every $10,000$ steps, and the best-performing checkpoint, determined by the validation loss, was retained for downstream evaluation. For sanity check and ablation studies, $14,400$ optimisation steps with batch size of 128 were used.

3.4 Model Evaluation

This section describes the baseline model and the evaluation metrics used for assessment of the trained diffusion model.

3.4.1 Baseline

The baseline model used in this study is an existing state-of-the-art foundation transformer encoder developed and provided by the InstaDeep team (InstaDeep, 2026). This model is an encoder-only transformer consisting of nine transformer layers, twelve attention heads, a hidden dimension of 768, and a feed-forward dimension of 1024. It also uses a preprocessing pipeline similar to that described in Section 3.2, including normalisation, peak ordering, fixed-length padding, and masking. Comparisons between the baseline and our diffusion-based encoder model were performed using the same embedding

evaluation pipeline, allowing the impact of diffusion-based representation learning on embedding quality, latent space structure, and downstream predictive performance to be assessed.

3.4.2 Model Evaluation

The model was evaluated using an embedding-based assessment pipeline. Evaluation was performed on the validation split of the dataset using the best-performing model checkpoint. Embeddings were generated using the same preprocessing pipeline employed during model training. The evaluation consisted of three tasks; embedding statistics, linear probe classification, and UMAP visualisation. A maximum of 140,000 spectra were used for evaluation with a fixed random seed of 42 to ensure reproducibility.

i) Embedding Statistical Metrics
Embedding statistical metrics were used to characterise the structure and distribution of the learned latent space. Metrics included embedding norm statistics, variance utilisation, principal component analysis (PCA) explained variance, silhouette score, and pairwise cosine similarity distributions. These metrics were used to assess embedding diversity, clustering behaviour,and potential representation collapse as discussed below.

- **Embedding Norm Statistics.** Embedding norm statistics were computed by measuring the L2 norm of each embedding vector and summarising the resulting distribution using the mean, standard deviation, and coefficient of variation. These metrics were used to assess the consistency of embedding magnitudes and verify that the latent representations remained appropriately normalised.
- **Collapse Indicator:** A collapse indicator was computed as the norm of the mean embedding vector across all samples (Chen et al., 2026). This metric was used to detect representation collapse, where embeddings become concentrated around a common direction and lose diversity.
- **Per-Dimension Variance:** Per-dimension standard deviation statistics were computed by measuring the standard deviation of each latent dimension across the evaluation set (Chen and He, 2020; Bardes et al., 2022). The mean and minimum standard deviation values were used to assess the utilisation of latent dimensions and identify potentially inactive features.
- **PCA Explained Variance:** Principal Component Analysis (PCA) was applied to the embedding matrix to analyse the distribution of variance within the latent space (Leske et al., 2026). Cumulative explained variance ratios and the participation ratio were used to estimate the effective dimensionality of the learned representation and determine whether information was concentrated within a limited number of components.
- **Pairwise Cosine Similarity:** Pairwise cosine similarities were computed between randomly sampled embedding pairs (Leske et al., 2026). The mean and standard deviation of the similarity distribution were used to evaluate embedding diversity and identify potential representation collapse resulting from excessively similar embeddings.
- **Clustering Structure:** K-means clustering was applied to the embedding representations to investigate latent space structure. Cluster quality was evaluated using the silhouette score, as introduced by (Rousseeuw, 1987). These analyses were motivated by prior representation-learning studies that examine latent space structure and clustering behaviour (Bardes et al., 2022; Leske et al., 2026).
- **Overall Quality Score:** A composite embedding quality score was computed from a set of threshold-based validation checks derived from the embedding statistics. The score provided a

high-level summary of latent space quality and was used to identify potential representation issues requiring further investigation.

ii) Linear Probes

Linear probe experiments were conducted by training simple linear classifiers and linear regressors on the learned embeddings, as inspired by Jiang and Cai (2025), Bushuiev et al. (2025) and Bardes et al. (2022). For the classifiers, the performance was measured using accuracy, while the regression tasks were measured using the coefficient of determination, R^2. These experiments were used to evaluate the extent to which biologically relevant information was encoded within the latent representations, v_t.

iii) Embedding Visualisation

Uniform Manifold Approximation and Projection (UMAP) visualisation was used to project the high-dimensional embeddings into a two-dimensional space for qualitative inspection of cluster structure and group separation, as inspired by Bushuiev et al. (2025) and McInnes et al. (2020). Together, these evaluation tasks provided both quantitative and qualitative measures of representation quality, latent space organisation, and downstream predictive utility.

3.5 Experimental Design

3.5.1 Sanity Check

In this study, sanity check experiments, as listed in Table 3.1, were conducted using the MNIST dataset to verify the correctness of the diffusion model implementation and the representation extraction pipeline (LeCun et al., 2010). For this check, the MNIST data undergoes an additional step before the main preprocessing pipeline to convert the 28 $\times$ 28 images into spectrum representations. This is achieved by using the value of each pixel as its intensity and the corresponding pixel position index as the associated m/z value, as shown in Fig. 3.7. The resulting 784-dimensional spectrum is then compressed to 300 dimensions using linear interpolation before being used to train the model and generate embeddings for the MNIST samples. Successful separation of digit classes in the embedding space is used as an indicator that the learned representations capture meaningful semantic information. These results, together with UMAP visualisations of the MNIST embeddings, are also compared with previous studies that use diffusion models for representation learning on the MNIST dataset.

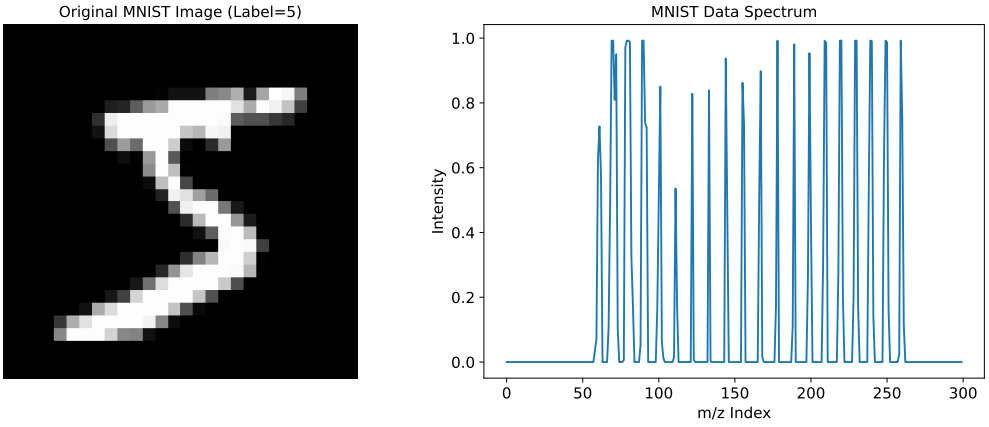


Figure 3.7: Visualization of the preprocessed MNIST data. Left: original MNIST image (28×28). Right: corresponding 300-dimensional compressed spectrum obtained via linear interpolation for dimensionality reduction.

Table 3.1: Sanity check experiment configurations used in the experiments.

Experiment	Description
Sanity-CL	Contrastive-loss-only sanity check without AdaLN
Sanity-DF	Diffusion (DF) objective-only sanity check without AdaLN
Sanity-AUX	Full model without auxiliary loss (AUX) without AdaLN
Sanity-AdaLN	Full multi-objective with AdaLN
Sanity-base	Full multi-objective loss setup without AdaLN

3.5.2 Foundational Diffusion Model Experiment

The foundational diffusion model experiment forms the main experimental setting of this study. The model is trained using the full multi-objective loss, and evaluated on the large data set described in section 3.1.2. The dataset is split into training, validation, and test sets to ensure proper evaluation and avoid overfitting.

Training is performed using hyperparameters summarized in Table 3.2. The same preprocessing pipeline described in section 3.2 is applied to all data splits. Once this foundational model is trained, the results obtained from it are compared with the baseline using the evaluation metrics highlighted in section 3.4.

Table 3.2: Training hyperparameters for the foundational diffusion model

Parameter	Value
Learning Rate	5×10^{-4}
Batch Size	256
Gradient Accumulation	4
Weight Decay	0.1
Optimizer	Adam
Gradient Clipping	1.0
Number of Training Steps	30,000
Validation Interval	10,000 steps
Warmup Fraction	0.1
Learning Rate Scheduler	Cosine warmup-hold
Minimum LR Factor	0.1
Seed	101
Diffusion Evaluation Time-Step (t)	100

Table 3.3: Training hyperparameters for the sanity check and ablation experiments.

Parameter	Value
Learning Rate	5×10^{-4}
Batch Size	128
Gradient Accumulation	4
Weight Decay	0.1
Optimizer	Adam
Gradient Clipping	1.0
Number of Training Steps	15,234
Validation Interval	1,000 steps
Warmup Fraction	0.1
Learning Rate Scheduler	Cosine warmup-hold
Minimum LR Factor	0.1
Seed	101
Diffusion Evaluation Time-Step (t)	100

3.5.3 Ablation Experiments

For this study, ablation studies, focusing on the components of the multi-objective loss function and AdaLN, were done to evaluate the contribution of these components to the proposed diffusion framework, as described in Table 3.4. All ablation experiments follow the hyperparameter configuration in Table 3.3.

Table 3.4: Ablation experiment configurations used in the experiments.

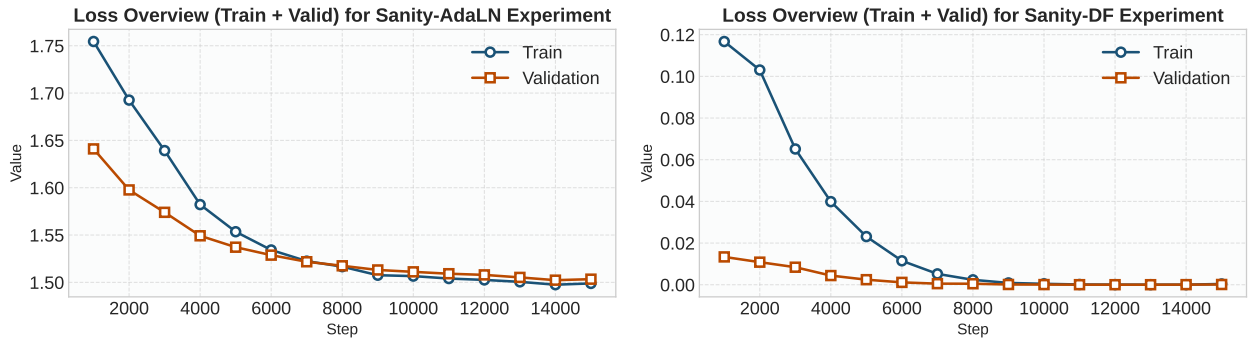
Ablation	**Description**
Ablation-1:DF	Model using diffusion masked reconstruction loss (DF) only objective without AdaLN ($\lambda_{\text{recon}} = 0.5, \lambda_{\text{recon\_full}} = 0, \lambda_{\text{contrast}} = 0, \lambda_{\text{mz}} = 0, \lambda_{\text{aux}} = 0, \lambda_{\text{var}} = 0$)
Ablation-2:+CL	Contrastive loss added to Ablation-1: DF ($\lambda_{\text{recon}} = 0.5, \lambda_{\text{recon\_full}} = 0, \lambda_{\text{contrast}} = 2.0, \lambda_{\text{mz}} = 0, \lambda_{\text{aux}} = 0, \lambda_{\text{var}} = 0$)
Ablation-3:+RL	Reconstruction loss added to Ablation-1: DF ($\lambda_{\text{recon}} = 0.5, \lambda_{\text{recon\_full}} = 0.1, \lambda_{\text{contrast}} = 0, \lambda_{\text{mz}} = 0, \lambda_{\text{aux}} = 0, \lambda_{\text{var}} = 0$)
Ablation-4:+CL+RL	Joint contrastive and full reconstruction losses added to Ablation-1: DF ($\lambda_{\text{recon}} = 0.5, \lambda_{\text{recon\_full}} = 0.1, \lambda_{\text{contrast}} = 2.0, \lambda_{\text{mz}} = 0, \lambda_{\text{aux}} = 0, \lambda_{\text{var}} = 0$)
Ablation-5: CL-only	Contrastive loss objective-only model without AdaLN ($\lambda_{\text{recon}} = 0, \lambda_{\text{recon\_full}} = 0, \lambda_{\text{contrast}} = 1.0, \lambda_{\text{mz}} = 0, \lambda_{\text{aux}} = 0, \lambda_{\text{var}} = 0$)
Ablation-6: AUX-only	Auxiliary-loss-only model without AdaLN ($\lambda_{\text{recon}} = 0, \lambda_{\text{recon\_full}} = 0, \lambda_{\text{contrast}} = 0, \lambda_{\text{mz}} = 0.5, \lambda_{\text{aux}} = 0.2, \lambda_{\text{var}} = 0$)
Ablation-7: +VAR	Variance loss added to Ablation-4:+CL+RL ($\lambda_{\text{recon}} = 0.5, \lambda_{\text{recon\_full}} = 0.1, \lambda_{\text{contrast}} = 2.0, \lambda_{\text{mz}} = 0, \lambda_{\text{aux}} = 0, \lambda_{\text{var}} = 0.5$)
Ablation-8:+VAR +AUX	Variance and auxiliary losses added to Ablation-4:+CL+RL ($\lambda_{\text{recon}} = 0.5, \lambda_{\text{recon\_full}} = 0.1, \lambda_{\text{contrast}} = 2.0, \lambda_{\text{mz}} = 0.5, \lambda_{\text{aux}} = 0.1, \lambda_{\text{var}} = 0.5$)
Ablation-9: Full	Full multi-objective model (with no centroid loss and AdaLN) ($\lambda_{\text{recon}} = 0.5, \lambda_{\text{recon\_full}} = 0.1, \lambda_{\text{contrast}} = 2.0, \lambda_{\text{mz}} = 0.5, \lambda_{\text{aux}} = 0.1, \lambda_{\text{var}} = 0.5, \lambda_{\text{cent}} = 0$)
Ablation-10:Full+CENT	Centroid (CENT) loss added to Ablation-9:Full ($\lambda_{\text{recon}} = 0.5, \lambda_{\text{recon\_full}} = 0.1, \lambda_{\text{contrast}} = 2.0, \lambda_{\text{mz}} = 0.5, \lambda_{\text{aux}} = 0.1, \lambda_{\text{var}} = 0.5, \lambda_{\text{cent}} = 0.5$)
Ablation-11: Full+CENT+AdaLN	Full model with centroid loss and AdaLN

4. Results and Discussion

This chapter presents and discusses the results from the evaluation of the proposed foundation diffusion model. First, we present sanity check experiments on the MNIST dataset to validate the correct implementation of the model and training pipeline. We then present ablation studies on mass spectrometry data to assess the contribution of different components and loss functions used in the model. Finally, we provide a comparison of the proposed model against the baseline to qualify its overall performance. In the experiments, we evaluate the latent representations of mass spectra using UMAP visualisations, linear probe performance, pairwise similarity distributions, and embedding statistics to understand both the structure and quality of the learned embedding space. While this section discusses the key findings, additional results for sanity and ablation experiments are shown in the Appendix A and B.

4.1 Sanity Check

Based on the training loss curves shown in Figure 4.1, both sanity check experiments successfully validate effective learning of the underlying computational framework. Using Sanity-AdaLN experiment (Figure 4.1a) and Sanity-DF experiment (Figure 4.1b), we show that the our diffusion model implementation effectively learns both the simple single loss objective in Figure 4.1b, as well as the harder multi-loss objective in Figure 4.1a. Specifically, in both experiments, the models achieve stable convergence, providing initial evidence that the optimization pipeline, loss formulations, and underlying computational graph are implemented correctly.



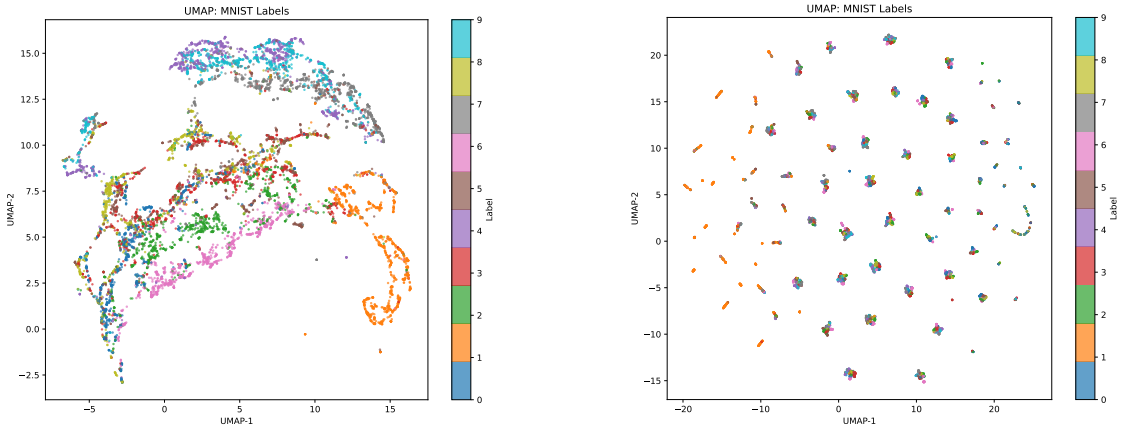
(a) Sanity-AdaLN

(b) Sanity-DF

Figure 4.1: Training and Validation Loss for Sanity check experiments

To assess the model's ability to learn meaningful latent representation, the two experiments; Sanity AdaLN and Sanity-DF were conducted using the MNIST data before applying the architecture to downstream proteomics mass spectra. In Figure 4.2a, the Sanity-AdaLN experiment produces well-defined clusters, with samples belonging to the same digit class generally grouped together and separated from other classes. In particular, clusters corresponding to visually similar digits such as 4, 7, and 9 are positioned in close proximity, consistent with observations in previous studies on MNIST embeddings by Jiang and Cai (2025) and Hosseini-Asl et al. (2016), indicating that the learned embeddings encode semantic information. In contrast, the Sanity-DF experiment shown in Figure 4.2b exhibits a more fragmented embedding structure, with samples distributed across numerous small clusters with limited visual separation between digit classes. This indicates that, although the model is able to optimize the

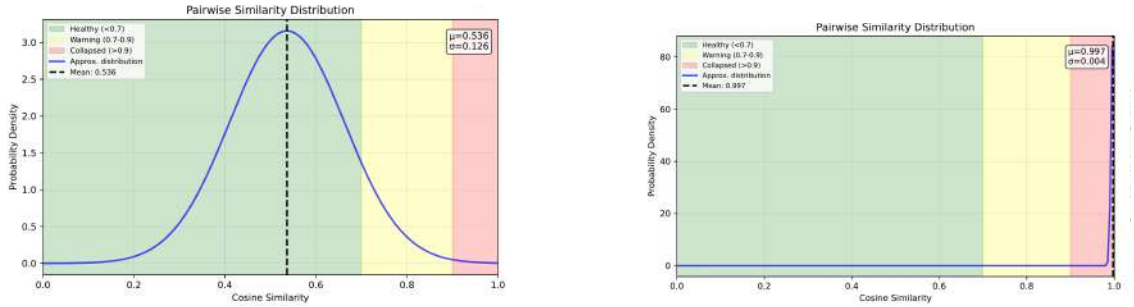
training objective, the resulting latent representations capture less class-discriminative information compared to Sanity-AdaLN. Using these results, we show that the ability of the model can learn to produce well-defined clusters in the embedding space with semantic information, but this ability depends on the model architecture and training objectives.



(a) Sanity-AdaLN (b) Sanity-DF

Figure 4.2: UMAP Visualisation of Embeddings from MNIST

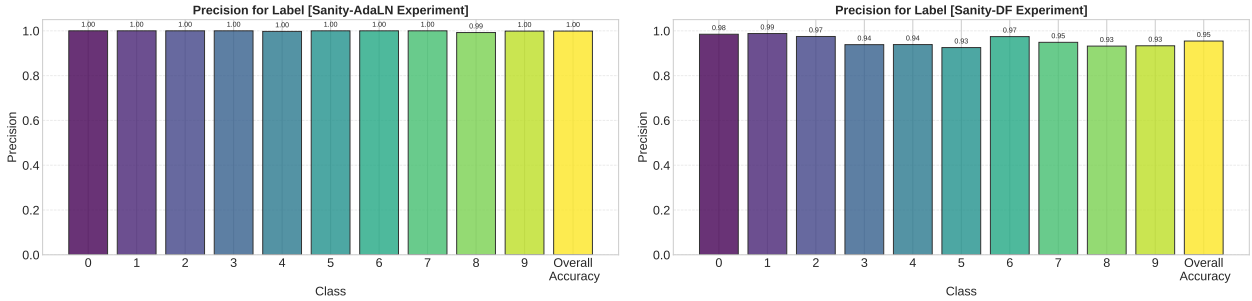
The results in Figures 4.3 and 4.4 together with Table 4.1 were used to further validate our embedding evaluation pipeline. In Figure 4.3, we show that the pairwise similarity distribution of the embeddings is consistent the observations from the UMAP visualizations. In particular, the better performing sanity configuration, Sanity-AdaLN in Figure 4.3a exhibits a well-distributed similarity profile that is approximately Gaussian, centred at a mean cosine similarity of $\mu = 0.536$ with a standard deviation of $\sigma = 0.126$, and most pairwise comparisons lie within the designated "Healthy" range (< 0.7), suggesting that the latent space preserves sufficient geometric variability and maintains meaningful separation between representations. On the other hand, the Sanity-DF configuration (Figure 4.3b) shows a strongly concentrated similarity distribution, with values tightly clustered near the upper bound. The mean cosine similarity increases to $\mu = 0.997$ with a very low variance ($\sigma = 0.004$), indicating that most embeddings become nearly indistinguishable in direction. This behaviour is consistent with a collapsed representation space, where unique inputs are represented by similar embeddings (Yao et al., 2026). The pairwise similarity distribution, therefore, serves as an important diagnostic for evaluating potential representation collapse.



(a) Sanity-AdaLN (b) Sanity-DF

Figure 4.3: Pairwise Similarity Distribution of Embeddings from MNIST

In Figure 4.4, we show that linear probes were able to extract the semantic information encoded in the spectral embeddings. Specifically, where the model encodes stronger semantic information, the probes achieve 100% accuracy and class precision (Figure 4.4a), compared to the Sanity-DF results (Figure 4.4b), which show lower probe accuracy (94%) and class precision. These results confirm our choice of using linear probes on spectral embeddings to assess semantic information.



(a) Sanity-AdaLN (b) Sanity-DF

Figure 4.4: Linear Probe Results based Embeddings from MNIST

The results in Table 4.1 also validate our choice of embedding statistics used in the evaluation pipeline. The values for the collapse indicator, Top-1 PCA variance ratio, participation ratio, effective number of components at 90% variance, and quality score are consistent with the observations from UMAP visualization, pairwise similarity distribution, and linear probes. In particular, these metrics consistently distinguish between collapsed and well-structured representations, where poor performing experiments such as Sanity-DF and Sanity-AUX exhibit near-maximal collapse indicators (≈ 0.998) and extreme concentration in the first principal component (PCA Top-1 ≈ 0.999), with an effective dimensionality of 1, indicating strong directional anisotropy and variance concentration along a single dominant direction. It is important to note that this notion of collapse captures loss of isotropy in the representation space rather than complete removal of task-relevant signals, such as small residual orthogonal components that may remain linearly separable, which explains why Sanity-DF can still achieve a 95% accuracy on the MNIST despite exhibiting strong collapse under these statistics. In contrast, better performing experiments such as Sanity-AdaLN, Sanity-CL, and Sanity-Base maintain substantially lower collapse indicators, higher effective dimensionality (20 to 44), and more distributed variance across components, consistent with their higher quality scores and better separation observed in other evaluation results.

Regarding silhouette score, we note that in this setting higher silhouette values do not necessarily correlate positively with representation quality. For example, Sanity-DF exhibits the highest silhouette score (0.3401) despite being a collapsed representation, indicating that silhouette is sensitive to clustering artefacts induced by anisotropy rather than meaningful semantic structure.

Overall, the sanity check results confirm that the model implementation and evaluation framework for representation learning on spectral data is correct and suitable use for subsequent experiments using mass spectrometry data.

Table 4.1: Embedding Statistics summary across sanity experiments.

Sanity-Base	Mean Norm	Collapse Ind.	PCA Top-1	Part. Ratio	Eff. Dim.	Silhouette	Mean Sim.	Qual. Score
Sanity-AdaLN	1.0000	0.7315	0.2090	13.5	44	0.1327	0.5362	0.90
Sanity-CL	1.0000	0.6750	0.1299	13.3	20	0.2128	0.4556	0.90
Sanity-DF	1.0000	0.9985	0.9922	1.0	1	0.3401	0.9971	0.10
Sanity-AUX	1.0000	0.9984	0.9930	1.0	1	0.3567	0.9969	0.10
Sanity-base	1.0000	0.7429	0.1835	15.6	43	0.1619	0.5525	0.90

Mean Norm	Mean ℓ_2 norm of embeddings	Part. Ratio	Participation Ratio
Collapse Ind.	Collapse Indicator	Eff. Dim.	No. components at 90% variance (effective dimensionality)
PCA Top-1	Top-1 PCA variance ratio	Mean Sim.	Mean pairwise cosine similarity
		Qual. Score	Composite Quality Score

4.2 Ablation Experiments

In this section, we show the performance of our foundation diffusion model on mass spectra using different ablation configurations in order to evaluate the contribution of different architectural components and training objectives to the quality of the learned latent representations.

4.2.1 UMAP Visualisation

Figures 4.5 and 4.6 present the resulting UMAP visualizations from different ablations, assessed based on how effectively the latent space organizes samples according to six physical properties: (a) precursor m/z, (b) precursor charge, (c) collision energy (V), (d) retention time (s), (e) number of peaks, and (f) peptide sequence length.

The UMAPs reveal that the model learns to organize samples within the latent space according to the chosen loss objective. For example, the simpler reconstruction objectives (Ablation-1, Ablation-3:+RL, and Ablation-6) show relatively compact and weakly structured embedding spaces, which is consistent with the observations from the sanity check in Section 4.1.

On the other hand, the harder multi-loss objectives (Ablation-2:+CL, Ablation-4:+CL+RL, Ablation-7:+VAR, Ablation-8:+AUX+VAR, Ablation-9: Full, Ablation-10:Full+CENT, and Ablation-11: Full+CENT+VAR) produce improved global organization of the embedding space. The results show that the latent space evolves into a more continuous manifold with clearer separation according to precursor charge and precursor mass states, as well as sequence length and number of peaks. It is also worth noting that although the harder objectives perform well, the structure and quality of their latent space is driven mainly by the contrastive loss. According to Ablation-5: CL-only (Figure 4.5), using the contrastive loss alone generates a similar latent space to those learned by these objectives.

Overall, our results show that the structure of the latent space learned by the different objectives is shaped by the contribution of the competing loss functions. The resulting latent space also exhibits patterns consistent with those observed in previous studies on mass spectra by Bushuiev et al. (2025).

4.2.2 Linear Probe Results

Figures 4.7 and 4.8, present the classification accuracies and regression R^2 scores respectively, of the linear probes across ablations with physical and experimental metadata targets. Specifically, for the classification task with the harder objectives, the linear probe achieves relatively strong performance for precursor charge prediction, with the highest accuracy observed for Ablation-9: Full at 0.910, followed closely by Ablation-11:Full+CENT+AdaLN at 0.906, while simpler objectives (Ablation-1:DF at 0.874,

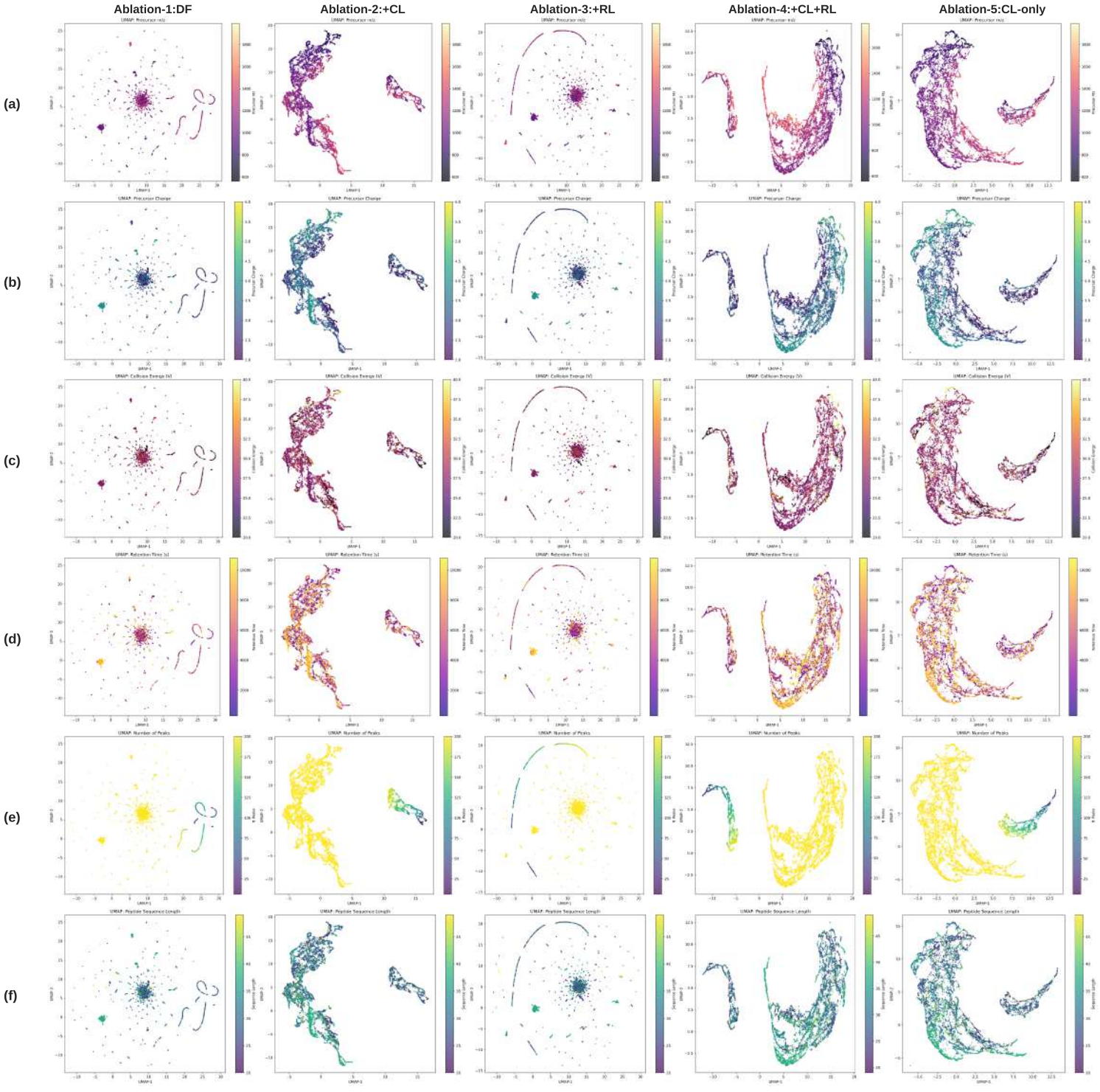


Figure 4.5: **UMAP visualization of latent embeddings across model ablations ; Ablation-1:DF, Ablation-2:+CL, Ablation-3:+RL, Ablation-4:+CL+RL and Ablation-5:CL-only.** Columns show observations from different ablation experiments, while rows display the same embedding projection coloured by metadata: (a) precursor mass, (b) precursor charge, (c) collision energy, (d) retention time, (e) number of peaks, and (f) sequence length.

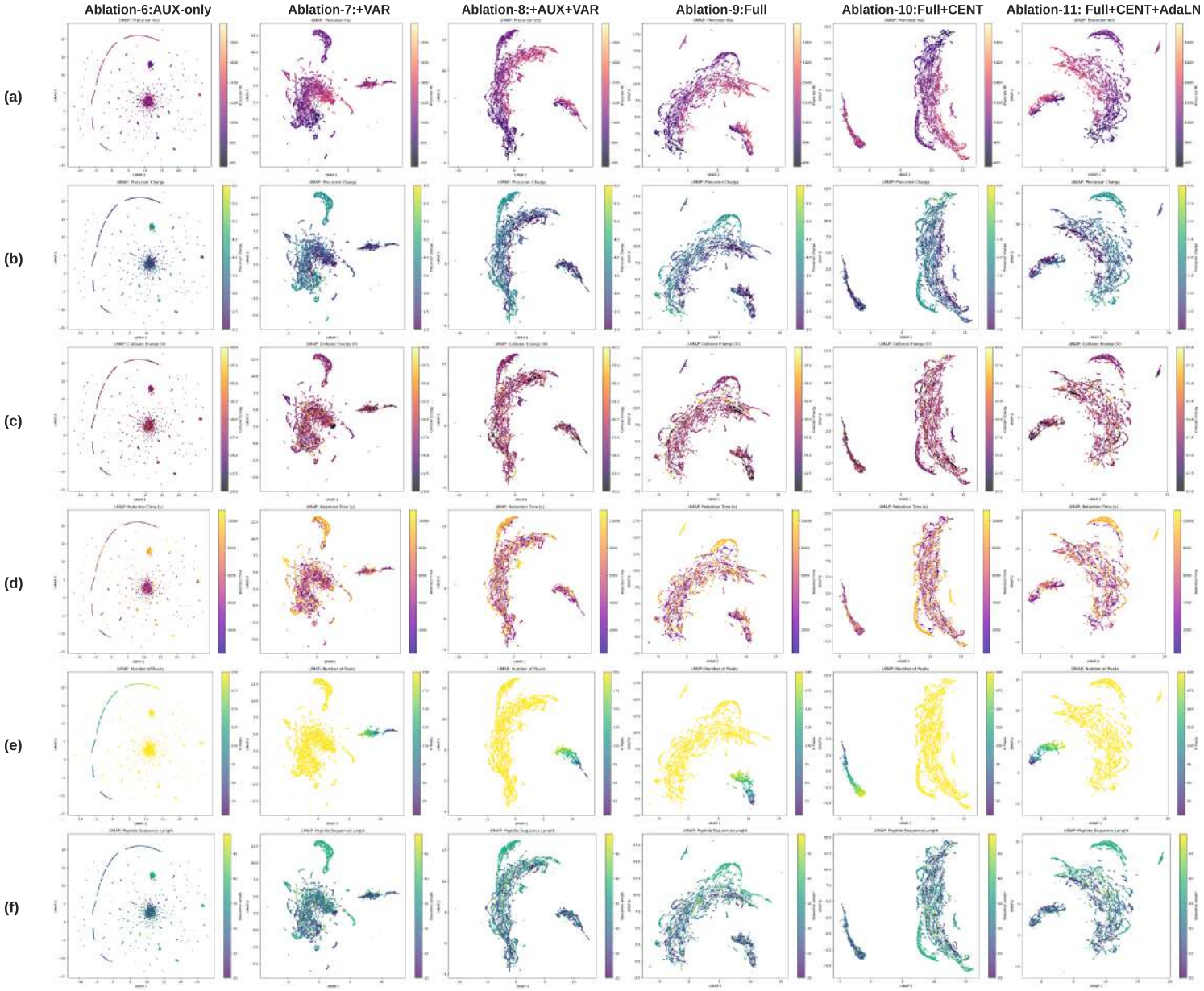


Figure 4.6: **UMAP visualization of latent embeddings across model ablations;Ablation-6:AUX-only, Ablation-7:+VAR, Ablation-8:+AUX+VAR, Ablation-9:Full, Ablation-10:Full+CENT and Ablation-11: Full+CENT+AdaLN.** Columns show observations from different ablation experiments, while rows display the same embedding projection coloured by metadata: (a) precursor mass, (b) precursor charge, (c) collision energy, (d) retention time, (e) number of peaks, and (f) sequence length.

Ablation-5:CL only at 0.852 and Ablation-6:AUX only at 0.867) achieve relatively lower accuracies. The classification performance for experiment name also remains consistently low across all ablations, with accuracies ranging from 0.041 to 0.084, indicating that the model is not capturing noisy signals from the experimental setups used to generate the spectral data.

Similarly, the regression tasks show a clear stratification in performance based on the complexity of the objective functions and the nature of the target physical properties. For precursor m/z, complex multi-loss frameworks yield a significant boost in linear predictability, with Ablation-11: Full+CENT+AdaLN and Ablation-9: Full achieving peak R^2 scores of 0.882 and 0.880, respectively. In contrast, the simpler objectives, like Ablation-1:DF at 0.734, yield relative lower R^2 scores.

Based on the linear probing results, the addition of the variance loss (Ablation-7:+VAR) and contrastive loss (Ablation2:+CL) results in a performance drop compared single-loss noise reconstruction objective (Ablation-1:DF). Specifically, Ablation-7:+VAR yields the lowest R^2 scores for Precursor m/z (0.700), retention time (0.301), and sequence Length (0.460), while also showing reduced precursor charge classification accuracy (0.812). Similarly, Ablation-2:+CL encourages clustering based on dominant categorical properties such as precursor charge, but appears to reduce the linear predictability of finer sequence-level characteristics, as reflected by the lower sequence length score (0.496). These observations demonstrate that while variance-preservation and contrastive losses fundamentally prevent representation collapse, they do not necessarily encourage the model to distribute samples in a semantically meaningful way, a finding consistent with previous studies on latent representations (Chung and Kim, 2026; Bardes et al., 2022).

In addition, the addition of centroid loss (Ablation-10: Full+CENT) introduces a clear performance penalty across both classification and regression tasks compared to Ablation-9: Full. While centroid regularization aims to enforce global class structure in the embedding space by clustering representations around class centroids and reducing intra-class variance, as highlighted by (Wen et al., 2016), this regularization can over-constrain feature variability and compress continuous geometric structure in the latent space. This degrades fine-grained signals such as retention time and precursor m/z, while also reducing the linear separability of class representations such as precursor charge states (Zeng, 2025; Yang et al., 2021).

Overall, the results of the linear probes are consistent with the observations from the UMAPs. In particular, the full multi-objective configurations, Ablation-9:Full and Ablation-11:Full+CENT+AdaLN, demonstrate the strongest overall performance across both classification and regression tasks, indicating that these representations preserve a broader range of physically meaningful information within the latent space.

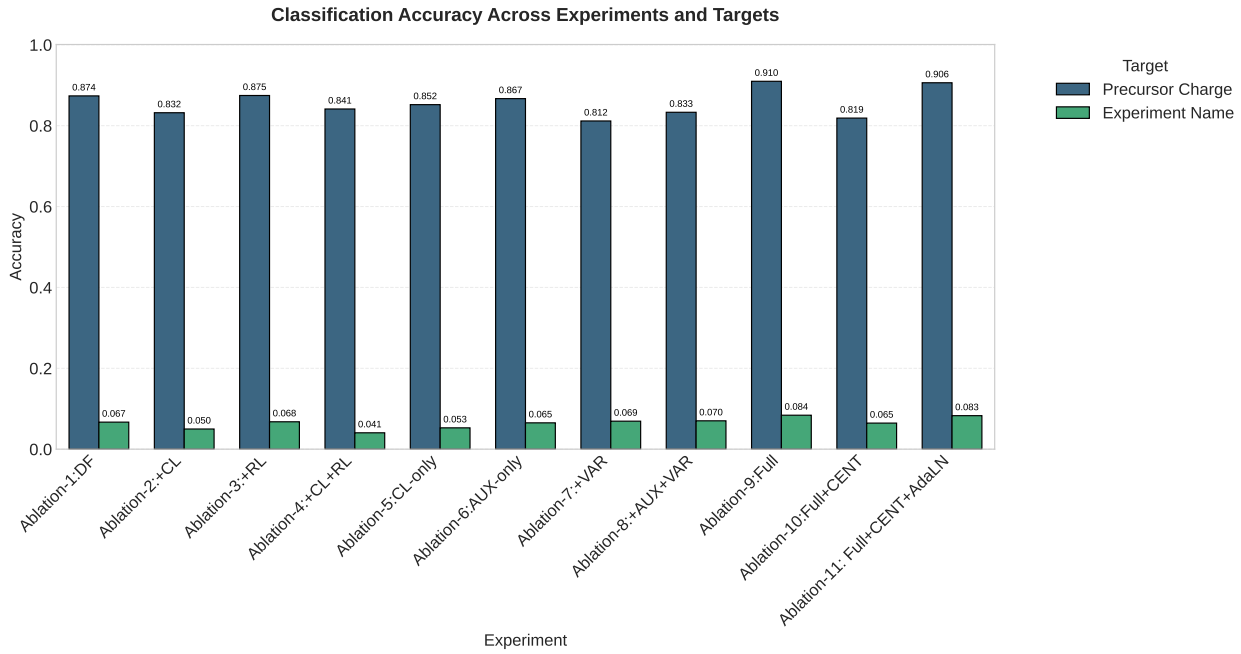


Figure 4.7: Linear Probe Results for classification task based on Ablation Studies

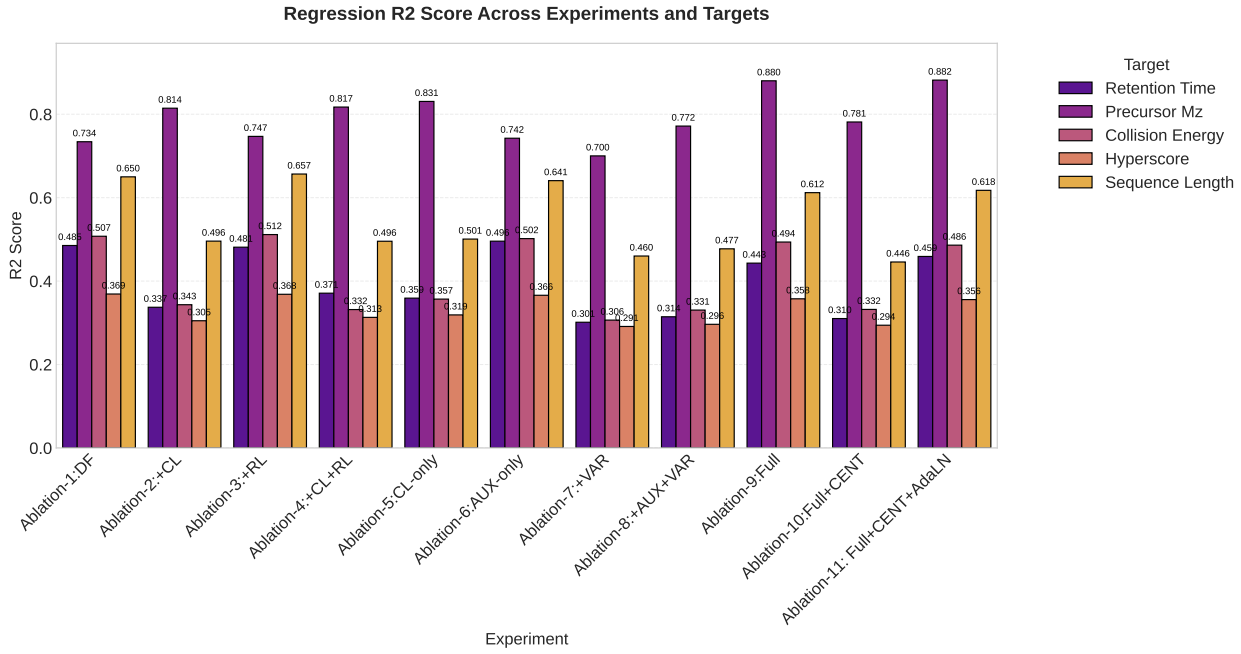


Figure 4.8: Linear Probe results for regression task based Ablation Studies

4.2.3 Embedding statistics

To further evaluate the properties of the learned latent spaces, we report a set of embedding statistics in Table 4.2. These metrics provide complementary evidence to the UMAP visualisations and linear probing results by capturing geometric collapse, and isotropy (reflected by the PCA Top-1 variance

ratio (PCA Top-1), effective dimensionality (Eff. Dim.), and participation ratio (Part. Ratio)) of the learned representations across ablation studies. Across all configurations, the mean embedding norm remains constant at 1.0 due to L2 normalization, ensuring that differences in representation quality are not influenced by scale variations.

Similar to UMAP and linear probe observations, the collapse indicator and PCA statistics reveal clear structural differences across objectives. The simple reconstruction-based objectives (Ablation-1:DF, Ablation-3:+RL, Ablation-6:AUX-only) exhibit high collapse and very low effective dimensionality, indicating that they produce almost one-dimensional representations, further reflected by dominant PCA Top-1 ratios ($\approx$ 0.92 to 0.93).

In contrast, contrastive and variance-based objectives (Ablation-5:CL-only, Ablation-4:+CL+RL, Ablation-7:+VAR, Ablation-8:+AUX+VAR, Ablation-9:Full, Ablation-10:Full+CENT and Ablation-11:Full+CENT+AdaLN) increase participation ratio and effective dimensionality, producing more distributed embeddings. In particular, Ablation-7:+VAR, Ablation-8:+AUX+VAR, Ablation-10:Full+CENT and Ablation-11:Full+CENT+AdaLN achieve the highest dimensionality (up to 50 components) with reduced a PCA dominance, indicating stronger variance spread and reduced collapse. However, we also show that higher dispersion does not always improve structure quality. For example, Ablation-7:+VAR shows a sharp drop in silhouette Score (0.0438), suggesting poor alignment with semantic structure despite high dimensionality. Similarly, CL-only training yields moderate dispersion but limited structural coherence.

The best overall embedding quality is observed for full multi-objective models (Ablation-9:Full and Ablation-11:Full+CENT+AdaLN), which balance dimensionality, collapse reduction, and structure. Centroid regularization (Ablation-10:Full+CENT) further reduces collapse but slightly degrades structural quality (with a reduced silhouette Score (0.0730) and mean similarity (0.4538)), which is consistent with over-constraining intra-class variance.

Generally, the qualitative score aligns with these trends, with higher scores assigned to ablations that maintain both sufficient embedding dispersion and meaningful geometric organisation. These results together with the UMAP visualisations and linear probe tasks, reinforce that optimal performance arises from a balance between preventing collapse and preserving structured variability in the latent space. This result also validates our choice of the multi-objective loss function to train our foundation diffusion model.

Table 4.2: Embedding Statistics summary across ablation experiments.

Ablation	**Mean Norm**	**Collapse Ind.**	**PCA Top-1**	**Part. Ratio**	**Eff. Dim.**	**Silhouette**	**Mean Sim.**	**Qual. Score**
Ablation-1:DF	1.0000	0.9762	0.9241	1.2	1	0.1808	0.9532	0.50
Ablation-2:+CL	1.0000	0.6733	0.1155	17.9	27	0.1734	0.4535	0.90
Ablation-3:+RL	1.0000	0.9793	0.9171	1.2	1	0.2202	0.9594	0.50
Ablation-4:+CL+RL	1.0000	0.7040	0.1524	13.8	22	0.1971	0.4949	0.90
Ablation-5:CL-only	1.0000	0.6692	0.1228	16.4	28	0.1637	0.4481	0.90
Ablation-6:AUX-only	1.0000	0.9692	0.9328	1.1	1	0.1982	0.9398	0.50
Ablation-7:+VAR	1.0000	0.7916	0.0773	22.3	50	0.0438	0.6268	0.80
Ablation-8:+AUX+VAR	1.0000	0.6235	0.0633	30.1	50	0.0707	0.3889	0.90
Ablation-9:Full	1.0000	0.7405	0.0969	22.7	50	0.0996	0.5488	0.90
Ablation-10:Full+CENT	1.0000	0.6735	0.0659	26.9	50	0.0730	0.4538	0.90
Ablation-11: Full+CENT+ AdaLN	1.0000	0.7452	0.0702	27.5	50	0.1017	0.5558	0.90

Mean Norm	Mean ℓ_2 norm of embeddings	Part. Ratio	Participation Ratio
Collapse Ind.	Collapse Index	Eff. Dim.	No. components at 90% variance (effective dimensionality)
PCA Top-1	Top-1 PCA variance ratio	Mean Sim.	Mean pairwise cosine similarity
		Qual. Score	Composite Quality Score

4.3 Embedding Extraction at different diffusion steps

In this section, we evaluate the ability of our model to condition the latent embedding, v_t according to the diffusion time step, t. Specifically, in Figure 4.9, we show that our model based on the Ablation-11+CENT+AdaLN experiment has learned to condition the produced spectrum embeddings, v_t, on the diffusion time step. The results confirm that the latent space evolves across time steps from $t = 1000$ (Eval 1000)to $t = 0$ (Eval 0), with different cluster structures forming at different stages. At the same time, the organisation of samples according to precursor charge and precursor mass changes across the diffusion process, indicating that the latent representation remains strongly conditioned on the diffusion time step. Using these results, we qualitatively show that the diffusion-based encoder progressively transforms the latent representation according to the time steps.

In Table 4.3, we report linear probe performance across diffusion time steps for Ablation-11: Full+CENT+AdaLN. The latent space evolves across steps from $t = 1000$ (noisy time-step) to $t = 0$ (clean time-step), consistent with the denoising process. Overall, the results do not exhibit a monotonic improvement trend across time steps. Instead, several properties peak at intermediate steps. For example, retention time R^2 increases from 0.4145 at Eval 0 to 0.4591 at Eval 100, before slightly decreasing at later steps (with Eval 500 at 0.4501 and Eval 1000 at 0.4542). A similar pattern is observed for sequence length, which peaks at Eval 100 (0.6176) and then slightly declines. Collision energy and hyperscore show relatively stable but slightly varying behaviour across steps, while precursor charge and precursor m/z remain largely stable.

Table 4.3: Linear Probe Results across Time-steps for Ablation-11:Full+CENT+AdaLN

Eval Step	Classification (Accuracy)		Regression (R^2)				
	Experiment Name	Precursor Charge	Collision Energy	Hyperscore	Precursor Mz	Retention Time	Sequence Length
Eval 0	0.0754	0.8949	0.4572	0.3489	0.8752	0.4145	0.5913
Eval 100	0.0829	0.9061	0.4861	0.3556	0.8818	0.4591	0.6176
Eval 500	0.0850	0.9046	0.4866	0.3645	0.8808	0.4501	0.6093
Eval 1000	0.0865	0.9060	0.4955	0.3613	0.8813	0.4542	0.6094

The embedding statistics shown in Table 4.4 are consistent with the linear probe results in Table 4.3. The collapse indicator shows a non-monotonic trend with small variations across time-steps, with a mild peak at Eval 100 (0.7452) compared to Eval 1000 (0.7308) and Eval 0 (0.7114). Similarly, the silhouette score and mean similarity vary slightly across steps, with slightly higher values at Eval 100 (with values of 0.1017 and 0.5558 respectively), but without a consistent monotonic pattern. The participation ratio also remains relatively stable across steps with only minor fluctuations. These results align with Table 4.3, indicating that intermediate steps, such as Eval 100, exhibit a mildly more separable latent structure, although the differences across time-steps remain small.

Together, these trends suggest that the model conditions the representations on the diffusion time step, but that the resulting changes in embedding structure are limited in magnitude. While the latent space properties vary slightly across time steps, the most informative representations are observed at intermediate time-steps, as reflected in Table 4.3, where the best linear probe performance is observed at $t = 100$ (Eval 100). This indicates that stronger semantic structure can emerge at intermediate diffusion steps, which is consistent with previous findings in (Jiang and Cai, 2025).

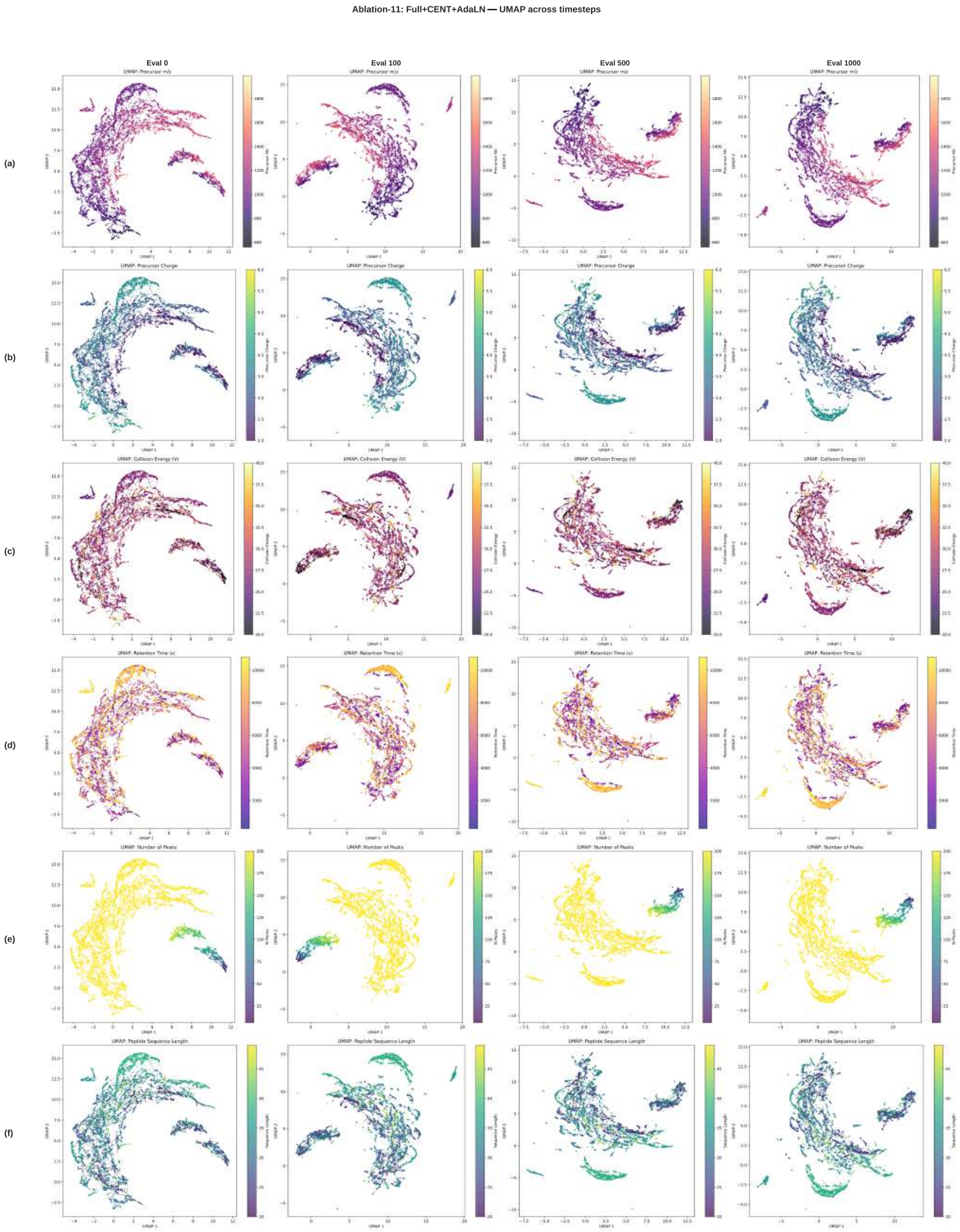


Figure 4.9: **UMAP visualization of latent embeddings at different time-steps base on Ablation-11:Full+CENT+AdaLN.** Columns show different observations at different time steps, while rows display the same embedding projection coloured by metadata: (a) precursor mass, (b) precursor charge, (c) collision energy, (d) retention time, (e) number of peaks, and (f) sequence length.

Table 4.4: Embedding Statistics across different time-steps for Ablation-11: Full+CENT+AdaLN

Experiment	Mean Norm	Collapse Ind.	PCA Top-1	Part. Ratio	Eff. Dim.	Silhouette	Mean Sim.	Qual. Score
Eval 0	1.0000	0.7114	0.0658	28.5	50	0.0848	0.5064	0.90
Eval 100	1.0000	0.7452	0.0702	27.5	50	0.1017	0.5558	0.90
Eval 500	1.0000	0.7257	0.0685	28.1	50	0.0992	0.5271	0.90
Eval 1000	1.0000	0.7308	0.0670	28.7	50	0.0995	0.5344	0.90

Mean Norm	Mean ℓ_2 norm of embeddings	Part. Ratio	Participation Ratio
Collapse Ind.	Collapse Indicator	Eff. Dim.	No. components at 90% variance (effective dimensionality)
PCA Top-1	Top-1 PCA variance ratio	Mean Sim.	Mean pairwise cosine similarity
		Qual. Score	Composite Quality Score

4.4 Foundation Diffusion Model Experiment

In this section, we present the comparison of our trained diffusion model with the baseline foundation model described in section 3.4.1. In Figure 4.10 particularly we show that both the baseline and our foundation model have distinct differences in their latent space structure according to the spectrum physical experimental meta data. Our foundational diffusion model learns a continuous global topology, rather than representing samples through sharply separated clusters, as evidenced by the UMAP visualisations for precursor charge, precursor m/z, sequence length, and number of peaks. Conversely, the baseline model segregates the data into more distinct clusters and isolated regions. While the continuous topology learned by the diffusion model suggests a globally organised representation, the stronger cluster separation observed in the baseline is advantageous given target semantic information contains categorical properties such as precursor charge, fragmentation type, and PTM presence, as evidenced by the baseline's superior performance in Figures 4.12 and 4.13 below.

We also show in Figure 4.11 that both models learnt to organise the latent space in relation to peptide properties, including hydrophobicity, fragmentation type, and post-translational modifications (PTMs). In particular, hydrophobicity, fragmentation type and PTMs follow noisy gradients. However, the Baseline forms significant disconnected clusters, which our foundation model fails to capture in its latent space. In Figures 4.12 and 4.13, we evaluate the performance of our foundation model relative to the baseline. For the regression tasks, our foundation model achieves an R^2 of 0.784 for precursor m/z, compared to 0.923 for the baseline. However, a larger performance gap is observed for hydrophobicity prediction, where the model achieves an R^2 of 0.137 compared to 0.616 for the baseline. Similarly, the model shows lower performance on the classification tasks, with accuracies of 0.647 and 0.624 for precursor charge and fragmentation type respectively, compared to the baseline values of 0.942 and 0.807. This lower classification performance is consistent with the latent space analysis observed in the UMAPs, where the diffusion model exhibits a simple more continuous organisation, while the baseline forms more distinct clusters that support for class separability.

This performance gap can primarily be attributed to the model's architectural constraint which involves the use of a single [LATENT]-token bottleneck, where up to 200 spectral peaks are compressed into a single 768-dimensional token that simultaneously serves as the downstream embedding as well as the only cross-attention memory for the DiT, and the AdaLN conditioning source. This imposes a strong information bottleneck, effectively limiting the cross-attention mechanism to a global conditioning signal rather than fine-grained token-level interactions. In addition, centroid regularisation and AdaLN-conditioned decoding may further contribute to this, because while these components are beneficial for learning compact latent representations, these mechanisms may also restrict fine-grained separability in the embedding space.

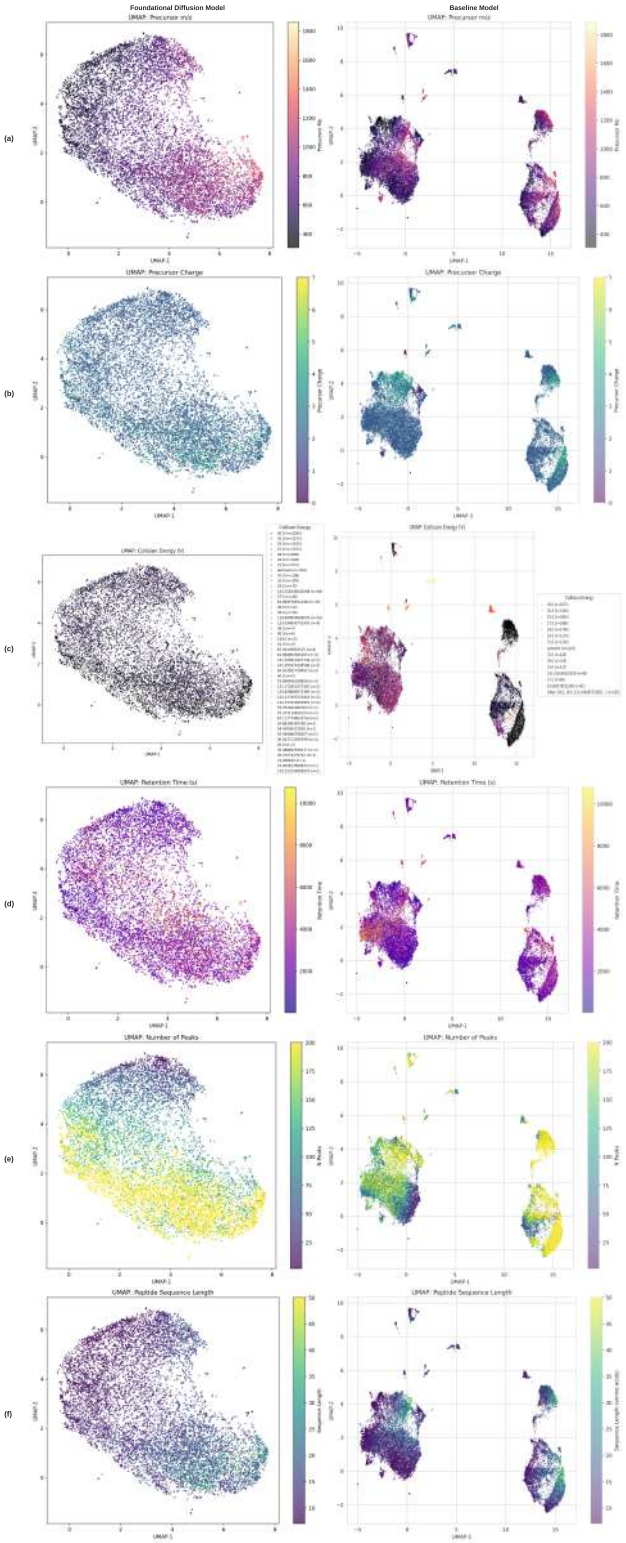


Figure 4.10: **UMAP visualization of latent embeddings from Foundation Diffusion Model and Baseline.** Columns show different observations from our foundation model and the baseline, while rows display the same embedding projection coloured by metadata: (a) precursor mass, (b) precursor charge, (c) collision energy, (d) retention time, (e) number of peaks, and (f) sequence length.

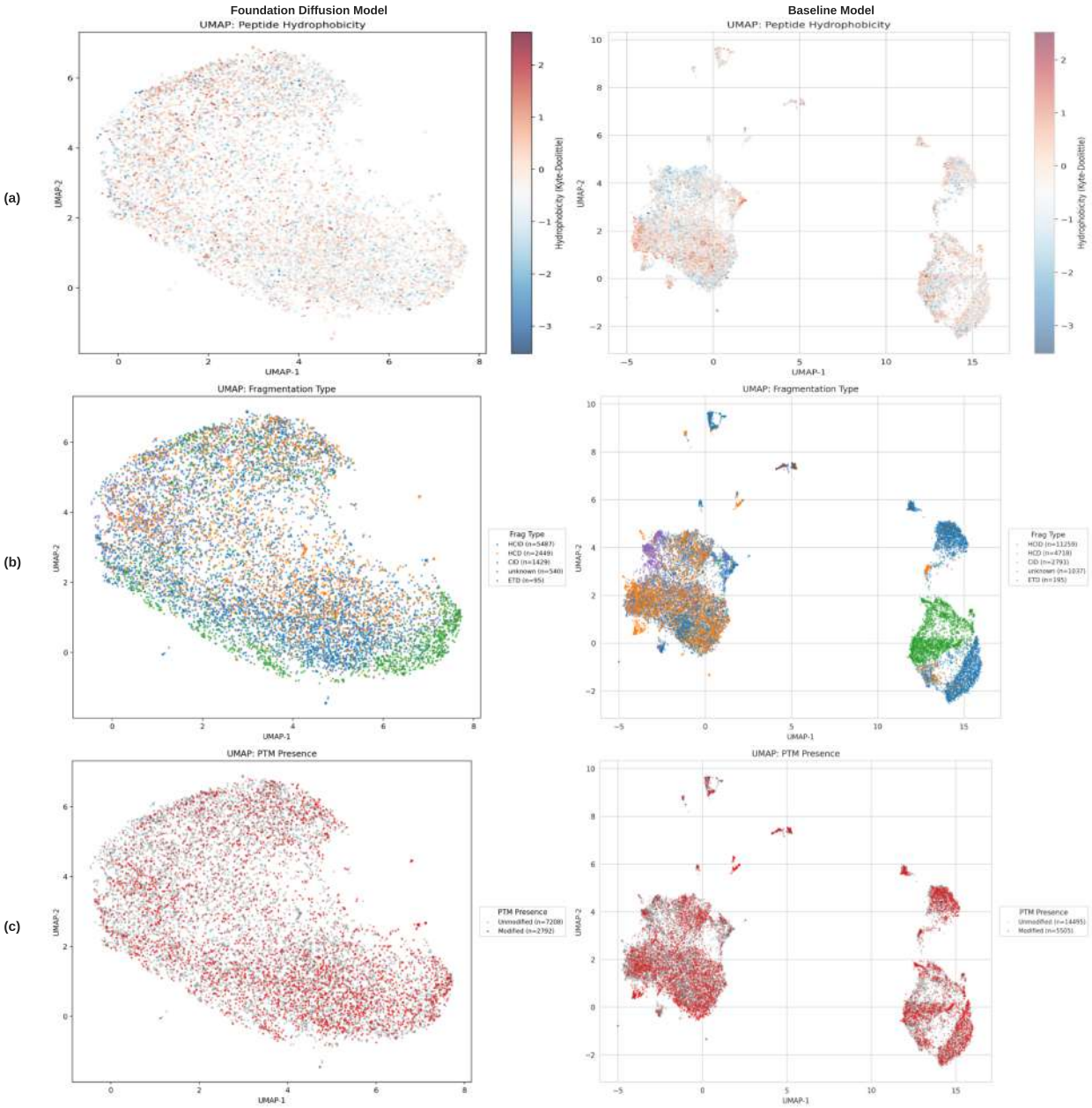


Figure 4.11: **UMAP visualization of latent embeddings from Foundation Diffusion Model and Baseline.** Columns show different observations from our foundation model and the baseline, while rows display the same embedding projection coloured by metadata: (a)hydrophobicity, (b) fragmentation type, (c) post-translational modification presence.

It is also important to note that our model uses a relatively small transformer architecture (with a 4-layer latent encoder (which is a bottleneck) and a 6-layer DiT) compared to the baseline model, which is a larger 9-layer transformer encoder. Previous studies also suggest that such differences in architectural scale can limit model performance (Liang et al., 2026). The current foundation model implementation is also likely limited by the fact that several hyperparameters used were adopted from the small-scale ablation experiments, and may not be optimal for training the foundation model on the large-scale dataset(18 million samples). The use of these parameters is due to the limited computational resources, which prevented extensive large-scale hyperparameter tuning and experimentation. The mechanistic interpretability analysis and confirmation of the model's exact failure mechanisms is, however, an area we propose for future research.

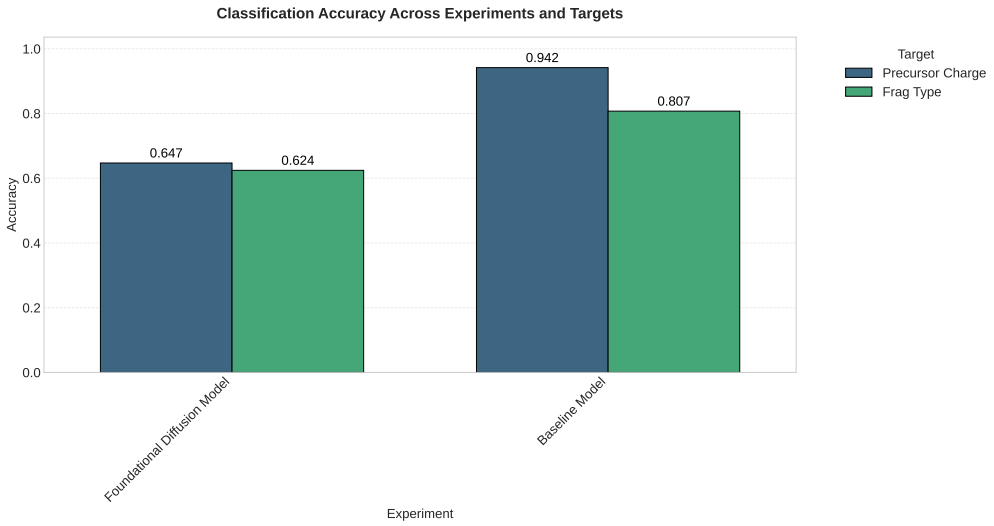


Figure 4.12: Linear Probe accuracy for classification task based on the Foundation Diffusion Model and Baseline

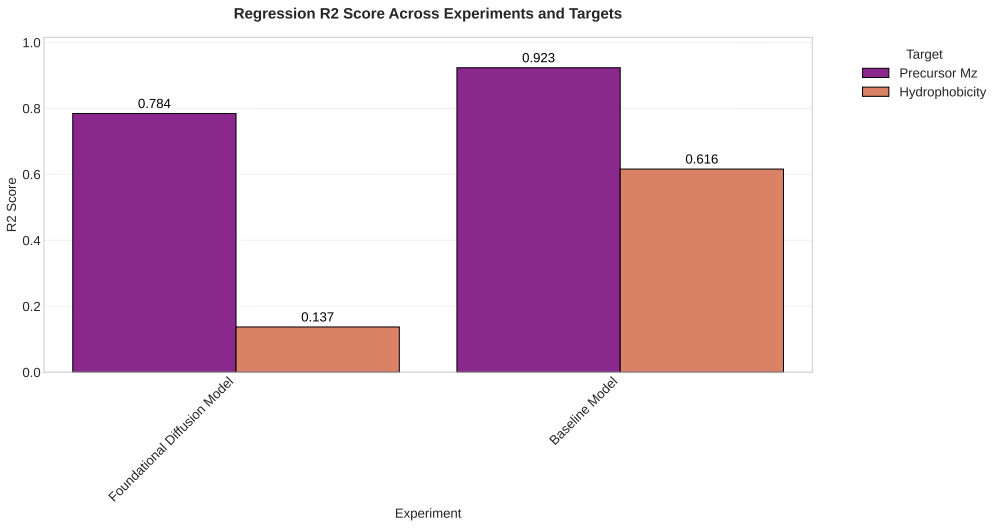


Figure 4.13: Linear Probe results for regression task based on the Foundation Diffusion Model and Baseline

The embedding statistics in Table 4.5 also confirm the observations from the linear probes. In comparison to the baseline (with an effective dimensionality of 160 and participation ratio of 24.0), our foundation model exhibits an effective dimensionality of 50, indicating that variance is concentrated in fewer principal directions. However, the participation ratio of our model is higher (28.3 compared to baseline with 24.0), suggesting that variance is more evenly distributed across the active subspace. In contrast, the baseline shows a higher effective dimensionality, reflecting a broader spread of variance across a larger number of dimensions. It important to mention that the metrics used capture different aspects of latent dispersion, with effective dimensionality reflecting the number of active directions and participation ratio reflecting how evenly variance is distributed across them. The observed reduction in effective dimensionality in our model's latent representation may be a consequence of the denoising objective, which we hypothesise encourages the model to filter out noise in the latent space. However, this effect may also arise from architectural factors such as the latent bottleneck and regularisation terms, rather than solely reflecting noise removal. Generally, we show that the two models organise variance in distinct ways within the latent space.

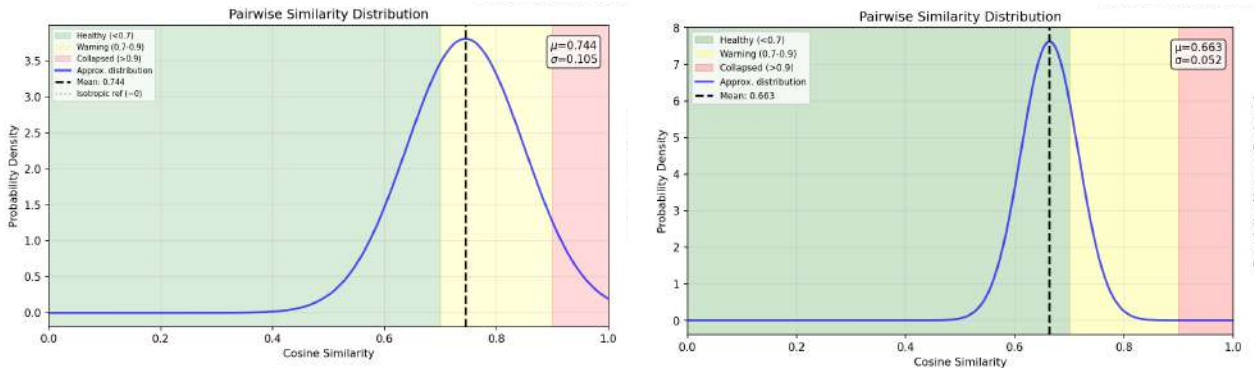
Table 4.5: Embedding Statistics summary across Foundation Diffusion Model and Baseline.

Experiment	**Mean Norm**	**Collapse Ind.**	**PCA Top-1**	**Part. Ratio**	**Eff. Dim.**	**Silhouette**	**Mean Sim.**	**Qual. Score**
Foundational Diffusion Model	1.0000	0.8146	0.0732	28.3	50	0.0066	0.6634	0.70
Baseline Model	nan	nan	0.1537	24.0	160	nan	0.7445	nan

Mean Norm	Mean ℓ_2 norm of embeddings	Part. Ratio	Participation Ratio
Collapse Ind.	Collapse Indicator	Eff. Dim.	No. components at 90% variance (effective dimensionality)
PCA Top-1	Top-1 PCA variance ratio	Mean Sim.	Mean pairwise cosine similarity
		Qual. Score	Composite Quality Score

Note: Due to limited access to the baseline model, some evaluation metrics were not available at the time of this report.

The pairwise similarity distribution of the learned embeddings in Figure 4.14 shows differences in representation geometry between the baseline and our foundation model. The baseline exhibits a higher mean pairwise similarity (0.7445), indicating more closely aligned embeddings on average, while our model shows lower mean similarity (0.6634), reflecting a more dispersed embedding distribution. These observations also suggest that the two models organise representations differently in latent space, and are consistent with the UMAP visualisations, which show continuous topology for our model and distinct clusters for the baseline.



(a) Baseline Model (b) Foundation Diffusion Model

Figure 4.14: Pairwise Similarity Distribution of Embeddings from foundation diffusion model and baseline.

From the above results, we recognize the lower performance of our model on classification tasks compared to the baseline. To contextualise this, we further assess our model against the majority-class and random baselines shown in Table 4.6. The majority-class baseline reflects the class imbalance observed in section 3.1, where the distribution of the data according to precursor charge has a dominant class (charge 2), while the random baseline provides a reference for lower-bound performance. This assessment shows that for precursor charge, the majority-class baseline achieves an accuracy of 0.734, which is higher than our model (0.647), while the random baseline achieves 0.125. For fragmentation type, the majority-class baseline achieves a value of 0.547 compared to our model (0.624), while the random baseline shows 0.250. Compared to these baselines, our model is above the random baseline but does not exceed the majority-class baseline for precursor charge, indicating its performance below a trivial (majority-class) baseline under the current experimental setup. The results show that the current model implementation learnt meaningful predictive information better than a random predictor. However, its failure against the majority-class baseline for the precursor charge suggests that the model is not effectively exploiting the input features to distinguish between the minority classes.

Table 4.6: Classification baselines summary across target variables.

Target Variable	**Majority Class Baseline**	**Random Baseline**
Precursor charge	0.734	0.125
Frag Type	0.547	0.250
Note:	The reported values are based on a random sample of 250,000 points drawn from the full dataset. Due to computational and data access constraints, computing baselines on the full dataset was not feasible; therefore, the values shown are approximate estimates derived from this subset.	

5. Conclusion, Limitations and Future Work

5.1 Conclusion

Our results show that the proposed Foundation Diffusion-based encoder model learns a more compact latent space compared to the baseline. However, the model trails the baseline in regression performance, achieving a coefficient of determination (R^2) of 0.784 for precursor m/z, compared to 0.923 for the baseline. In terms of variance, this corresponds to a higher proportion of unexplained variance (21.6% relative to 7.7% for the baseline), indicating that while the model captures a meaningful signal, it still leaves a larger proportion of the underlying structure unmodelled. For the classification task, the foundation model achieves an accuracy of 0.647 for precursor charge on the full dataset, which is below the majority-class baseline of approximately 0.734, reflecting the strong class imbalance where charge 2 dominates the training distribution. For fragmentation type, the model achieves an accuracy of 0.624, which is above the majority-class baseline of 0.547 and the random baseline of 0.250, indicating better-than-trivial performance for this task. In contrast, the InstaDeep baseline achieves 0.942, demonstrating strong performance well above trivial predictors. The higher accuracy (0.910) observed in the smaller-scale ablation studies reflects a different experimental setting and is therefore not directly representative of the full dataset setting. This discrepancy highlights the sensitivity of the model to dataset scale and class imbalance, and suggests that the current implementation does not yet yield robust predictive representations under real-world distributions. Overall, while the diffusion-based encoder using masking and the multi-loss objective supports learning organised and compact latent representations, additional mechanisms,such as parameter tuning at scale, are required to improve predictive performance, particularly under imbalanced classification settings and at the full dataset scale. In addition, the results show that the centroid loss requires mechanisms, such as AdaLN modulation, to mitigate its performance penalty.

5.2 Limitations

A limitation of this study is the restricted comparison with the baseline model, as access to the baseline implementation and evaluation pipeline was limited. Consequently, some metrics and analysis could not be computed under identical conditions. In addition, the experiments were conducted on a single mass spectrometry dataset, which may limit the generalisation of the findings to other datasets. Furthermore, the vast majority of experiments and ablation studies were performed on personal compute resources (a single H100 GPU Colab runtime with limited capacity), with access to a 4×H100 compute node only available during the final two weeks of the project and used indirectly through the project supervisor, which limited its use to final evaluations rather than full model development and experimentation.

5.3 Future Work

We recommend that future work should investigate the interpretability of foundation diffusion-based encoder models for mass spectrometry data to better understand the biological and proteomic information captured by the learned representations. In addition, evaluating a wider range of model architectures and scales may provide further insight into the relationship between model capacity and representation quality. The potential impact of score-based diffusion objectives on latent representation learning for mass spectrometry also needs further investigation. Finally, assessing the proposed approach across more datasets and benchmarks would provide a broader evaluation of its robustness and transferability.

Acknowledgements

I would like to thank AIMS for providing an excellent academic and research environment, and for supporting this work. I also extend my gratitude to Google DeepMind for their support through the Google DeepMind Scholarship, and to my supervisor, Divanisha Patel, Jeroen Van Goey, Jemma Daniel, and Konstantinos Kalogeropoulos from InstaDeep, for their invaluable guidance and contribution to the direction and success of this work.

I would further like to thank the entire InstaDeep team for providing their codebase for the preprocessing pipeline as well as the baseline model used in this study.

References

Aebersold, R. and Mann, M. Mass spectrometry-based proteomics. *Nature*, 422:198–207, 04 2003. doi: 10.1038/nature01511.

Bardes, A., Ponce, J., and LeCun, Y. Vicreg: Variance-invariance-covariance regularization for self-supervised learning, 2022. URL https://arxiv.org/abs/2105.04906.

Bittremieux, W., Ananth, V., Fondrie, W. E., Melendez, C., Pominova, M., Sanders, J., Wen, B., Yilmaz, M., and Noble, W. S. Deep learning methods for de novo peptide sequencing. *Mass Spectrometry Reviews*, 45(3):507–526, 2026. doi: https://doi.org/10.1002/mas.21919. URL https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/abs/10.1002/mas.21919.

Bushuiev, R., Bushuiev, A., Samusevich, R., Brungs, C., Sivic, J., and Pluskal, T. Self-supervised learning of molecular representations from millions of tandem mass spectra using dreams. *Nature Biotechnology*, pages 1–11, 05 2025. doi: 10.1038/s41587-025-02663-3.

Cao, H., Chen, M., Han, Y., and Xu, R. Diffusion models for adapted sequential data generation. In *NeurIPS 2025 Workshop MLxOR: Mathematical Foundations and Operational Integration of Machine Learning for Uncertainty-Aware Decision-Making*, 2025. URL https://openreview.net/forum?id=2pfnuv913O.

Chai, C., Jin, K., Tang, N., Fan, J., Qiao, L., Wang, Y., Luo, Y., Yuan, Y., and Wang, G. Mitigating data scarcity in supervised machine learning through reinforcement learning guided data generation. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pages 3613–3626, 2024. doi: 10.1109/ICDE60146.2024.00278.

Chen, R., Guo, J., Wang, H., Xiang, Y., Xian, Y., and Yu, Z. Energy-balanced hyperspherical graph representation learning via structural binding and entropic dispersion, 2026. URL https://arxiv.org/abs/2512.24062.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. E. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709, 2020. URL https://arxiv.org/abs/2002.05709.

Chen, X. and He, K. Exploring simple siamese representation learning. *CoRR*, abs/2011.10566, 2020. URL https://arxiv.org/abs/2011.10566.

Chen, X., Liu, Z., Xie, S., and He, K. Deconstructing denoising diffusion models for self-supervised learning, 2024. URL https://arxiv.org/abs/2401.14404.

Cho, W. C. Proteomics technologies and challenges. *Genomics, Proteomics & Bioinformatics*, 5(2):77–85, 06 2007. ISSN 1672-0229. doi: 10.1016/S1672-0229(07)60018-7. URL https://doi.org/10.1016/S1672-0229(07)60018-7.

Chung, J. and Kim, S. J. Global geometry is not enough for vision representations, 2026. URL https://arxiv.org/abs/2602.03282.

de Jonge, N., van der Hooft, J. J. J., and Probst, D. To bin or not to bin: Alternative representations of mass spectra, 2025. URL https://arxiv.org/abs/2502.10851.

Domon, B. and Aebersold, R. Challenges and opportunities in proteomics data analysis. *Molecular & cellular proteomics : MCP*, 5:1921–6, 11 2006. doi: 10.1074/mcp.R600012-MCP200.

Du, P., Stolovitzky, G., Horvatovich, P., Bischoff, R., Lim, J., and Suits, F. A noise model for mass spectrometry based proteomics. *Bioinformatics*, 24(8):1070–1077, 04 2008.

Eijkelboom, F., Bartosh, G., Naesseth, C. A., Welling, M., and van de Meent, J.-W. Variational flow matching for graph generation, 2025. URL https://arxiv.org/abs/2406.04843.

Eloff, K., Kalogeropoulos, K., Mabona, A., Morell, O., Catzel, R., Rivera-de Torre, E., Jespersen, J., Williams, W., Beljouw, S., Skwark, M., Laustsen, A., Brouns, S., Ljungars, A., Schoof, E., Van Goey, J., auf dem Keller, U., Beguir, K., Carranza, N., and Jenkins, T. Instanovo enables diffusion-powered de novo peptide sequencing in large-scale proteomics experiments. *Nature Machine Intelligence*, 7: 565–579, 03 2025. doi: 10.1038/s42256-025-01019-5.

Feng, G., Geng, Y., Guan, J., Wu, W., Wang, L., and He, D. Theoretical benefit and limitation of diffusion language model, 2025. URL https://arxiv.org/abs/2502.09622.

Franceschi, L., Donini, M., Perrone, V., Klein, A., Archambeau, C., Seeger, M., Pontil, M., and Frasconi, P. Hyperparameter optimization in machine learning. *Foundations and Trends® in Machine Learning*, 18(6):1054–1201, 2025. ISSN 1935-8245. doi: 10.1561/2200000088. URL http://dx.doi.org/10.1561/2200000088.

Fröhlich, K., Fahrner, M., Brombacher, E., Seredynska, A., Maldacker, M., Kreutz, C., Schmidt, A., and Schilling, O. Data-independent acquisition: A milestone and prospect in clinical mass spectrometry–based proteomics. *Molecular & Cellular Proteomics*, 23(8):100800, 2024. ISSN 1535-9476. doi: https://doi.org/10.1016/j.mcpro.2024.100800. URL https://www.sciencedirect.com/science/article/pii/S1535947624000902.

Gat, I., Remez, T., Shaul, N., Kreuk, F., Chen, R. T. Q., Synnaeve, G., Adi, Y., and Lipman, Y. Discrete flow matching. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=GTDKo3Sv9p.

Griss, J., Perez-Riverol, Y., Lewis, S., Tabb, D. L., Dianes, J. A., del Toro, N., Rurik, M., Walzer, M., Kohlbacher, O., Hermjakob, H., Wang, R., and Vizcaíno, J. A. Recognizing millions of consistently unidentified spectra across hundreds of shotgun proteomics datasets. *Nature Methods*, 13(8):651–656, 2016. doi: 10.1038/nmeth.3902.

Hayakawa, S., Takida, Y., Imaizumi, M., Wakaki, H., and Mitsufuji, Y. Distillation of discrete diffusion through dimensional correlations, 2025. URL https://arxiv.org/abs/2410.08709.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *CoRR*, abs/2006.11239, 2020. URL https://arxiv.org/abs/2006.11239.

Hoogeboom, E., Nielsen, D., Jaini, P., Forré, P., and Welling, M. Argmax flows and multinomial diffusion: Learning categorical distributions, 2021. URL https://arxiv.org/abs/2102.05379.

Hosseini-Asl, E., Zurada, J. M., and Nasraoui, O. Deep learning of part-based representation of data using sparse autoencoders with nonnegativity constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 27(12):2486–2498, Dec. 2016. ISSN 2162-2388. doi: 10.1109/tnnls.2015.2479223. URL http://dx.doi.org/10.1109/TNNLS.2015.2479223.

InstaDeep. Instadeep foundation transformer encoder (unpublished model), 2026. Unpublished internal model, accessed under restricted availability.

Jiang, L. and Cai, Y. *Automated Learning of Semantic Embedding Representations for Diffusion Models*, page 1–10. Society for Industrial and Applied Mathematics, Jan. 2025. ISBN 9781611978520. doi: 10.1137/1.9781611978520.1. URL http://dx.doi.org/10.1137/1.9781611978520.1.

Jolicoeur-Martineau, A., Li, K., Piché-Taillefer, R., Kachman, T., and Mitliagkas, I. Gotta go fast when generating data with score-based models. *CoRR*, abs/2105.14080, 2021. URL https://arxiv.org/abs/2105.14080.

Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., and Levy, O. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020. doi: 10.1162/tacl_a_00300. URL https://aclanthology.org/2020.tacl-1.5/.

Kharyuk, P., Nazarenko, D., Oseledets, I., Rodin, I., Shpigun, O., Tsitsilin, A., and Lavrentiev, M. Employing fingerprinting of medicinal plants by means of lc-ms and machine learning for species identification task. *Scientific Reports*, 8, 11 2018. doi: 10.1038/s41598-018-35399-z.

Lai, C.-H., Song, Y., Kim, D., Mitsufuji, Y., and Ermon, S. The principles of diffusion models, 2025. URL https://arxiv.org/abs/2510.21890.

LeCun, Y., Cortes, C., and Burges, C. Mnist handwritten digit database. *ATT Labs [Online]*, 2, 2010.

Leske, M., Bottacini, F., de Castro Cuadrat, R. R., Afli, H., and Andrade, B. G. N. On the impact of embedding anisotropy in genomic language models for bacterial taxonomy. In *ICLR 2026 Workshop on Generative Models and Language Models in Genomics (Gen²)*, 2026. URL https://openreview.net/pdf?id=C7DlAqrKxE.

Levner, I. Feature selection and nearest centroid classification for protein mass spectrometry. *BMC bioinformatics*, 6:68, 02 2005. doi: 10.1186/1471-2105-6-68.

Li, X., Zhang, J., Zhang, S., Chen, T., Lin, L., and Wang, G. In-situ tweedie discrete diffusion models, 2025. URL https://arxiv.org/abs/2510.01047.

Liang, X., Ma, W., Paquet, E., Viktor, H., and Michalowski, W. Prot-gfdm: A generative fractional diffusion model for protein generation. *Computational and Structural Biotechnology Journal*, 27: 3464–3480, 2025. doi: 10.1016/j.csbj.2025.07.045. URL https://spj.science.org/doi/abs/10.1016/j.csbj.2025.07.045.

Liang, Z., He, H., Yang, C., and Dai, B. Scaling laws for diffusion transformers, 2026. URL https://arxiv.org/abs/2410.08184.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Madiona, R., Alexander, D., Winkler, D., Muir, B., and Pigram, P. Information content of tof-sims data: Effect of spectral binning. *Applied Surface Science*, 493:1067–1074, Nov. 2019. ISSN 0169-4332. doi: 10.1016/j.apsusc.2019.07.044.

Marty, M. T. Fundamentals: How do we calculate mass, error, and uncertainty in native mass spectrometry? *Journal of the American Society for Mass Spectrometry*, 33(10):1807–1812, 2022. doi: 10.1021/jasms.2c00218. URL https://doi.org/10.1021/jasms.2c00218. PMID: 36130030.

McInnes, L., Healy, J., and Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction, 2020. URL https://arxiv.org/abs/1802.03426.

Nakkiran, P., Bradley, A., Zhou, H., and Advani, M. Step-by-step diffusion: An elementary tutorial, 2024. URL https://arxiv.org/abs/2406.08929.

Nichol, A. and Dhariwal, P. Improved denoising diffusion probabilistic models, 2021. URL https://arxiv.org/abs/2102.09672.

Ning, K., Fermin, D., and Nesvizhskii, A. I. Computational analysis of unassigned high-quality ms/ms spectra in proteomic data sets. *PROTEOMICS*, 10(14):2712–2718, 2010. doi: https://doi.org/10.1002/pmic.200900473. URL https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/abs/10.1002/pmic.200900473.

Patel, Z., DeLoye, J., and Mathias, L. Exploring diffusion and flow matching under generator matching, 2024. URL https://arxiv.org/abs/2412.11024.

Perez, E., Strub, F., de Vries, H., Dumoulin, V., and Courville, A. Film: Visual reasoning with a general conditioning layer, 2017. URL https://arxiv.org/abs/1709.07871.

Rousseeuw, P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987. ISSN 0377-0427. doi: https://doi.org/10.1016/0377-0427(87)90125-7. URL https://www.sciencedirect.com/science/article/pii/0377042787901257.

Russo, D., Roy, B. V., Kazerouni, A., Osband, I., and Wen, Z. A tutorial on thompson sampling, 2020. URL https://arxiv.org/abs/1707.02038.

Sanders, J., Yilmaz, M., Russell, J. H., Bittremieux, W., Fondrie, W. E., Riley, N. M., Oh, S., and Noble, W. S. Foundation model for mass spectrometry proteomics, 2025. URL https://arxiv.org/abs/2505.10848.

Schiff, Y., Sahoo, S. S., Phung, H., Wang, G., Boshar, S., Dalla-torre, H., de Almeida, B. P., Rush, A., Pierrot, T., and Kuleshov, V. Simple guidance mechanisms for discrete diffusion models, 2025. URL https://arxiv.org/abs/2412.10193.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models, 2022. URL https://arxiv.org/abs/2010.02502.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *CoRR*, abs/1907.05600, 2019. URL http://arxiv.org/abs/1907.05600.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations, 2021. URL https://arxiv.org/abs/2011.13456.

Syed, H. F. How ai learns hidden patterns: The power of embeddings in predictions. *International Journal on Science and Technology*, 2025. URL https://api.semanticscholar.org/CorpusID:277389695.

Tbahriti, H. F., Boukadoum, A., and Eljamil, E. M. A. M. Generative mass spectrometry via physics-informed machine learning: A framework for biomarker discovery in neurodegenerative diseases. *International Journal of Mass Spectrometry*, 519:117551, 2026. ISSN 1387-3806. doi: https://doi.org/10.1016/j.ijms.2025.117551. URL https://www.sciencedirect.com/science/article/pii/S1387380625001551.

Urban, J. and Štys, D. Noise and baseline filtration in mass spectrometry. In Ortuño, F. and Rojas, I., editors, *Bioinformatics and Biomedical Engineering*, pages 418–425, Cham, 2015. Springer International Publishing. ISBN 978-3-319-16480-9.

van den Oord, A., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018. URL http://arxiv.org/abs/1807.03748.

Voronov, G., Lightheart, R., Davison, J., Krettler, C. A., Healey, D., and Butler, T. Multi-scale sinusoidal embeddings enable learning on high resolution mass spectrometry data. *arXiv preprint arXiv:2207.02980*, 2022.

Wang, L., Rong, Y., Xu, T., Zhong, Z., Liu, Z., Wang, P., Zhao, D., Liu, Q., Wu, S., Wang, L., and Zhang, Y. Diffspectra: Molecular structure elucidation from spectra using diffusion models, 2025. URL https://arxiv.org/abs/2507.06853.

Wen, B. and Noble, W. A multi-species benchmark for training and validating mass spectrometry proteomics machine learning models. *Scientific Data*, 11, 11 2024. doi: 10.1038/s41597-024-04068-4.

Wen, Y., Zhang, K., Li, Z., and Qiao, Y. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, 2016. URL https://api.semanticscholar.org/CorpusID:4711865.

Yang, L., Wang, Y., Liu, L., Wang, P., Chi, L., Yuan, Z., Wang, C., and Zhang, Y. Center prediction loss for re-identification, 2021. URL https://arxiv.org/abs/2104.14746.

Yao, L. H., Li, Y., and Liu, S. A minimal model of representation collapse: Frustration, stop-gradient, and dynamics, 2026. URL https://arxiv.org/abs/2604.09979.

Yilmaz, M., Fondrie, W. E., Bittremieux, W., Nelson, R., Ananth, V., Oh, S., and Noble, W. S. Sequence-to-sequence translation from mass spectra to peptides with a transformer model. *bioRxiv*, 2023. doi: 10.1101/2023.01.03.522621. URL https://www.biorxiv.org/content/early/2023/01/04/2023.01.03.522621.

Yu, R., Li, Q., and Wang, X. Discrete diffusion in large language and multimodal models: A survey, 2025. URL https://arxiv.org/abs/2506.13759.

Yuan, W., Jeong, S., and Aghazadeh, A. On the error-correcting effects of stochasticity in discrete diffusion, 2026. URL https://arxiv.org/abs/2605.26582.

Zeng, D. Comparing contrastive and triplet loss: Variance analysis and optimization behavior, 2025. URL https://arxiv.org/abs/2510.02161.

Zhao, B., Zhang, X., Dong, Y., Lu, X., Lin, W., and Zhu, Q. Delay flow matching. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=6lH1XblLpo.

Zhou, J., Wang, S., Li, S., Wang, W., Chen, S., Xia, J., Yu, C., and Li, S. Z. Peaknovo: Towards the robust de novo peptide sequencing for missing spectral peaks, 2026. URL https://openreview.net/forum?id=0oqEBQA0UD.

Appendix A. Sanity Check

A.1 Additional UMAP Visualization for sanity check

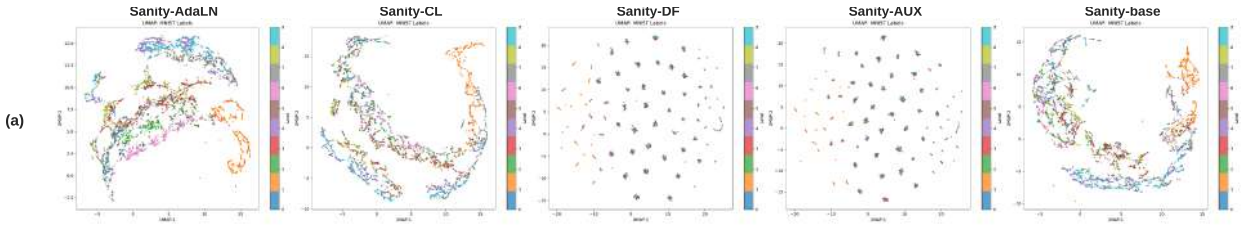


Figure A.1: **UMAP visualization of latent embeddings across sanity check experiments.** Columns show different experiment variants, while the row displays the same embedding projection coloured by: (a) MNIST Labels.

A.2 Embedding Statistics at different Time Steps

Table A.1: Embedding Statistics summary across Sanity-AdaLN experiment.

Experiment	Mean Norm	Collapse Ind.	PCA Top-1	Part. Ratio	Eff. Dim.	Silhouette	Mean Sim.	Qual. Score
Eval 0	1.0000	0.7803	0.2080	13.4	46	0.1258	0.6102	0.90
Eval 100	1.0000	0.7315	0.2090	13.5	44	0.1327	0.5362	0.90
Eval 500	1.0000	0.7098	0.2009	13.7	43	0.1304	0.5050	0.90
Eval 1000	1.0000	0.7552	0.1897	15.2	46	0.1248	0.5715	0.90

Mean Norm	Mean ℓ_2 norm of embeddings	Part. Ratio	Participation Ratio
Collapse Ind.	Collapse Indicator	Eff. Dim.	No. components at 90% variance (effective dimensionality)
PCA Top-1	Top-1 PCA variance ratio	Mean Sim.	Mean pairwise cosine similarity
		Qual. Score	Composite Quality Score

A.3 Additional Linear Probe Results for sanity check

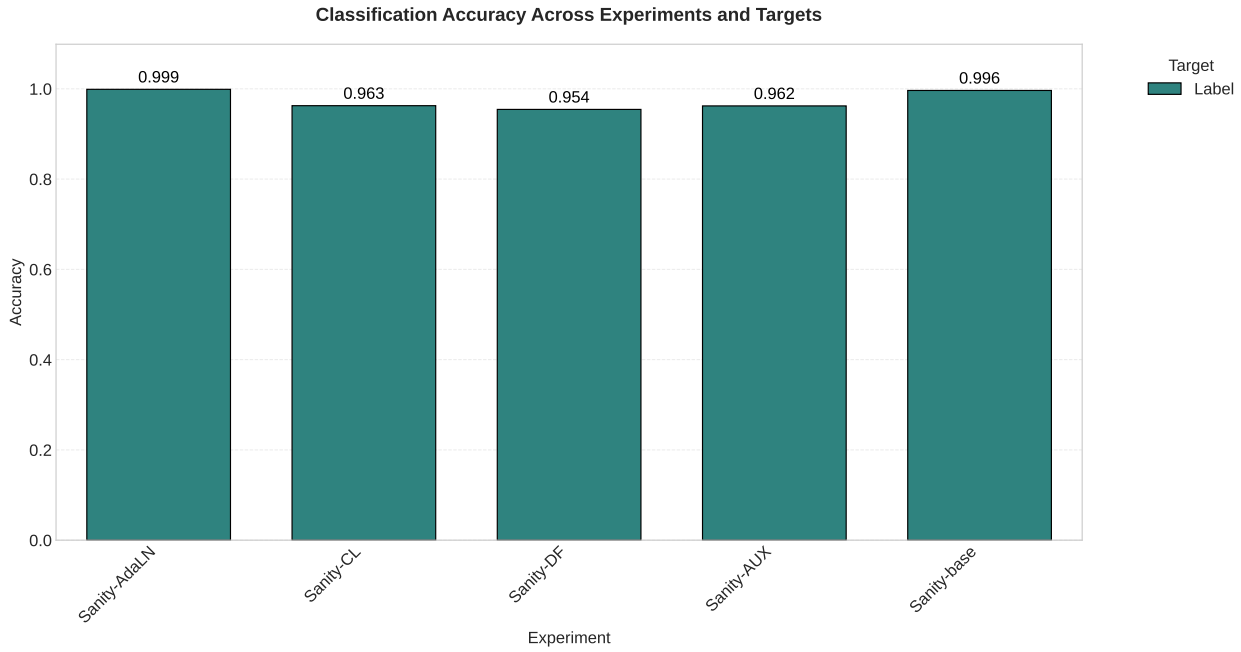


Figure A.2: Linear Probe Results for classification task based on sanity check experiments

A.4 Training and validation loss results for sanity check

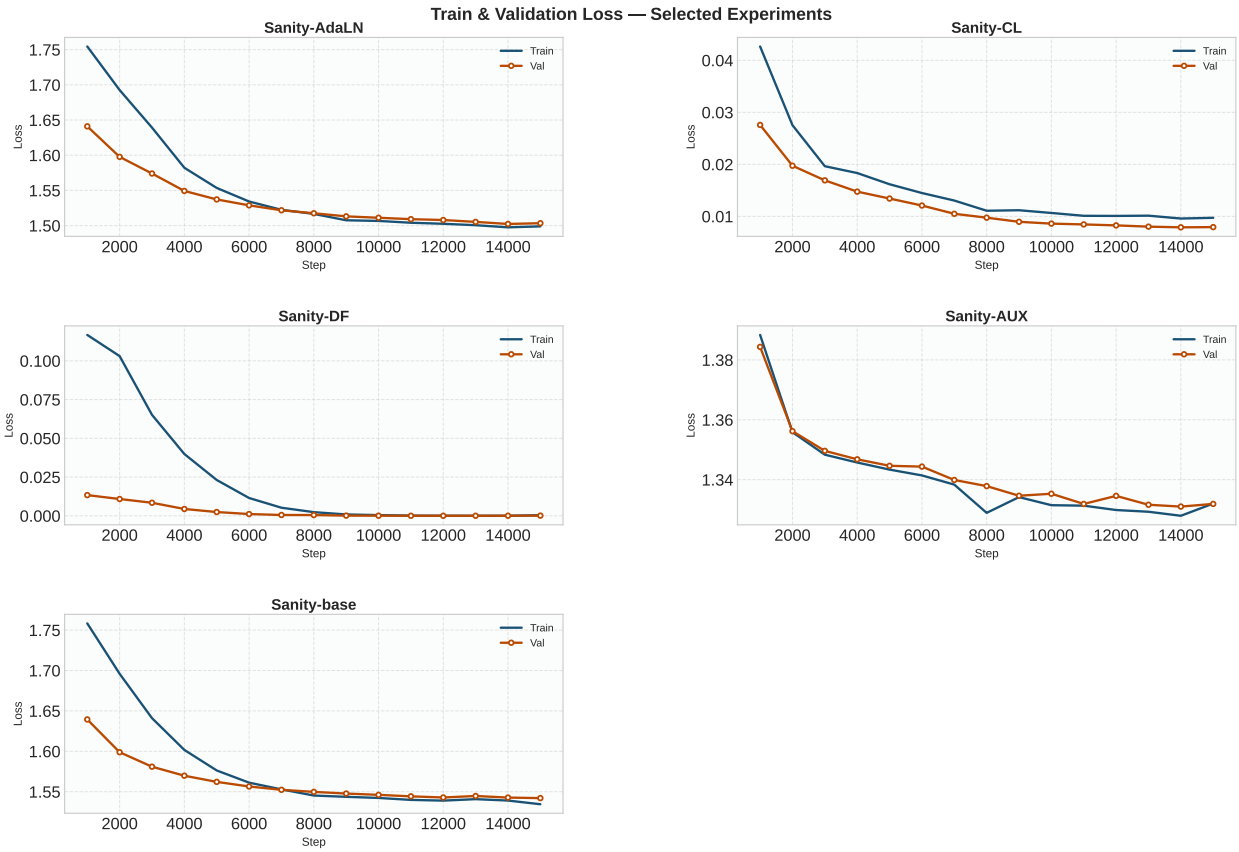


Figure A.3: Training and validation loss for different sanity experiments.

Appendix B. Ablation Studies

B.1 Training and validation loss results for Ablation Studies

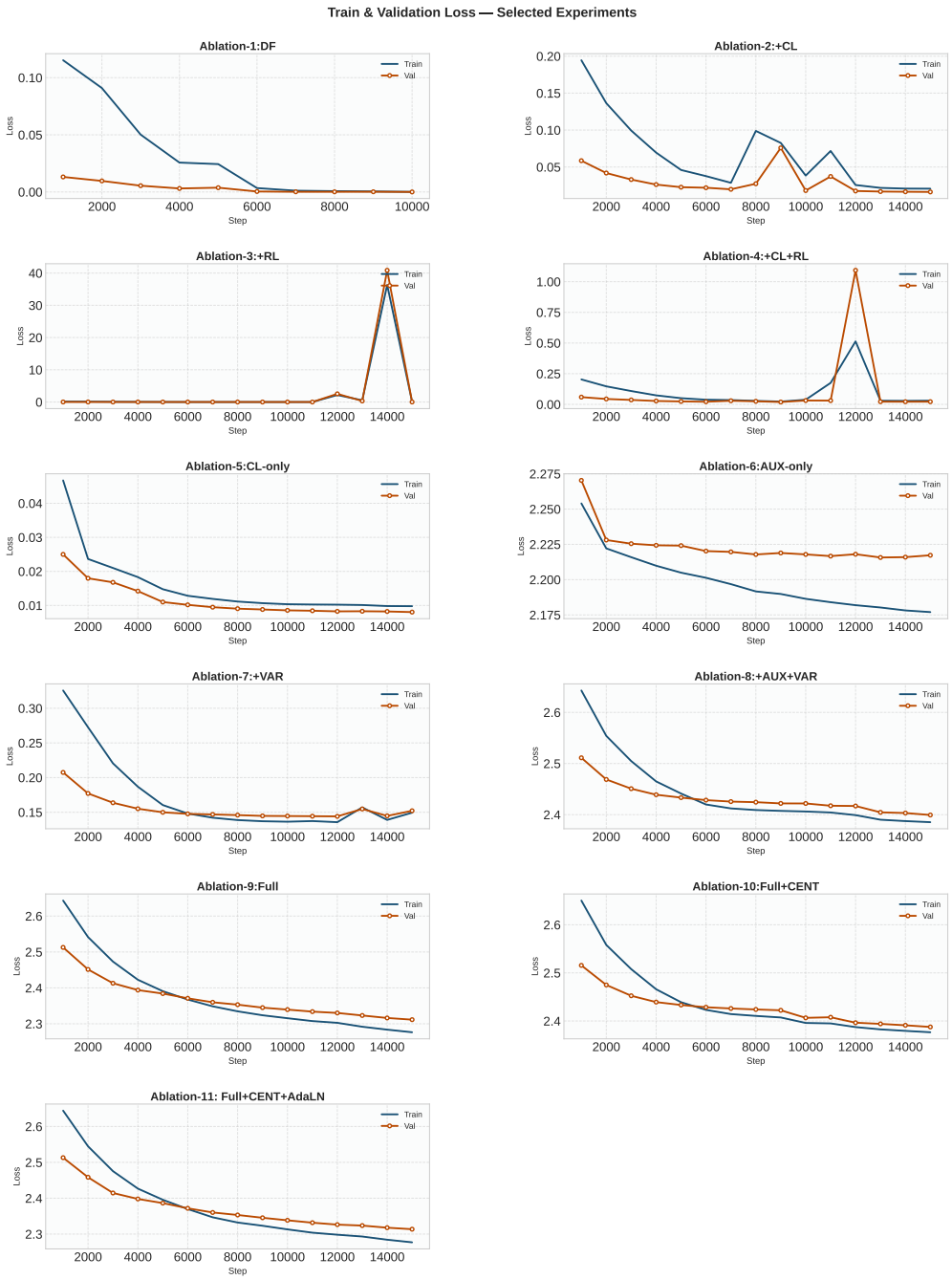


Figure B.1: Training and validation loss for different ablation experiments.

Appendix C. Embedding Statistics Formulas

Metric	Formula	Description
Mean Norm	$\frac{1}{N}\sum_{i=1}^{N}\|e_i\|_2$	$e_i \in \mathbb{R}^D$ embedding vector, N number of embeddings, $\|\cdot\|_2$ is L2 norm over D dimensions
STD Norm	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\|e_i\|_2 - \mu)^2}$	μ is mean norm, e_i embedding vector, N number of embeddings
Per Dimension STD Mean	$\frac{1}{D}\sum_{j=1}^{D}\sigma_j$	D embedding dimension, σ_j std of dimension j across all embeddings
Per Dimension STD Min	$\min_j \sigma_j$	Minimum variance across embedding dimension, $j \in \{1, 2, ..., D\}$
Collapse Indicator	$\left\\|\frac{1}{N}\sum_{i=1}^{N} e_i\right\\|_2$	e_i embeddings, N samples, measures deviation of mean vector from origin
PCA Top-1	$\frac{\lambda_1}{\sum_{k=1}^{D}\lambda_k}$	λ_k PCA eigenvalues, D total components, variance share of first component
PCA Top-n	$\frac{\sum_{k=1}^{n}\lambda_k}{\sum_{k=1}^{D}\lambda_k}$	First n PCA eigenvalues relative to total variance
Participation Ratio	$\frac{(\sum_{k=1}^{D}\lambda_k)^2}{\sum_{k=1}^{D}\lambda_k^2}$	λ_k PCA eigenvalues measuring effective dimensionality
Cosine Mean	$\mu_s = \frac{1}{M}\sum_{(i,j)} s_{ij}$	$s_{ij} = \hat{e}_i^\top \hat{e}_j$, M sampled pairs, $\hat{e}_i$ L2-normalized embeddings
Silhouette Score	$\frac{1}{N}\sum_{i=1}^{N}\frac{b_i - a_i}{\max(a_i, b_i)}$	a_i intra-cluster distance, b_i nearest-cluster distance, N samples

where STD is the standard deviation.