Sparse Autoencoders for Interpretability of InstaNovo

Mohammed Esameldin Adam Nagi
African Institute for Mathematical Sciences (AIMS)

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

June 25, 2026

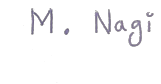
Submitted in partial fulfillment of a structured masters degree at AIMS South Africa

Abstract

Transformer models for *de novo* peptide sequencing reconstruct peptide sequences from mass spectra with high accuracy but offer no insight into which spectral features drive their predictions. This work investigates whether sparse autoencoders can uncover chemically meaningful representations inside InstaNovo, a *de novo* peptide sequencer. We train sparse autoencoders at four depths of the model's encoder and pair them with a fragment-ion annotation pipeline that labels each spectral peak with chemical concepts derived from first principles, allowing us to ask which learned features correspond to which chemical events. The trained autoencoders reconstruct the encoder's internal representations with greater than 92% fidelity at every layer while preserving more than 99.5% of the model's sequencing loss. A substantial fraction of features associate significantly with specific chemical concepts, and these associations are organised by depth: physical spectral properties are represented most sharply in early layers, while chemically specific properties such as cleavage specificity and modification localisation emerge in the deeper layers. Causal ablation experiments establish the degree to which these associations reflect genuine use by the model rather than coincidental correlation. The findings show that sparse autoencoders are a viable tool for mechanistic interpretability of transformer models used in proteomics, while exposing a structural limitation in how correlation-based feature ranking interacts with the chemical concepts that co-occur throughout mass spectrometry data.

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.



Mohammed Esameldin Adam Nagi, June 25, 2026

Contents

1. Introduction

1.1 De Novo Peptide Sequencing and the Interpretability Gap

Mass spectrometry is the dominant technology for identifying the proteins present in a biological sample. A typical bottom-up proteomics experiment isolates short peptide fragments of those proteins, fragments each peptide further inside the mass spectrometer, and records the resulting tandem mass spectra: lists of mass-to-charge ratios paired with intensities corresponding to the charged fragment ions produced. The interpretation step that turns those spectra back into peptide sequences is called peptide identification, and it is the bottleneck around which downstream proteomics analysis revolves (Aebersold and Mann, 2016).

The conventional approach is database search: each observed spectrum is compared against theoretical spectra computed from a reference proteome, and the best match is reported (Sadygov et al., 2004). This works well when the relevant peptides are present in the database, but fails by construction whenever they are not. Genomic mutations, alternative splicing, post-translational modifications outside the modelled set, novel microorganisms with no sequenced reference, antibody sequencing tasks, and metaproteomics are all settings where the database is incomplete and a database-search workflow returns nothing useful. *De novo* peptide sequencing addresses this gap: given only the spectrum and a precursor mass, predict the amino-acid sequence directly, without any prior knowledge of the proteome (Tran et al., 2017).

Recent progress on *de novo* peptide sequencing has been driven by transformer architectures (Yilmaz et al., 2024; Bittremieux et al., 2026). InstaNovo, the model we focus on in this work, is one such model (Eloff et al., 2025). It treats the spectrum as a sequence of peak tokens, encodes them through nine self-attention layers, and decodes the peptide sequence autoregressively, conditioned on the encoder representations and the precursor information. On the standard nine-species benchmark (Wen and Noble, 2024), it produces accuracy competitive with or exceeding the best database-search workflows on novel-peptide tasks. The internal computation that produces these predictions, however, is opaque. Individual neurons in the encoder layers respond to many apparently unrelated stimuli, a phenomenon now well-documented in transformer language models and termed polysemanticity (Elhage et al., 2022).

Without a way to read out what the model is representing, several questions important to a working proteomics researcher cannot be answered: which spectral features drive a specific prediction, whether the model is using genuine fragmentation evidence or a statistical shortcut, and where its predictions are likely to fail. Understanding what the encoder has learned is also crucial for principled model improvement: once a miscalibrated or missing representation is identified, the model can be redesigned, fine-tuned, or its training data curated to address it directly (Räuker et al., 2023).

1.2 Sparse Autoencoders as an Interpretability Tool

Sparse autoencoders (SAEs) have become the leading approach for decomposing transformer representations into human-interpretable units. The construction is simple: a wide overcomplete autoencoder is trained to reconstruct internal activations through a sparse code, with the hope that individual code units correspond to coherent concepts that the model uses internally. Bricken et al. (2023) demonstrated that this works in practice on a one-layer transformer and recovered interpretable features such as Arabic-script detectors and DNA-string detectors. Templeton et al. (2024) scaled the method to a production-grade model and reported features corresponding to abstract concepts including code vulnerabilities and unsafe outputs. Most relevant to this work, Simon and Zou (2025) applied SAEs to the

protein language model ESM-2 and recovered features that aligned closely with established structural and functional protein annotations, showing that the technique generalises beyond natural language to biological sequence models.

Whether the same is true for a model trained on physical spectra rather than sequence is the question this research project sets out to answer. The expected feature set is different from the protein-language-model case: instead of evolutionary motifs, we should expect detectors for fragmentation patterns, ion series, charge states, neutral losses, and the specific mass shifts associated with post-translational modifications. Whether these concepts are represented as separable directions in InstaNovo's residual stream, and whether SAE training in fact recovers them, is unknown *a priori*.

1.3 Contributions

This thesis makes four contributions.

A multi-layer SAE pipeline for InstaNovo, trained and evaluated at encoder layers 2, 4, 6, and 8 on 639,286 spectra from the nine-species benchmark. The trained SAEs explain between 92.0% and 98.4% of activation variance and preserve more than 99.5% of the model's sequencing loss at every layer, confirming that the sparse representations are faithful proxies for the original encoder outputs.

A per-peak fragment-ion annotation pipeline that assigns each spectral peak to a structured chemical concept set derived entirely from first principles. Every peak is matched against the theoretical fragment ions of its labelled peptide. The resulting labels encode 50 binary concepts across 14 families, including ion type, fragmentation position, neutral-loss family, cleavage specificity, residue coverage, and post-translational modification localisation at the fragment level. This is, to our knowledge, the first per-peak annotation pipeline coupled to an SAE for a mass spectrometry model, and an enabling tool for any future per-token interpretability analysis on InstaNovo.

An eight-phase evaluation suite that tests reconstruction quality, feature structure and sparsity, feature-concept association under Benjamini–Hochberg (BH) correction across the full contingency table, and causal feature ablation with firing-rate-matched random controls and a null resampling check on the strongest associations.

A causal and correlational characterisation of InstaNovo's encoder representations, including two findings that a correlational analysis alone would not produce. First, the strongest causal signal in the evaluation belongs to y-ion features at layer 2, despite having only modest correlational scores. Second, the most strongly correlated feature in the evaluation, the layer-2 immonium ion detector, produces no detectable causal effect when ablated, establishing that correlational association and functional necessity are distinct properties that must be evaluated separately.

1.4 Outline

Chapter 2 reviews the necessary background on tandem mass spectrometry, *de novo* peptide sequencing, the mechanistic interpretability paradigm, and the sparse autoencoder methods adopted here. Chapter 3 describes the activation-extraction, the SAE architecture and training procedure, the per-peak annotation, and the evaluation framework. Chapter 4 presents the empirical results across all four encoder layers, from reconstruction quality and feature structure through concept associations and causal ablation. Chapter 5 interprets the results, situates the work within the broader literature, discusses the limitations of the thesis, and outlines directions for future work.

2. Background and Related Work

This chapter provides an overview of the research and technical context required for the rest of the thesis. It begins with a brief overview of tandem mass spectrometry and the problem that *de novo* peptide sequencing tries to solve, then describes the InstaNovo architecture and its position within the field. The second half of the chapter traces the interpretability tools that motivate this work: the mechanistic interpretability framework, sparse autoencoders and the design choices the field has developed, and the prior applications of SAEs to biological sequence models from which we take direct methodological inspiration.

2.1 Tandem Mass Spectrometry and Peptide Identification

The dominant strategy for characterising the proteins present in a biological sample is bottom-up proteomics (Aebersold and Mann, 2016). A protein mixture is first digested enzymatically, producing a population of short peptide fragments. The peptides are separated by liquid chromatography and then analysed in two sequential mass spectrometry steps. The first step (MS1) measures the mass-to-charge ratio of intact peptide ions. The second step (MS2) isolates each peptide of interest, fragments it further inside the instrument through collision-induced dissociation, and records the masses of the resulting fragment ions. This thesis concerns data-dependent acquisition (DDA), in which precursor ions are selected and fragmented individually based on MS1 signal intensity. The MS2 output, the tandem mass spectrum, carries the sequence information and forms the input to computational peptide identification.

When a peptide backbone is fragmented, the cleavage occurs preferentially at peptide bonds. Fragment ions that retain the N-terminus of the peptide are labelled b-ions; those retaining the C-terminus are labelled y-ions (Steen and Mann, 2004). For a peptide of length n, backbone cleavage at position k produces a b_k ion covering residues 1 through k and a y_{n-k} ion covering residues $k+1$ through n. A complete, noise-free spectrum would contain a b-ion and a y-ion for every cleavage position, forming two complementary ladders that together can reconstruct the original sequence. In practice, spectra are incomplete and noisy: not every position produces a detectable ion, fragment ions can carry multiple charges, neutral losses shift some ions away from their nominal masses, and internal fragments (arising from two backbone cleavages) and immonium ions add additional peaks that do not belong to either ladder. Co-fragmenting peptides and chemical contaminants contribute further peaks with no relationship to the precursor sequence. The result is a variable-length, partially annotated, inherently noisy bag of $(m/z$, intensity$)$ pairs; a sparse fingerprint of the underlying sequence rather than a direct readout of it.

This physical picture has direct consequences for the modelling task. The spectrum has no fixed length, no inherent ordering of peaks, and no guarantee that the information needed to identify the peptide is present. The relationship between spectrum and sequence is complex and non-invertible in general: different sequences can produce similar spectra, and the same sequence may produce quite different spectra depending on instrument settings, charge state, and fragmentation method. Any model that translates spectra into sequences must implicitly learn the physics of fragmentation and the combinatorics of peptide sequences.

2.2 Database Search and Its Limits

The conventional approach to peptide identification is database search: each observed spectrum is compared against theoretical spectra computed from every candidate peptide in a reference proteome, and the best-scoring match is reported as the identification (Sadygov et al., 2004). Confidence in the identification is calibrated through a target-decoy framework in which the search is simultaneously run against a reversed or shuffled decoy database; the false discovery rate at a given score threshold is estimated from the proportion of decoy hits among all accepted identifications at that threshold (Elias and Gygi, 2007). This framework is statistically principled and computationally mature, and database search remains the dominant identification strategy for most proteomics workflows.

Its central limitation is that it can only identify peptides whose sequences are present in the database. This restriction is consequential in a wide range of practically important settings. Peptides arising from genomic mutations, alternative splicing events, single-nucleotide polymorphisms, or carrying post-translational modifications are missed unless the specific variant is present in the reference. Novel organisms, metaproteomics samples, and engineered sequences such as nanobodies or therapeutic antibodies have no established reference and return no database hits. Post-translational modifications outside the modelled set inflate the search space exponentially, causing the false discovery rate to rise and the number of identifications to fall (Muth and Renard, 2018). Empirically, roughly half of all acquired MS2 spectra remain unidentified in typical shotgun proteomics experiments (Griss et al., 2016), representing a large body of biological signal that database search cannot access.

2.3 *De Novo* Peptide Sequencing

De novo peptide sequencing addresses this gap by predicting the amino-acid sequence directly from the spectrum and the precursor mass, without relying on any reference database. The task is well-posed: the precursor mass constrains the total composition of the peptide, and the fragment-ion ladders in the spectrum constrain the sequence consistent with that composition. The challenge is that the solution space is exponentially large and the observations are noisy and incomplete.

The field was dominated for many years by algorithms based on spectrum graph representations, in which nodes correspond to observed masses and edges correspond to mass differences consistent with amino-acid residues (Ma et al., 2003; Frank and Pevzner, 2005). Dynamic *de novo* sequencing reduces to finding the highest-scoring path through this graph. These methods made the fragmentation physics explicit and interpretable, but they were sensitive to noise, struggled with multiply-charged ions and neutral losses, and did not generalise well across instrument types.

The introduction of recurrent neural network approaches to *de novo* sequencing (Tran et al., 2017) replaced hand-crafted graph traversal with learned sequence generation, substantially improving accuracy on standard benchmarks. Subsequent work extended this to data-independent acquisition (Tran et al., 2019) and demonstrated that the performance gap between *de novo* and database search could be substantially closed with sufficient training data and model capacity. The most recent generation of *de novo* sequencing models is transformer-based, reframing the problem as a sequence-to-sequence translation task in which the spectrum tokens are encoded by a transformer encoder and the peptide sequence is decoded autoregressively (Yilmaz et al., 2024; Bittremieux et al., 2026). These models achieve accuracy competitive with database search on standard benchmarks while generalising to novel peptides that database search cannot identify.

InstaNovo (Eloff et al., 2025), the model analysed in this thesis, is a representative of this transformer

generation. Its encoder consists of nine self-attention layers, each with 16 attention heads and a hidden dimension of 768, producing a 768-dimensional residual-stream representation per input token. Each spectral peak is embedded using multi-scale sinusoidal encodings of its m/z value (Voronov et al., 2022), a representation that preserves the high-precision mass information needed for fragment matching, with a separate scalar embedding for its log-intensity. A learned latent token is prepended to the peak sequence at position zero; it accumulates spectral context across the encoder layers and serves as a summary representation of the full spectrum. The nine-layer decoder performs causal autoregressive decoding, cross-attending to the concatenated encoder output, the latent spectrum embedding, and a precursor encoding that conditions each prediction on the precursor mass and charge. A knapsack beam search ensures every predicted sequence is mass-consistent with the precursor. On the nine-species benchmark, InstaNovo achieves accuracy competitive with or exceeding database search on novel-peptide tasks; yet the internal computation that produces these predictions is entirely opaque. Understanding how the encoder represents the spectral evidence that drives sequence generation is the question this thesis addresses.

2.4 Mechanistic Interpretability

2.4.1 What mechanistic interpretability asks

Most interpretability methods aim to explain a model's outputs: they ask which parts of the input were most influential for a given prediction. Integrated gradients (Sundararajan et al., 2017) compute input-attribution scores; linear probes (Alain and Bengio, 2016) check whether a target concept is linearly decodable from a layer's activations; attention visualisations (Vig, 2019) highlight which tokens a layer's attention heads weight most heavily. These methods are useful for understanding individual predictions but they do not answer the more fundamental question of what the model has learned to represent: what internal vocabulary it uses to encode information as it flows from input to output.

The only prior work that directly attempts to interpret a *de novo* sequencing transformer is the piNovo model (Wang et al., 2024), which introduces an attention-based attribution module that links each predicted residue to the peaks that most influenced it. Attention weights, however, are not causal attributions; they reflect the routing decisions of the attention mechanism rather than the information content of the routed signal (Jain and Wallace, 2019; Wiegreffe and Pinter, 2019) and the method cannot characterise what the encoder has learned to represent independently of the prediction task.

Mechanistic interpretability has a different aspiration: to reverse-engineer the computational algorithms implemented inside the model (Olah et al., 2020). Under this paradigm, the basic unit of analysis is not the neuron but the *feature*; a direction in activation space that corresponds to a coherent concept the model uses internally. Circuits are subgraphs of the model in which features are connected by weights that implement specific algorithms, such as copying a token, detecting a grammatical pattern, or identifying a fragment ion ladder. The goal is an account of model behaviour not in terms of inputs and outputs but in terms of the intermediate computations the model performs.

2.4.2 Polysemanticity and the superposition hypothesis

A fundamental obstacle to mechanistic interpretability is *polysemanticity*: the observation that individual neurons in trained networks respond to multiple, apparently unrelated stimuli (Elhage et al., 2022). A neuron that fires on Arabic script, base-64 strings, and Python keywords simultaneously does not correspond to any human-understandable concept, and there is no principled way to assign it a single meaning.

Elhage et al. (2022) explain polysemanticity through the *superposition hypothesis*: when a network needs to represent more features than it has dimensions, it can encode features as non-orthogonal directions in activation space, accepting some interference between features in exchange for additional representational capacity. A model with d neurons can represent far more than d features in this way, at the cost of making individual neurons polysemantic. The consequence is that the right object to study is not the neuron basis but the set of directions the model actually uses.

2.4.3 The linear representation hypothesis

The superposition argument assumes that features are encoded as *linear* directions in activation space; that adding two feature-direction vectors superimposes the corresponding concepts additively. This assumption, formalised as the *linear representation hypothesis* (Park et al., 2023), is supported by extensive empirical evidence from language models: arithmetic can be performed by vector addition, semantic relationships are encoded as consistent vector offsets, and probes trained on individual directions generalise across contexts. If the hypothesis holds in the mass spectrometry domain, then each chemical concept that InstaNovo represents internally should correspond to a specific direction and the task of interpretability is to find those directions. The residual stream is the natural target for this search as it is the one representation that persists across the full depth of the network, with each layer reading from and writing back to it, making it the best approximation of a global internal vocabulary that the transformer architecture provides (Elhage et al., 2022).

Sparse autoencoders are designed to find them empirically, without supervision and without any prior knowledge of which directions are meaningful.

2.5 Sparse Autoencoders for Feature Discovery

2.5.1 From sparse coding to neural dictionaries

The idea of representing signals as sparse combinations of basis elements is older than deep learning. Olshausen and Field (1996) showed that training a dictionary on natural image patches to minimise reconstruction error subject to a sparsity constraint produces Gabor-like filters resembling the receptive fields of simple cells in primary visual cortex; an early demonstration that sparsity, as an inductive bias, promotes interpretable feature discovery. Dictionary learning generalises this idea: given a data matrix, find an overcomplete basis (a dictionary) such that each data point can be represented as a sparse combination of dictionary elements. Sparse autoencoders are a neural implementation of this principle, where both the dictionary and the sparse codes are learned end-to-end from transformer activations.

2.5.2 Architecture

Let $x \in \mathbb{R}^d$ denote a transformer activation at the layer of interest. A sparse autoencoder consists of an encoder $f : \mathbb{R}^d \to \mathbb{R}^F$ with weight matrix $W_{\text{enc}} \in \mathbb{R}^{d \times F}$ and bias $b_{\text{enc}} \in \mathbb{R}^F$, and a decoder $g : \mathbb{R}^F \to \mathbb{R}^d$ with weight matrix $W_{\text{dec}} \in \mathbb{R}^{F \times d}$ and bias $b_{\text{dec}} \in \mathbb{R}^d$, with $F \gg d$:

$$z = \sigma(W_{\text{enc}}(x - b_{\text{dec}}) + b_{\text{enc}}), \tag{2.5.1}$$

$$\hat{x} = W_{\text{dec}}\, z + b_{\text{dec}}. \tag{2.5.2}$$

The decoder bias b_{dec} is subtracted before encoding and added after decoding, centring the activation distribution at the encoder input. The sparsity nonlinearity σ controls which features are active for a given token. The autoencoder is trained to minimise reconstruction error $\|x - \hat{x}\|_2^2$ subject to the

sparsity constraint imposed by σ. If training succeeds, nonzero entries of z correspond to conceptually distinct features that the model uses to represent x, and the corresponding decoder columns in W_{dec} are the directions in which those features are written into the residual stream.

SAE features are substantially more interpretable than individual neurons. Huben et al. (2024) demonstrated this on Pythia-70M, recovering monosemantic features and using them to localise the circuits responsible for specific model behaviours more precisely than neuron-level analysis permitted. Bricken et al. (2023) showed the same on a one-layer MLP, recovering detectors for specific scripts, DNA strings, and numerical patterns. Templeton et al. (2024) scaled the approach to a production-grade model and found features corresponding to abstract concepts. The consistent finding across settings is that SAE features express concepts that individual neurons suppress through superposition, and that the more overcomplete the dictionary (F/d large), the richer the recovered feature set.

2.5.3 Sparsity enforcement

The choice of sparsity nonlinearity σ is the central design decision in SAE training and has evolved substantially as the field has matured. Three broad approaches have been explored, each with distinct properties.

The earliest SAE work imposed an L1 penalty on the feature activations z, encouraging most entries to be exactly zero while allowing the rest to take any non-negative value (Bricken et al., 2023). L1 penalisation is differentiable and integrates cleanly into standard gradient descent, but it introduces a systematic *shrinkage bias*: the gradient of the penalty term pulls active feature magnitudes toward zero, so the magnitudes of activated features are systematically underestimated relative to their true contribution to the reconstruction. This matters for interpretability because feature magnitudes are used to rank features by importance and to compare activation levels across tokens, both operations become unreliable when magnitudes are compressed. The shrinkage also interacts poorly with causal ablation experiments; if active feature magnitudes are biased downward, the apparent effect of zeroing them is also reduced.

Hard top-k selection avoids shrinkage entirely by zeroing all but the k largest pre-activations and applying no penalty to the retained ones (Makhzani and Frey, 2013; Gao et al., 2024). This makes sparsity an explicit hyperparameter rather than an implicit trade-off against reconstruction quality, with exactly k features active per token by construction. The magnitudes of active features are unconstrained by the sparsity mechanism and so reflect the true activation level. Hard selection also places the SAE on a cleaner sparsity-fidelity Pareto frontier than L1 penalisation, producing sparser feature activations with less magnitude distortion at equal reconstruction quality.

The limitation of standard per-token top-k is a positive feedback loop. Features that are poorly aligned with the data at initialisation rarely win the per-token competition, never receive gradient, and progressively move further from useful directions. A substantial fraction of dictionary entries can become permanently dormant; consuming capacity without contributing to reconstruction or interpretability.

Bussmann et al. (2024) break the dead-feature feedback loop by shifting the competition from per-token to per-batch: the top $k \cdot B$ activations are selected jointly across a batch of B tokens rather than the top k per token independently. Average sparsity is preserved, but a feature now needs only to be competitive on some tokens in the batch rather than on every individual token, substantially reducing the probability of permanent dormancy. Empirically, BatchTopK reduces dead feature rates without sacrificing reconstruction quality relative to standard top-k.

A complementary approach, useful at inference time and for interventional analyses that require a

batch-independent activation rule, replaces the batch-level competition with a learned per-feature threshold (Rajamanoharan et al., 2024). Each feature has a threshold θ_f, and the activation function is a step: zero below the threshold, the raw pre-activation above it. The thresholds are learned during training via a straight-through estimator and can be scaled uniformly at inference time to slide along the sparsity-fidelity frontier without retraining; a useful property when the interpretability-fidelity trade-off at deployment differs from the training setting.

2.5.4 Dead feature recovery

Even with BatchTopK, some fraction of dictionary entries may never activate on any training token, wasting capacity and reducing interpretability coverage. Three responses to this problem have been proposed.

Regularisation adds a penalty term that encourages dormant features to activate. The fundamental difficulty is that any penalty strong enough to resurrect a genuinely dead feature competes with the reconstruction objective and degrades the quality of live features (Bricken et al., 2023).

Periodic re-initialisation identifies dead features at intervals and resets their encoder and decoder weights to directions that explain current residual error. This is more aggressive than regularisation and does not compromise the reconstruction objective, but it requires a monitoring loop and introduces a discontinuity into training (Bricken et al., 2023).

Auxiliary reconstruction takes a softer approach by constructing a separate reconstruction task that involves only dead features, insulating the recovery gradient from the main reconstruction gradient (Gao et al., 2024). After the main reconstruction $\hat{x}$ is computed from the top-k active features, the residual error $e = x - \hat{x}$ is reconstructed using only the currently dormant features. A small coefficient α weights this auxiliary loss so that dead features receive gradient on every step without affecting the gradient on live features. The two objectives are orthogonal by construction: the auxiliary head operates on features that the main reconstruction ignores. This design avoids regularisation tension and requires no monitoring loop, making it an effective and widely adopted standard in the field.

2.5.5 Evaluating SAE quality

Assessing the quality of a trained SAE requires measuring both how well it reconstructs the original activations and how well it serves as a proxy for the model whose activations it encodes. This evaluation typically relies upon a core set of complementary metrics.

The *fraction of variance explained* (FVE) is the primary reconstruction metric:

$$\mathrm{FVE} = 1 - \frac{\mathbb{E}[\|x - \hat{x}\|_2^2]}{\mathbb{E}[\|x - \bar{x}\|_2^2]}, \tag{2.5.3}$$

where $\bar{x}$ is the mean activation. FVE of 1.0 means the reconstruction is perfect; 0.0 means the SAE explains no more variance than the mean. Values above roughly 0.9 are generally considered sufficient for the SAE to be a faithful proxy, though the threshold depends on how sensitive the downstream task is to the reconstruction residual (Gao et al., 2024; Templeton et al., 2024).

The *L0 norm* is the mean number of active features per token under the calibrated activation threshold. For hard top-k SAEs this is fixed at k during training; for threshold-based activation functions it is measured after calibration. L0 is the primary measure of sparsity and controls the interpretability-fidelity trade-off: lower L0 produces more monosemantic features but leaves more activation variance in the residual.

The *dead feature rate* is the fraction of dictionary entries that never activate on any evaluation token. A high dead feature rate indicates wasted capacity and motivates the auxiliary reconstruction approaches described above. It is also a diagnostic for the activation function: per-token top-k typically produces higher dead rates than batch-level selection at equal L0.

Loss recovered measures how much of the downstream task performance survives when the SAE reconstruction is patched into the model's forward pass in place of the original activations:

$$\mathrm{LR} = 1 - \frac{\mathrm{CE}_{\mathrm{sae}} - \mathrm{CE}_{\mathrm{clean}}}{\mathrm{CE}_{\mathrm{zero}} - \mathrm{CE}_{\mathrm{clean}}}, \tag{2.5.4}$$

where CE is the cross-entropy of the model's predictions, $\mathrm{CE}_{\mathrm{clean}}$ is the baseline without any intervention, and $\mathrm{CE}_{\mathrm{zero}}$ is the cross-entropy when the target layer's output is set to zero (a destructive baseline). A loss recovered of 1.0 means the SAE preserves the model's behaviour exactly; 0.0 means the SAE reconstruction is no better than zeroing the layer. This metric is more stringent than FVE because it tests whether the *task-relevant* variance is preserved, not just the total variance.

The *Gini coefficient* of the feature firing-rate distribution measures inequality in feature utilisation: a Gini of 0 means all features fire equally often; a Gini near 1 means a small number of features account for nearly all activations. High Gini values are expected under top-k sparsity; the distribution is inherently unequal because concepts differ in their prevalence in the data but pathologically high values may indicate that the dictionary has collapsed into a small effective vocabulary.

2.6 Sparse Autoencoders for Biological Sequence Models

Interpretability for protein language models initially relied on analysing their attention patterns. Vig et al. (2021) showed that attention heads in protein transformers recover structural contacts, binding sites, and other biophysical properties directly from sequence. As with attention analysis in natural language, this gives a correlational view of what the model attends to rather than a mechanistic account of the representations it builds.

One of the first applications of SAEs to a biological sequence model was InterPLM (Simon and Zou, 2025), which trained SAEs on the residue embeddings of ESM-2 (Lin et al., 2023) at multiple model scales. ESM-2 is a protein language model trained by masked-token prediction on hundreds of millions of protein sequences; its representations encode rich structural and evolutionary information without any explicit supervision. Simon and Zou (2025) found that SAE features recover thousands of biologically interpretable concepts per layer, including secondary structure motifs, Pfam domains, binding sites, disordered regions, and post-translational modification sites, while individual neurons show far less conceptual alignment, providing strong evidence that ESM-2 stores these concepts in superposition. The same finding was confirmed independently and concurrently by Adams et al. (2025) (InterProt), who trained SAEs on the larger ESM-2-650M model and additionally characterised *family-specific* features: dictionary entries that activate strongly only within a specific protein family, such as the central helix of beta-lactamases, rather than across all proteins sharing a structural motif. This specificity-versus-generality axis in feature structure has a direct parallel in the present work, where some features fire broadly on a common ion type and others fire selectively on rare fragmentation events.

Both InterPLM and InterProt are trained on encoder-only protein language models, where the full representational burden falls on a single stack of transformer layers. InstaNovo differs in being an encoder-decoder architecture, so the choice of where to train SAEs requires justification. The encoder processes the raw spectrum into a contextualised representation before any sequence generation occurs,

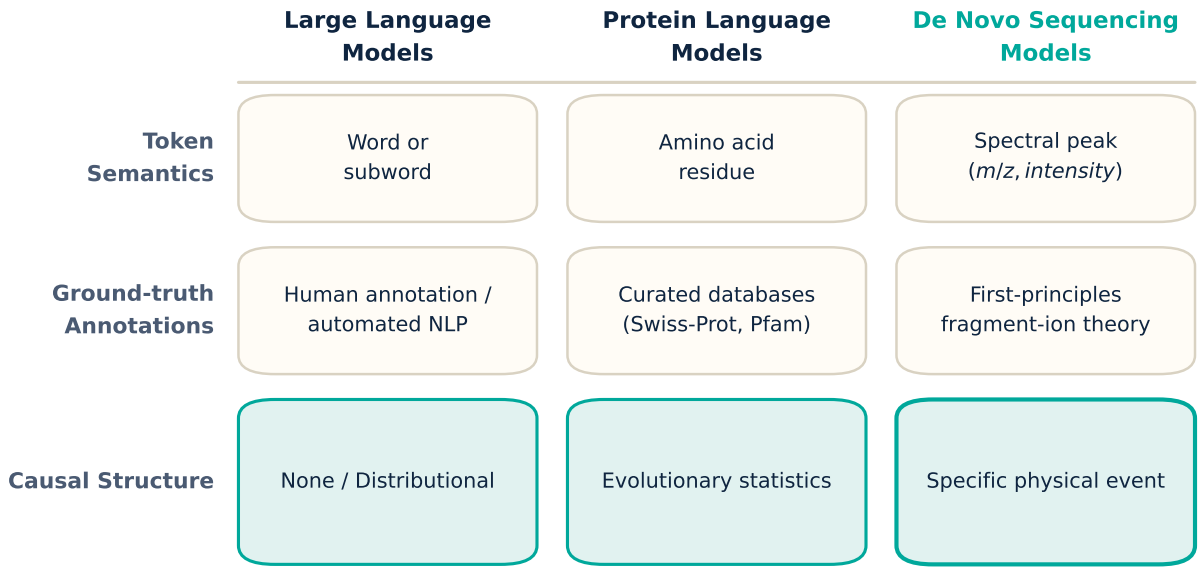


Figure 2.1: Comparison of SAE application contexts across large language models, protein language models, and *de novo* sequencing models. The three settings differ in the semantic grounding of their tokens and in the availability of ground-truth concept annotations. The mass-spectrometry setting is unique in that per-token annotations can be derived entirely from first principles using fragment-ion theory, with no dependence on external databases.

making it directly analogous to the models studied in InterPLM and InterProt. The decoder, by contrast, conditions on both the encoder output and its own autoregressive context, conflating input recognition with sequence generation in a way that is difficult to ground in fragment-ion theory. The encoder is therefore the natural target for concept-level interpretability in this work.

Language models, protein language models, and *de novo* sequencing models differ in how tokens are semantically grounded and in what concept annotations are available for evaluation (Figure 2.1). The mass-spectrometry setting is distinctive in that per-token annotations can be derived entirely from first principles using fragment-ion theory, with no dependence on external databases or human curation.

Both InterPLM and InterProt adopt the same evaluation strategy: compare feature activation patterns against per-residue concept labels drawn from biological databases (Swiss-Prot, Pfam, InterPro), and measure alignment using domain-adjusted recall, a metric that credits a feature for firing on at least one concept-positive position in a sequence rather than requiring it to fire on every such position. We adopt an analogous selectivity-based metric suited to the token structure of mass spectra, described in Section 3.5.3.

2.7 From Correlation to Causation in Feature Evaluation

A feature that fires reliably on concept-positive tokens is not necessarily the feature the model uses to represent that concept. Correlational evidence establishes that the feature and the concept are statistically associated in the data, but it does not establish that the association reflects a functional relationship within the model. The feature might be spuriously correlated with the concept because

both co-occur with a third variable; it might be one of many redundant encodings of the concept, such that ablating it has negligible effect because the model compensates through other features; or it might detect a surface-level spectral property (high intensity, low m/z) that happens to co-occur with the concept in the particular dataset used for evaluation.

This gap between correlation and causation is well-recognised in mechanistic interpretability (Meng et al., 2022). The standard response is to combine correlational evidence with *intervention evidence*. If ablating a feature group affects the model's predictions exclusively for concept-positive inputs to a greater degree than would be expected from random ablation, this constitutes evidence of a genuine functional relationship.

The simplest intervention is to zero out the activations of a feature or feature group and measure the change in the model's downstream behaviour (cross-entropy, top-1 accuracy, or KL divergence from the clean prediction distribution). If the change is disproportionately large on concept-positive inputs (selectivity) and larger in magnitude than for a matched set of randomly chosen features (importance), the feature group is providing information to the model that is genuinely used in its predictions. The interpretation must be hedged when the absolute effect is small, because redundant encoding can absorb the intervention: a model that represents the same concept through many overlapping features may be largely unaffected by ablating any particular subset.

Simple ablation is susceptible to the possibility that any high-firing feature group produces a disproportionate effect simply because it accounts for a large fraction of total activation. A separate check on the correlational associations addresses this: for the strongest feature-concept pairs, the observed chi-square value is compared against a null distribution generated by resampling co-occurrence counts under the constraint that both the feature's firing rate and the concept's base rate are held fixed. If the observed statistic exceeds the resampled null, the association cannot be explained by the marginal frequencies alone. A feature-concept association that passes this check is not an artefact of imbalanced class frequencies or the firing-rate distribution of the feature.

Together, these tools form a hierarchy of evidence; the fixed-margin resampling check validates the correlational association and direct ablation establishes necessity. The evaluation framework in Chapter 3 applies them in that order, and the results in Chapter 4 are interpreted with explicit attention to which tier of evidence supports each claim.

3. Methodology

This chapter describes the methodology used to verify whether InstaNovo's encoder organises its internal representations according to chemically meaningful structure. We train sparse autoencoders on the encoder's residual-stream activations, annotate every token with a rich set of fragment-ion concepts derived from first principles, and then ask which learned dictionary elements correspond to which chemical events. Figure 3.1 shows how the four components of the pipeline are connected; the remainder of this chapter describes each in turn.

3.1 Experimental Setup and Layer Selection

InstaNovo's nine-layer encoder (Section 2.3) presents a natural target for multi-depth representation analysis. Throughout this work, encoder layers are identified by their zero-based index in the model's layer stack: layer 0 is the first block applied to the peak embeddings, and layer 8 is the final block, whose output the decoder cross-attends. We train SAEs on four layers spanning the encoder's depth. Layers 0 and 1 are excluded because their outputs remain close to the input peak embeddings and carry little cross-peak context. We retain the final layer 8: as the representation the decoder consumes directly, it is the most task-proximal point in the encoder and therefore the most informative about which chemical features the model carries into sequencing. The resulting set $\{2, 4, 6, 8\}$ samples the stack at even intervals, enabling a cross-layer analysis of how chemical representations evolve from shallow spectral processing to deep task-specific encoding.

SAEs as representation probes. Training a sparse dictionary on a layer's activations and then testing whether the dictionary elements correspond to chemically meaningful concepts is a form of representation probing. A key limitation of this approach is that correlational evidence – a feature fires consistently on concept-positive tokens – does not establish that the model uses that feature to represent the concept. This distinction motivates the causal ablation experiments in Section 3.5.4, which intervene on specific features and measure the downstream effect on model behaviour.

3.2 Activation Extraction

We register forward hooks on `model.encoder.layers[`N`]` for each target layer $N \in \{2, 4, 6, 8\}$. The hook captures the layer's output tensor before it is consumed by the subsequent layer. Because InstaNovo uses a standard post-norm `TransformerEncoderLayer` (PyTorch default, `norm_first=False`), the layer output is the post-second-LayerNorm residual-stream hidden state, of shape $[B,\, L + 1,\, d_{\text{model}}]$, where B is the batch size, L is the number of spectral peaks (up to 200), and $d_{\text{model}} = 768$. All four layers are captured in a single forward pass, which eliminates any token-count or ordering discrepancy between layers that would complicate cross-layer comparison (Section 3.5.5).

Each spectrum contributes two classes of token to the activation tensor. Position 0 holds the *latent token*: a learned summary vector that aggregates spectral context for the decoder's cross-attention and that is never aligned to a specific mass peak. Positions 1 through L hold *peak tokens*, one per observed spectral peak, each carrying the model's joint embedding of the peak's m/z value and log-intensity. Because the latent token's role is architecturally distinct from that of peak tokens, it receives separate treatment in the annotation pipeline (Section 3.4).

The activations at each target layer are written to disk as raw floating-point tensors with no further

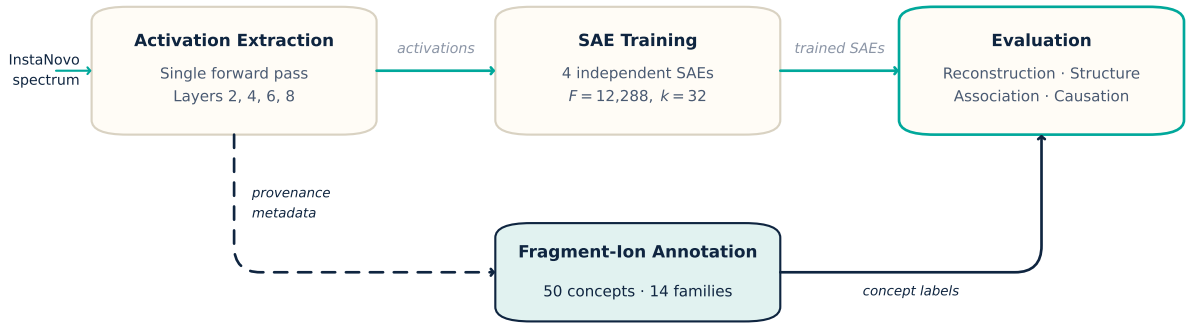


Figure 3.1: Overview of the interpretability pipeline. A single forward pass through InstaNovo captures residual-stream activations at four encoder depths simultaneously. Each layer's activations are used to train an independent SAE; the annotation pipeline runs in parallel on the provenance metadata to produce per-token concept labels. Evaluation joins all four outputs.

rescaling; the BatchTopK sparsity scheme used during SAE training (Section 3.3.2) is robust to the moderate norm variation present in post-LayerNorm activations and requires no explicit standardisation.

Every stored token is accompanied by a (*spectrum index*, *peak index*) identifier that ties it back to a specific peak in a specific spectrum. This provenance record is the thread that connects the SAE's feature activations to the fragment-ion annotations computed independently by the annotation pipeline. Without it, a feature's firing pattern could not be interpreted in terms of chemical events. Alongside the provenance indices, the chunk metadata stores the ProForma peptide sequence, precursor m/z, precursor charge, and per-peak m/z and intensity, so that the annotation pipeline can run without re-reading the original dataset.

The extraction was applied to the full nine-species benchmark (Wen and Noble, 2024), comprising 639,286 spectra and producing approximately 67.5 million encoder tokens per layer.

3.3 Sparse Autoencoder Architecture and Training

3.3.1 Architecture

Let $x \in \mathbb{R}^{d_{\text{model}}}$ denote a post-LayerNorm encoder activation at the layer of interest. The SAE follows the architecture described in Section 2.5.2, with encoder and decoder defined by equations (2.5.1) and (2.5.2). We use an expansion factor of 16, giving $F = 12{,}288$ features per layer. This capacity is chosen to be wide enough to represent the combinatorial space of ion types, positions, charges, and modifications present in a typical tryptic digest, while remaining tractable to train and evaluate on the nine-species benchmark.

3.3.2 Sparsity: BatchTopK

We enforce sparsity with BatchTopK (Section 2.5.3), selecting the top $k \cdot B$ pre-activations jointly across each batch of B tokens rather than the top k independently per token. The training sparsity target is $k = 32$, giving on average 32 active features per token (L0 ≈ 32) from the 12,288-feature dictionary. This value is chosen since a single peak token can simultaneously carry ion-type, fragment-charge, m/z-region, residue-coverage, and modification information, and too tight a sparsity budget would force these co-occurring properties to share features, working directly against the goal of isolating them. Selecting jointly across the batch is what makes the wide dictionary usable here: at $F = 12{,}288$, per-token selection would leave a large fraction of features that never win the competition, never receive gradient, and stay dormant and uninterpretable. We retain the per-batch selection boundary (the smallest kept pre-activation in each batch) during training, since it is also the quantity used to set the inference-time threshold (Section 3.3.4).

3.3.3 Dead Feature Recovery: AuxK

To keep rarely-selected features from going permanently dormant, we add an auxiliary reconstruction term. Our implementation differs from the dead-only formulation of Gao et al. (2024) in one respect; rather than restricting the auxiliary set to features that have not fired within a recent window, we form it from the top k_{aux} pre-activations among all features not selected by the main BatchTopK pass. These are reconstructed against the main-pass residual $e = x - \hat{x}$, which is detached so that the auxiliary objective never perturbs the main reconstruction, and the auxiliary decoder omits the bias b_{dec} (already subtracted by the main reconstruction). The total training loss is

$$\mathcal{L} = \underbrace{\|x - \hat{x}\|_2^2}_{\text{reconstruction}} + \alpha \underbrace{\left\|e - \text{AuxK}(z_{\text{aux}})\, W_{\text{dec}}^{\top}\right\|_2^2}_{\text{auxiliary (unselected features)}}, \tag{3.3.1}$$

where z_{aux} denotes the pre-activations of the top-k_{aux} features not selected by the main pass. Both terms are evaluated as mean squared errors so the norms denote per-coordinate means. The fixed $1/(|\mathcal{B}|\, d_{\text{model}})$ factor is common to both terms. We use $k_{\text{aux}} = 512$ and $\alpha = 1/32$, following Gao et al. (2024), and this configuration keeps the dead-feature rate low in practice. Drawing the auxiliary set from all unselected features rather than the dormant subset is a deliberate simplification: it removes the need to maintain a per-feature firing history during training, at the cost of letting the auxiliary gradient also reach live features that merely lost the per-batch competition on a given token. Because the residual is detached, this broader auxiliary set still cannot degrade the main reconstruction; it only widens the set of features kept receiving gradient.

3.3.4 Inference: JumpReLU

The interventional causal analysis requires an activation rule that does not depend on a batch, so at inference we switch to JumpReLU (Section 2.5.3),

$$\text{JumpReLU}(z, \theta)_f = z_f \cdot \mathbf{1}[z_f > \theta], \tag{3.3.2}$$

using a single global threshold θ shared across all features rather than a learned per-feature threshold. We set θ once, after BatchTopK training has converged, as the mean of the per-batch BatchTopK selection boundaries over 50 calibration batches drawn from the training split. Calibrating to the selection boundary rather than to the mean selected activation is what keeps the inference-time L0 close to the training target $k = 32$: the mean activation sits above the boundary and would make JumpReLU under-fire. Because θ is global, scaling it by a constant slides L0 against reconstruction fidelity without retraining, which we use to generate the threshold-sweep diagnostics in Section 4.2.

3.3.5 Optimisation

The SAE is trained with AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.99$, no weight decay) with a peak learning rate of 2×10^{-4}, a cosine decay schedule to 10% of the peak, and 1,000 linear warm-up steps. The batch size is 8,192 tokens, chosen to make the BatchTopK boundary stable. Gradient norms are clipped to 1.0. Decoder columns are re-projected to unit norm after every optimiser step. Training runs for 3 epochs, though convergence is typically reached within 1.5–2 epochs on the full dataset. A separate SAE is trained for each encoder layer; the training processes are entirely decoupled, with each run executing independently using the activation files for a single layer. Consequently, the four runs share no state beyond the common architecture and hyperparameters.

3.4 Per-Peak Fragment-Ion Annotation

3.4.1 The annotation challenge

For protein language models such as ESM-2, per-residue concept labels can be drawn from curated databases of secondary structure, domain membership, and binding-site annotation (Simon and Zou, 2025). No analogous database exists for InstaNovo's encoder. The model's input tokens correspond to specific mass peaks in a specific mass spectrum, and whether a given peak belongs to a fragment-ion series depends on the physics of gas-phase fragmentation (Section 2.1) and must be computed from first principles for every token in every spectrum.

A further resolution question motivates per-peak rather than per-spectrum annotation. A spectrum-level label (e.g. "this spectrum comes from an oxidised peptide") cannot distinguish a feature that fires on the specific ion carrying the oxidation from one that fires on any peak whenever the spectrum contains an oxidised residue. Token-level labels make this distinction testable: the contrast between `ion_contains_oxidation` and `spectrum_contains_oxidation` provides a direct test of whether a feature is genuinely localising the modification at the ion level.

3.4.2 Theoretical fragment matching

We use `spectrum_utils` (Bittremieux, 2020; Bittremieux et al., 2023) to generate the theoretical fragment ions of each peptide and match them against the observed peaks. Matching is performed at a tolerance of 20 ppm, which is appropriate for the Orbitrap data in the nine-species benchmark. Only monoisotopic fragments are generated (`max_isotope=0`): isotope peaks of a real fragment fall through to the noise label, which is the conservative choice for per-token labelling.

Modification masses in the benchmark are written as rounded delta-mass strings (e.g. `+15.99` for oxidation). At 20 ppm, this rounding can push a modified fragment outside the tolerance window. We therefore substitute each recognised PTM's delta mass with its exact monoisotopic value before annotation: $+15.994915$ Da for oxidation, $+57.021464$ Da for carbamidomethyl-cysteine, and $+0.984016$ Da for deamidation. PTM identity is resolved through InstaNovo's canonical `LEGACY_PTM_TO_UNIMOD` look-up table, which maps the model's own modification tokens to UNIMOD identifiers, ensuring that the concept labels are aligned with what the encoder saw during training.

For each matched peak, the best candidate annotation is selected by a priority ordering: monoisotopic over isotopic, no neutral loss over neutral loss, b/y backbone ions over a/c/x/z, lower charge over higher charge, and smaller mass error as a tiebreaker. This ordering reflects the physical expectation that the dominant annotation for a matched peak is the one most consistent with the observed mass.

Ion geometry and positional coverage. For a peptide of length n, a backbone ion of type t and index k covers a contiguous set of residue positions:

$$\text{covers}_b(k, p) = \mathbf{1}[1 \leq p \leq k], \tag{3.4.1}$$

$$\text{covers}_y(k, p) = \mathbf{1}[n - k + 1 \leq p \leq n], \tag{3.4.2}$$

where p is a 1-indexed residue position. For internal fragments with start index s and end index e, coverage is $\mathbf{1}[s \leq p \leq e]$. Immonium ions are a special case: they attest to the presence of a specific residue anywhere in the peptide but carry no positional information, so they drive the `is_I_ion` and matching `covers_`R concepts but never the position or PTM-localisation concepts. The coverage function is what makes ion-level PTM localisation possible: a concept such as `ion_contains_oxidation` is true for a matched peak if and only if (i) the peak matches a fragment ion and (ii) at least one of the modified residue's positions falls within the ion's coverage range.

3.4.3 Concept registry

Each token is labelled with 50 binary concepts organised into 14 families. Table 3.1 lists the full registry. The families group naturally into ion identity properties (type, charge, neutral losses), precursor context (precursor charge, peptide length), positional and coverage properties (terminus proximity, residues covered, terminal ion status), spectral properties (m/z position, relative intensity), cleavage specificity, and PTM localisation at both the ion and spectrum level The final family – spectrum-level PTM diagnostics – serves as a negative control: these labels are true for every token in a spectrum containing a given modification, regardless of whether the specific ion localises it. A feature that scores well on `ion_contains_oxidation` but poorly on `spectrum_contains_oxidation` is a genuine localisation detector; the reverse pattern indicates a spectrum-level association.

3.5 Evaluation Framework

The evaluation is organised around four claims that the pipeline must establish in sequence. First, the SAE must be a *faithful proxy* for the encoder layer it is trained on; features identified in the SAE reconstruction must reflect what the encoder is actually computing, not artefacts of the dictionary. Second, the learned features must be *sparse and well-distributed*; a dictionary in which a handful of features account for nearly all activations offers much less interpretable coverage than one in which firing is spread across many features. Third, individual features must *correspond to chemical concepts* in a statistically principled way. Fourth, the associations identified must reflect *genuine causal use* by the model, not coincidental co-occurrence in the data.

3.5.1 Reconstruction quality

The primary reconstruction metric is the fraction of variance explained (FVE, Eq. 2.5.3); all reported values use the centred formulation, where the denominator is computed relative to the per-dimension mean so that a reconstructor with no explanatory power scores exactly zero. We additionally report cosine similarity between x and $\hat{x}$, norm preservation $\|\hat{x}\|/\|x\|$, and FVE disaggregated by token type (latent vs. peak), ion type, and precursor charge state.

To measure how faithfully the SAE preserves InstaNovo's sequencing behaviour, we evaluate the model in four configurations on 5,000 held-out spectra:

1. *Clean*: original encoder activations, no intervention.

Table 3.1: The 50-concept, 14-family annotation registry. Concepts marked with † belong to the diagnostic family.

Family	Concepts	Notes / Definition
Ion type	`is_b_ion`, `is_y_ion`	Canonical N- and C-terminal backbone fragments.
	`is_I_ion`	Immonium ion; indicates residue presence but lacks positional information.
	`is_internal_fragment`	Fragment generated by two backbone cleavages.
	`is_precursor_related`	Unfragmented precursor or precursor with neutral loss.
	`is_matched_peak`, `is_noise_peak`	Any theoretically matched fragment vs. unmatched background.
	`is_latent_token`	The position-0 summary token appended by InstaNovo.
Neutral loss	`is_H2O_loss`, `is_NH3_loss`	Fragment with specific loss of water or ammonia.
	`has_neutral_loss`	Fragment carrying any neutral loss.
Fragment charge	`is_fragment_charge_1`, `is_fragment_charge_2`	Charge state of the matched fragment ion.
Precursor charge	`precursor_charge_2`, `precursor_charge_3`, `precursor_charge_4`	Charge state of the intact parent peptide.
Position	`position_Nterm`, `position_middle`, `position_Cterm`	Fragment sequence coverage relative to the peptide termini.
Mass region	`mz_low`, `mz_mid`, `mz_high`	$m/z < 300$, $300 \leq m/z \leq 1000$, and $m/z > 1000$ Da.
Intensity	`top_decile_intensity`	Peak intensity falls within the top 10% of its respective spectrum.
First/last ion	`is_first_ion`, `is_last_ion`	Terminal ions that anchor the start or end of a b- or y-ion ladder.
Canonical cleavage	`cleaves_C_to_Lys`, `cleaves_C_to_Arg`	Fragment generated by tryptic cleavage at Lysine (K) or Arginine (R).
Enhanced cleavage	`cleaves_N_to_Pro`, `cleaves_C_to_Asp`, `cleaves_C_to_Glu`	Non-tryptic cleavages known to be highly favoured in CID/HCD fragmentation.
Residue coverage	`covers_K`, `covers_R`, `covers_W`, `covers_F`, `covers_Y`, `covers_P`, `covers_M`, `covers_C`, `covers_D`, `covers_E`, `covers_N`	The fragment's sequence span explicitly covers the specified amino acid.
Peptide length	`peptide_length_short`, `peptide_length_medium`, `peptide_length_long`	Parent peptide length: < 7, 7–15, and ≥ 16 residues.
Ion-level PTM	`ion_contains_oxidation`, `ion_contains_cam_cys`, `ion_contains_deamidation`	The modification falls strictly within the fragment's specific sequence coverage.
Spectrum PTM†	`contains_oxidation`, `contains_cam_cys`, `contains_deamidation`	The sequence contains the modification anywhere (diagnostic negative control).

2. *SAE*: the target layer's output replaced by the SAE reconstruction $\hat{x} = \mathrm{SAE}(x)$.
3. *Zero*: the target layer's output set to zero; a destructive baseline that establishes a floor on task performance.
4. *Mean*: the target layer's output replaced by the mean activation over the batch; a less destructive baseline.

The headline measure is loss recovered (Eq. 2.5.4). We additionally report the top-1 accuracy drop in percentage points.

3.5.2 Feature structure and sparsity

Per-feature firing counts are accumulated across the full evaluation set and categorised by token type. We compute the per-token L0 distribution (number of active features per token), the firing-rate distribution across features, and the number of dead features. A feature is considered *dead* if it never fires on any evaluation token.

For dictionary geometry, we report encoder–decoder alignment, the effective rank of the decoder matrix, and the prevalence of near-duplicate directions. Encoder–decoder alignment is the absolute cosine similarity between each feature's encoder direction $W_{\mathrm{enc}}[:, f]$ and its decoder direction $W_{\mathrm{dec}}[f, :]$, averaged across all features:

$$\mathrm{align}(f) = \left|\cos\left(W_{\mathrm{enc}}[:, f],\ W_{\mathrm{dec}}[f, :]\right)\right|, \tag{3.5.1}$$

Perfect alignment ($\mathrm{align} = 1$) would mean the encoder is simply the transpose of the decoder, preventing the SAE from learning a useful sparse projection while near-zero alignment would indicate that the detection and reconstruction directions share no geometric relationship. The effective rank of W_{dec} is the entropy-perplexity of its singular value spectrum, where σ_i are the singular values of W_{dec} and p is their distribution normalised to sum to one.

$$\mathrm{erank}(W_{\mathrm{dec}}) = \exp\big(H(p)\big), \qquad p_i = \frac{\sigma_i}{\sum_j \sigma_j}, \tag{3.5.2}$$

This entropy-perplexity measure quantifies how many independent directions the decoder is effectively using: it equals the full rank when all singular values are equal, and collapses toward one when a single direction dominates. Pairwise decoder-column cosine similarities are estimated on a random sample of 2,000 features; near-duplicate pairs are those with cosine similarity above 0.95 within the sample. Exact computation of the full $12{,}288 \times 12{,}288$ similarity matrix is computationally prohibitive, so the sample count is a lower bound on the true number of near-duplicate pairs rather than an exact global count. We also report the Gini coefficient of the firing-rate distribution as a measure of inequality in feature utilisation. We quantify the sparsity–fidelity trade-off by evaluating FVE and mean L0 at six multiplicative scalings of the calibrated JumpReLU threshold $\{0.5\times, 0.75\times, 1.0\times, 1.25\times, 1.5\times, 2.0\times\}$ on a held-out subsample.

3.5.3 Feature-concept associations

For each feature f and concept c, we maintain a 2×2 contingency table of token counts:

	$c = 1$	$c = 0$
f fires	n_{11}	n_{10}
f silent	n_{01}	n_{00}

(3.5.3)

Tokens for which the concept is not applicable (e.g. ion-type concepts are not defined for the latent token, which has no mass assignment) are assigned $c = -1$ and excluded from that concept's table. From the contingency table, we derive precision $n_{11}/(n_{11} + n_{10})$, recall $n_{11}/(n_{11} + n_{01})$, and lift $P(c \mid f)/P(c)$.

Statistical significance is assessed with a chi-squared test on each contingency table and corrected for multiple comparisons across the full $F \times K = 12{,}288 \times 50 = 614{,}400$ feature-concept pairs using the Benjamini–Hochberg (BH) procedure at $\alpha = 0.05$, controlling the false discovery rate rather than the family-wise error rate.

Dominance-weighted $\mathrm{F1}_{\mathrm{dom}}$. Our concept labels are defined at the level of individual tokens, a peak either matches a y-ion or it does not, but the features that SAE training discovers operate at their own resolution, which need not coincide with ours. A feature encoding *high-m/z y-ions near the C-terminus* will fire on a strict subset of all y-ion tokens; evaluated against `is_y_ion` it incurs a recall penalty for every y-ion token it correctly ignores, even though it is detecting a chemically coherent sub-concept. The inverse problem is equally common; a feature that fires on both y-ions and the most intense noise peaks in the same m/z region will achieve high recall but is not a specific y-ion detector. Standard token-level recall therefore conflates two distinct failure modes, insufficient coverage and insufficient specificity, and neither alone is adequate as a ranking criterion.

The critical metric for interpretability is not coverage but *selectivity*: the feature should fire substantially more often on concept-positive tokens than on concept-negative tokens. We measure this through the within-class conditional firing rates.

$$p_+(f, c) = \frac{n_{11}}{n_{11} + n_{01}}, \tag{3.5.4}$$

$$p_-(f, c) = \frac{n_{10}}{n_{10} + n_{00}}, \tag{3.5.5}$$

where p_+ is the probability that f fires given a concept-positive token and p_- is the corresponding probability on negative tokens. The dominance factor equals 1 when the feature fires exclusively on positive tokens, falls continuously to 0 as the firing rates on positive and negative tokens converge, and is clamped to 0 whenever $p_- \geq p_+$.

$$\mathrm{dom}(f, c) = \max\Big(0,\ 1 - \frac{p_-(f, c)}{p_+(f, c)}\Big) \tag{3.5.6}$$

The headline association metric is the product of token-level F1 and the dominance factor.

$$\mathrm{F1}_{\mathrm{dom}}(f, c) = \mathrm{F1}(f, c) \times \mathrm{dom}(f, c), \tag{3.5.7}$$

This penalises features that achieve a good F1 by firing indiscriminately at the cost of no additional specificity, while rewarding features that are both accurate and selective regardless of whether they cover the full extent of concept-positive tokens. Per-concept top-feature tables are ranked by the composite score $\mathrm{F1}_{\mathrm{dom}} \times \log(1 + \mathrm{lift})$, which additionally up-weights features with strong positive enrichment over the concept base rate.

3.5.4 Causal evaluation

Correlational evidence that a feature fires on concept-positive tokens is necessary but not sufficient to conclude that the model uses that feature to represent the concept (Marks et al., 2025). To move from correlation to causation, we perform a targeted ablation intervention and measure its effect against two controls.

Ablation procedure. For each concept c with at least one BH-significant feature, a group of 10 features is selected for ablation using a stratified procedure; BH-significant features are partitioned into five equal-frequency bins on the log-scale firing-rate distribution, and two features are sampled from each bin weighted by $\mathrm{F1}_{\mathrm{dom}}$. Stratifying by firing rate ensures that narrowly selective and low-frequency features, which a global $\mathrm{F1}_{\mathrm{dom}}$ ranking would demote in favour of high-frequency features, are represented in the ablation group. The selected features' activations are zeroed and the modified SAE reconstruction is substituted for the target encoder layer's output for 5,000 evaluation spectra. The ablation baseline is the SAE reconstruction *without* ablation, not the clean model; this ensures that the SAE's own reconstruction error is common to both the ablated and unablated forward passes and cancels in the delta, so Δ_{CE} isolates the marginal causal effect of the ablated features rather than the cumulative approximation error of the SAE.

The key quantity is whether ablating the feature group degrades prediction selectively on spectra where the concept is prevalent. Each spectrum is assigned a prevalence score for concept c: the fraction of its encoder tokens that carry label c. Spectra are then divided into terciles by this prevalence score, and selectivity is the contrast between the mean Δ_{CE} on the top tercile and the mean on the bottom tercile:

$$\mathrm{sel}(f, c) = \mathbb{E}[\Delta_{\mathrm{CE}} \mid \mathrm{prev}_c \geq q_{2/3}] - \mathbb{E}[\Delta_{\mathrm{CE}} \mid \mathrm{prev}_c \leq q_{1/3}], \tag{3.5.8}$$

where $q_{1/3}$ and $q_{2/3}$ are the empirical tercile boundaries of the per-spectrum prevalence distribution. A positive value indicates that ablating the feature group degrades sequence prediction more severely on high-prevalence spectra than on low-prevalence ones, the falsifiable signature of a concept-specific feature. If the prevalence distribution is degenerate (e.g. a concept present in every spectrum), the tercile contrast is undefined and selectivity is reported as `NaN`.

We additionally compute an *orthogonalised selectivity* by restricting the contrast to spectra that carry none of the concepts correlated with c (those whose ϕ-coefficient with c exceeds 0.3). This removes the possibility that the selectivity signal is borrowed from a co-occurring concept rather than from c itself. When too few such spectra remain ($n < 6$), the result is reported as `NaN`, reflecting that the concepts are not separable in the evaluation set rather than that the test passed.

To control for the possibility that any high-firing feature group produces a selective effect by sheer magnitude, we compare against a *firing-rate-matched random control*: for each ablation we draw five control groups of the same size, each feature matched within ± 0.3 dex on the log-scale firing-rate distribution. This yields standardised effect sizes:

$$\mathrm{sel}_z = \frac{\mathrm{sel}_{\mathrm{target}} - \mu_{\mathrm{random}}}{\sigma_{\mathrm{random}}}, \qquad \mathrm{mag}_z = \frac{\Delta_{\mathrm{CE}}^{\mathrm{target}} - \mu_{\mathrm{random}}^{\Delta}}{\sigma_{\mathrm{random}}^{\Delta}}, \tag{3.5.9}$$

where μ and σ are estimated from the five control replicates. $\mathrm{sel}_z > 0$ indicates concept-specific causal involvement above the firing-rate baseline; $\mathrm{mag}_z > 0$ indicates that the ablation effect is larger in magnitude than expected from features of equivalent activity. We use $|z| > 1.96$ as a nominal reference throughout.

3.5.5 Cross-layer feature stability

The four-layer training produces four independent SAEs. Features that represent the same chemical concept at different encoder depths should produce similar latent representations on the same tokens; the linear representation hypothesis predicts that the same concept is written into representation space consistently across layers, even as the model builds progressively more abstract encodings on top of it.

We examine cross-layer stability through the Pearson correlation between the per-token SAE feature activation vectors of matched features across layers. The *anchor* features are the top-100 features at the target layer by $F1_{dom}$ across any BH-significant concept, selecting the features with the strongest and most reliable concept associations rather than simply the most active ones. For each anchor f at the target layer, the best-matching feature at every other layer is identified as

$$f^*(f) = \arg \max_{f' \in \mathcal{F}_{L'}} \mathrm{corr}(a_f, a_{f'}) \,, \tag{3.5.10}$$

where $a_f \in \mathbb{R}^T$ is the vector of post-JumpReLU SAE feature activations of feature f over a sample of $T = 100{,}000$ tokens drawn from the extraction chunks, and corr denotes the Pearson correlation (mean-centred dot product normalised by the product of norms). The top-5 matches per anchor are recorded to allow inspection of near-ties; the best match is used for the stability statistics below.

Features with a best-match correlation above 0.9 indicate a stable representational direction that persists across encoder layers. Features with a best-match correlation below 0.5 are layer-specific: the information they encode is either computed at the target depth and not yet present in the earlier layer, or is present in a qualitatively different form that the earlier SAE does not recover as a single feature. These layer-specific features are the most informative targets for causal analysis because they represent computations that emerge at a specific depth rather than information that persists through the encoder stack.

4. Results

This chapter reports the empirical findings from applying the SAE pipeline to the full nine-species benchmark. Independent SAEs were trained at encoder layers 2, 4, 6, and 8 and evaluated against the 50-concept per-peak annotation registry described in Chapter 3. The sections follow the evaluation hierarchy established in Section 3.5: each builds on the previous, and the narrative through the chapter mirrors the logic of the claims; from faithfulness, through structure, through association, to causation.

4.1 Reconstruction Fidelity

4.1.1 Fraction of variance explained

Table 4.1 reports the primary reconstruction metrics at each layer. All four SAEs explain more than 92% of activation variance under $k = 32$ active features per token, and mean L0 sits within 0.5 of the training target at every layer, confirming that BatchTopK and the calibrated JumpReLU threshold are jointly enforcing the intended sparsity level. FVE decreases monotonically from 98.4% at layer 2 to 92.0% at layer 8. Shallow layers process representations that are close to the input embeddings and vary smoothly across spectra, while deeper layers encode the output of progressive, task-specific computation whose variation is harder to compress into 32 active features. The calculated FVE at layer 8 means the sparse representation captures the great majority of variation in the activations that drive sequence generation.

4.1.2 Threshold sensitivity

A well-calibrated SAE should not depend critically on the precise threshold used at inference. Figure 4.1 reports FVE and mean L0 as a function of a multiplicative scaling of the calibrated JumpReLU threshold.

Two patterns stand out. Layers 2 and 4 are relatively insensitive: doubling the multiplier from $1.0\times$ to $2.0\times$ costs only 1.2 and 2.4 FVE points while reducing mean L0 to approximately 22 and 19. Layers 6 and 8 are substantially more sensitive: the same doubling costs 4.7 and 7.7 FVE points. Because raising the threshold zeroes the lowest-magnitude activations first, this cost measures how much of a layer's reconstruction its weaker-firing features carry. At layers 6 and 8 a meaningful share rests on low- and medium-magnitude features; at layers 2 and 4 the reconstruction is dominated by a few high-magnitude features, and the weaker activations contribute so little to its variance that raising the threshold discards them at almost no cost. For interpretability analyses where a lower L0 improves feature readability, a $1.5\times$ multiplier is a reasonable choice at layers 2 and 4 at negligible fidelity cost; at layers 6 and 8 the trade-off is less favourable.

4.2 Feature Structure and Dictionary Health

4.2.1 Sparsity and feature utilisation

Table 4.2 reports feature-health metrics across layers.

The strict-dead rate decreases monotonically with depth, from 44.1% at layer 2 to 29.0% at layer 8. Under the per-token sparsity budget, a feature stays alive only if it becomes competitive across a broad range of inputs, so the dead rate reflects how much recurring structure a layer presents to a dictionary of fixed size. At shallow layers, where cross-peak context is not yet integrated, the uniform $16\times$ expansion

Table 4.1: Reconstruction quality at each encoder layer on the full nine-species benchmark. Mean L0 is the average number of active features per token under the calibrated JumpReLU threshold.

	Layer 2	Layer 4	Layer 6	Layer 8
FVE	0.984	0.969	0.943	0.920
Mean L0	32.3	32.0	31.9	31.8

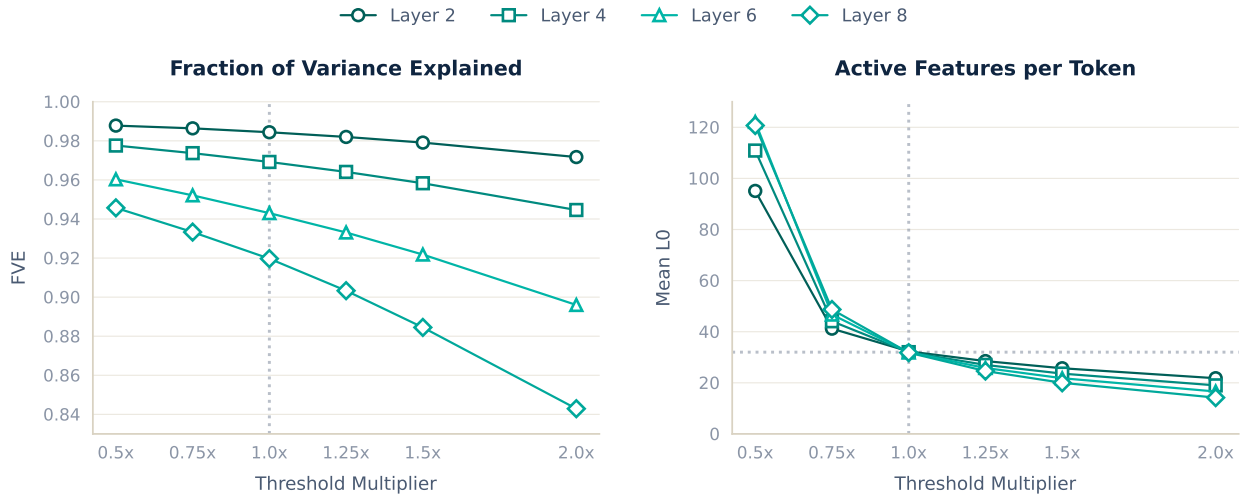


Figure 4.1: Threshold sensitivity sweep. FVE (left) and mean L0 (right) as functions of a multiplicative scaling of the calibrated JumpReLU threshold, evaluated on a 100-chunk held-out subsample; the calibrated value corresponds to multiplier $1.0\times$. Layers 2 and 4 are substantially more robust to the scaling than layers 6 and 8.

offers more capacity than the activations can fill, leaving a larger share of features dead; deeper layers present more recurring structure and take up correspondingly more of the dictionary. The trend is therefore best read as a capacity-utilisation signal rather than a direct measure of representational richness: it is consistent with deeper activations being richer, but the high shallow-layer dead rate equally reflects a dictionary larger than these layers can use, which a depth-matched expansion factor would fit more closely.

The Gini coefficient of the firing-rate distribution, a measure of inequality ranging from 0 (uniform) to 1 (maximally concentrated), sits between 0.908 and 0.938 across all layers. Elevated Gini values are partly a structural consequence of the top-k sparsity scheme: dead features contribute firing rates of exactly zero, inflating the inequality of the distribution. But the pattern also reflects a genuine property of the data: concepts such as ion type (present in every spectrum) necessarily drive higher firing rates than rare events such as specific cleavage sites or PTM localisation, so a dictionary that represents both will exhibit firing-rate inequality by design. The modest but consistent decrease in Gini coefficient with depth is consistent with a broader distribution of firing rates at later layers, where more of the dictionary is active.

The fraction of features with at least one BH-significant concept association grows from 54.8% at layer 2 to 70.9% at layer 8, tracking the decrease in dead-feature rate but exceeding it: features at deeper layers are not merely better-utilised in terms of firing rate, they are also more semantically structured relative to the chemical annotation registry.

Table 4.2: Feature utilisation and sparsity across encoder layers. Features with ≥ 1 concept counts features with at least one BH-significant concept association (Section 4.3).

Metric	Layer 2	Layer 4	Layer 6	Layer 8
Total features (F)	12,288	12,288	12,288	12,288
Strict-dead (%)	44.1	39.7	37.2	29.0
Features with $\geq$1 concept	54.8%	59.5%	62.1%	70.9%
Gini coefficient	0.937	0.930	0.924	0.908

Table 4.3: Geometric properties of the trained SAE dictionaries. Encoder-decoder alignment is the mean absolute cosine similarity between each feature's encoder direction $W_{\text{enc}}[:, f]$ and its decoder direction $W_{\text{dec}}[f, :]$. Effective rank is the entropy perplexity of the singular value spectrum of W_{dec}.

Metric	Layer 2	Layer 4	Layer 6	Layer 8
Encoder–decoder alignment	0.604	0.590	0.591	0.442
Effective rank	705.1	703.3	707.3	670.3
Near-duplicate pairs	0	0	0	0

4.2.2 Geometric structure

Table 4.3 reports three geometric properties of the trained dictionaries: how closely each feature's detection direction matches the direction it writes back (alignment), how many independent directions the decoder spans (effective rank), and whether any features have collapsed onto a shared direction (near-duplicate pairs). The alignment metric and the meaning of its extremes are defined in Section 3.5.2.

At layers 2–6, encoder–decoder alignment is moderate, at approximately 0.60. The direction along which a feature is detected and the direction along which it is reconstructed are related but not identical, as expected of a well-trained untied dictionary. Layer 8 is the exception, at 0.442. The lower value means the encoder singles out each feature along a direction noticeably different from the one it adds back during reconstruction. Layer 8 carries the model's most chemically specific features: cleavage specificity and modification localisation score highest here (Section 4.3), and resolving many such fine distinctions forces their directions to overlap once they share a 768-dimensional residual stream. When feature directions overlap, detecting one cleanly requires projecting along a direction that suppresses interference from its neighbours, which need not match the direction used to write it back; the encoder tilts its detectors away from the decoder columns accordingly, and alignment falls. The lower effective rank at layer 8 is the same crowding seen from the decoder side: across a fixed set of 12,288 columns, fewer independent directions means the reconstruction atoms are more linearly dependent, and so more overlapping, than at the shallower layers.

This denser packing stops short of redundancy. No decoder column pair exceeds cosine similarity 0.95 at any layer, so the dictionary has not collapsed into duplicate detectors; a failure mode that would waste capacity and inflate the apparent feature count. Layer 8 packs many *distinct* chemical features into a shared subspace; it does not encode the same feature twice.

Table 4.4: Aggregate feature-concept association statistics. BH-significant pairs are (feature, concept) pairs passing Benjamini–Hochberg correction at $\alpha = 0.05$ over all $12{,}288 \times 50 = 614{,}400$ pairs.

Metric	Layer 2	Layer 4	Layer 6	Layer 8
BH-significant pairs	277,291	310,660	324,746	394,464
Features with $\geq$1 sig. concept	6,735	7,309	7,633	8,718
Maximum $F1_{dom}$	0.999	0.999	0.998	0.994

4.3 Feature-Concept Associations

4.3.1 Aggregate association statistics

Table 4.4 summarises the scale of BH-significant feature-concept associations across layers.

The number of significant feature-concept pairs grows monotonically with depth, increasing by 42% from layer 2 to layer 8. The growth accelerates between layers 6 and 8, which contribute 69,718 additional pairs compared to 47,455 in the preceding three layers. This acceleration is consistent with the geometric finding of lower encoder-decoder alignment at layer 8: a dictionary that is resolving finer distinctions will necessarily produce more and sharper concept associations.

4.3.2 Concept-level results

Table 4.5 reports the best-feature $F1_{dom}$ for each concept at each layer. Rather than narrating the full table, we highlight the structural patterns it reveals.

The table reveals three broad regimes that we discuss in turn.

Physical spectrum properties. `top_decile_intensity`, a property carried directly by the input peak embeddings, is represented most strongly at the shallowest layer examined: it reaches its maximum $F1_{dom}$ of 0.671 at layer 2 and is lower at every deeper layer, indicating that relative peak intensity is encoded as a strong, separable direction early in the stack. `peptide_length_long`, a spectrum-level property, shows the same depth profile, peaking at layer 2 ($F1_{dom} = 0.286$) and falling at every deeper layer. Mass region does not share this shallow-layer profile: `mz_low` and `mz_mid` are strongest at layer 4, while `mz_high` is detected comparably at both ends of the stack ($F1_{dom} = 0.617$ at layer 2 and 0.628 at layer 8, dipping in between) but by different top features at each end.

Immonium ions: an early-layer signal. Immonium ions are detected at layer 2 by a single, essentially pure feature (F2666): its $F1_{dom}$ of 0.777 is the highest of any concept in the evaluation other than the latent summary token. The feature fires almost exclusively on immonium ions (dominance ≈ 1) and recovers a majority of them. This is the cleanest single-feature detection of any fragment concept, and it occurs at the shallowest layer examined; counter to the general trend of concepts becoming more detectable as the encoder accumulates context. Immonium ions sit at fixed, low m/z values characteristic of each residue, directly readable from the input m/z embedding before it is heavily transformed, which might account for such a sharp detector emerging this early.

The feature does not dissolve at greater depth. An equally pure immonium detector is present at every layer (dominance ≈ 1, lift near $30\times$ throughout); what falls after layer 2 is coverage; its $F1_{dom}$ drops by almost half at layer 4 and stays well below the layer-2 value thereafter, because no single deeper feature spans the full set of immonium ions. This is the same coverage effect seen for backbone ions: selectivity is retained, but the broad `is_I_ion` label is no longer captured by one feature.

Table 4.5: Best-feature $F1_{dom}$ per concept per layer. The highest score for each concept across the four layers is shown in bold.

Concept	L2	L4	L6	L8
Ion type				
is_latent_token	**0.999**	0.999	0.998	0.994
is_I_ion	**0.777**	0.440	0.502	0.406
is_y_ion	0.283	0.270	0.328	**0.372**
is_b_ion	0.129	0.172	0.209	**0.223**
is_first_ion	**0.561**	0.415	0.549	0.556
is_last_ion	0.062	0.088	0.185	**0.232**
is_matched_peak	0.187	0.180	0.169	**0.208**
is_noise_peak	0.224	0.140	0.112	**0.308**
is_internal_fragment	0.137	0.144	0.196	**0.295**
is_precursor_related	0.042	0.053	**0.159**	0.092
Neutral loss				
is_H2O_loss	0.096	0.277	**0.351**	0.319
is_NH3_loss	0.142	0.133	0.165	**0.257**
has_neutral_loss	0.163	0.319	**0.328**	0.300
Fragment charge				
is_fragment_charge_1	0.202	**0.278**	0.198	0.181
is_fragment_charge_2	0.158	0.177	0.173	**0.187**
Precursor charge				
precursor_charge_2	0.065	0.065	**0.105**	0.053
precursor_charge_3	**0.063**	0.040	0.038	0.037
precursor_charge_4	0.131	0.104	**0.134**	0.079
Position				
position_Nterm	0.176	0.100	0.173	**0.181**
position_middle	0.194	0.240	0.195	**0.286**
position_Cterm	0.215	0.221	**0.288**	0.227
Mass region				
mz_low	0.365	**0.453**	0.328	0.326
mz_mid	0.368	**0.425**	0.363	0.405
mz_high	0.617	0.464	0.431	**0.628**
top_decile_intensity	**0.671**	0.472	0.591	0.474
Cleavage specificity				
cleaves_N_to_Pro	0.190	0.182	0.259	**0.327**
cleaves_C_to_Lys	0.061	0.097	0.099	**0.233**
cleaves_C_to_Arg	0.117	0.119	0.170	**0.244**
cleaves_C_to_Asp	0.059	0.104	0.253	**0.279**
cleaves_C_to_Glu	0.052	0.105	0.143	**0.246**
Residue coverage (selected)				
covers_R	0.161	0.197	0.226	**0.239**
covers_K	0.135	0.157	0.188	**0.229**
covers_W	**0.194**	0.193	0.186	0.144
covers_P	0.103	0.122	0.173	**0.174**
Peptide length				
peptide_length_short	**0.126**	0.074	0.092	0.115
peptide_length_medium	**0.204**	0.133	0.172	0.107
peptide_length_long	**0.286**	0.151	0.162	0.091
Ion-level PTM localisation				
ion_contains_oxidation	0.081	0.151	0.158	**0.302**
ion_contains_cam_cys	0.089	0.135	**0.159**	0.148
ion_contains_deamidation	**0.065**	0.065	0.064	0.064

Ion ladders and terminal ions: a non-monotonic pattern. General backbone-ion detection strengthens with depth: `is_b_ion` rises monotonically to its maximum at layer 8, and `is_y_ion` is also strongest there. This fits backbone-series membership being a property that depends on relating a peak to the rest of the spectrum, which the encoder resolves more sharply as it accumulates cross-peak context with depth. The modest absolute $F1_{dom}$ of both concepts is a coverage effect, not a selectivity one: the best b- and y-ion features fire almost exclusively on backbone ions (dominance above 0.9) but each detects only a specific sub-type of the series, covering a small fraction of all b- or y-ions and so taking a recall penalty against the undivided label; the coverage-versus-selectivity distinction that $F1_{dom}$ is built to separate (Section 3.5.3). Their lift confirms genuine enrichment, at $6.5\times$ the base rate for the best layer-8 b-ion feature.

The `is_first_ion` concept behaves differently. It is strongly detected at every depth and at far higher enrichment than the general backbone concepts (lift near $40\times$ throughout), and its $F1_{dom}$ stays high at layers 2, 6, and 8 with a single dip at layer 4; the same feature (F5910) leads at layers 4 and 6, so the layer-6 value is that detector sharpening rather than a new one emerging. At layer 8, one feature (F296) is the top detector for both `is_first_ion` and `position_Cterm`; a single direction tracking terminal-ion identity together with C-terminal position, consistent with a detector tuned to the C-terminal anchor of the ladder.

Neutral losses and fragment charge. Several neutral-loss and fragment-charge concepts are represented most strongly in the middle of the stack: `is_H2O_loss` and `has_neutral_loss` peak at layer 6, and `is_fragment_charge_1` at layer 4. The other members of these families peak deeper, at layer 8 (`is_NH3_loss` and `is_fragment_charge_2`), so neither family is uniformly mid-stack. The mid-stack concepts are consistent with properties that need more than the raw m/z and intensity in the input embeddings, but not the full contextual integration the layer-8 features draw on.

Cleavage specificity and PTM localisation. The most chemically specific concepts are concentrated in the final layer. All five cleavage concepts reach their best $F1_{dom}$ at layer 8, as do ion-level oxidation localisation and most residue-coverage concepts. Lift is striking for these rare events: the best layer-8 feature for `C_to_Arg` fires at $557\times$ the base rate, a highly concentrated detector for Arg-C-terminal cleavage fragments, though its $F1_{dom}$ stays modest at 0.244.

Ion-level PTM localisation shows the same depth dependence: `ion_contains_oxidation` rises monotonically from layer 2 to a layer-8 peak (0.302, lift $46\times$), consistent with identifying *which* ion carries a modification being a computation that accumulates with depth. The spectrum-level diagnostic `spectrum_contains_oxidation` stays low at every layer; by construction, since it marks every token in an oxidised spectrum, no single sparse feature can cover it. That the ion-level concept scores well above this diffuse control is the evidence that the layer-8 signal is genuine ion-level localisation rather than a coarse spectrum-level correlate.

`ion_contains_deamidation` is the exception to the oxidation pattern above: its $F1_{dom}$ is uniformly low (≈ 0.065) across all layers, with no dedicated detector at any depth. Deamidation produces a $+0.984$ Da shift that sits about 0.02 Da from the $+1$ isotope of the unmodified fragment. A plausible reason it is missing is that the shift lies so close to the isotope spacing that the deamidation signal is hard to separate from the ordinary isotopic envelope in the spectrum itself, which would discourage the model from dedicating a feature to it.

Table 4.6: End-to-end task preservation. CE is cross-entropy per amino-acid position. Top-1 drop is in percentage points.

	Layer 2	Layer 4	Layer 6	Layer 8
CE (clean)	0.630	0.630	0.630	0.630
CE (SAE patched)	0.632	0.636	0.644	0.642
CE (zero ablation)	3.458	3.857	3.782	3.212
CE (mean ablation)	3.402	3.288	3.113	2.996
Loss recovered	99.94%	99.82%	99.56%	99.55%
Top-1 accuracy (clean)		82.6%		
Top-1 drop (SAE)	0.08 pp	0.22 pp	0.42 pp	0.82 pp

Table 4.7: Cross-layer activation correlation for the top-100 layer-8 features (by $F1_{dom}$) matched to their nearest layer-6 counterpart.

Metric	Value
Mean correlation	0.684
Median correlation	0.705
Correlation > 0.9	27
Correlation > 0.7	51
Correlation < 0.5	22

4.4 Task Preservation

Before drawing causal conclusions from the feature-concept associations, we verify that the SAE reconstructions preserve the model's downstream peptide sequencing behaviour. Table 4.6 reports the loss recovery analysis across all four layers on 5,000 held-out spectra.

All four SAEs preserve more than 99.5% of the model's sequencing loss relative to zero ablation. The top-1 accuracy drop from patching the SAE into the forward pass is at most 0.82 percentage points (at layer 8), and at layer 2 is essentially imperceptible (0.08 pp). Loss recovered decreases slightly with depth, consistent with the decrease in FVE: the harder a layer is to reconstruct, the more the downstream behaviour is affected. Crucially, even at layer 8, the layer closest to the decoder and therefore most sensitive to perturbation, the SAE reconstruction degrades sequencing accuracy by less than one percentage point. This confirms that the sparse representations are faithful proxies for the original activations, and that causal conclusions drawn from ablating specific features in the SAE code can be attributed to those features rather than to reconstruction artefacts.

4.5 Cross-Layer Feature Stability

To characterise which layer-8 features reflect stable, encoder-wide representations versus layer-specific computations, we matched the top-100 layer-8 features by $\mathrm{F1}_{\mathrm{dom}}$ against their nearest counterparts at layer 6 by Pearson correlation of per-token activation vectors. Table 4.7 summarises the distribution.

The distribution is approximately bimodal. Twenty-seven features have correlations above 0.9, indicating that they encode essentially the same direction at both layers 6 and 8. Inspection shows that this cluster corresponds overwhelmingly to latent-token features: the top-3 highest correlations (0.992–0.994) all

involve layer-8 anchors (F46, F4646, F5019) that match into the same tight cluster of layer-6 latent-token features (F478, F5308, F9368), confirming that the latent summary token's representation is effectively constant from layer 6 to layer 8.

At the other end, 22 features have correlations below 0.5, indicating they encode information that is either newly computed at layer 8 or is present at layer 6 in a qualitatively different form. These layer-specific features are the most informative targets for causal analysis because they represent computations that emerge, rather than information that simply persists. The causal ablation results below confirm that several of the strongest causal signals at layer 8 correspond to low-correlation features.

4.6 Causal Ablation

Correlational association does not establish that the model uses a feature (Section 3.5.4); the ablation test below is what moves from correlation to causation. That test is only informative if it can detect a genuine effect, so we begin with a positive control; features whose causal role is not in question, before moving to other detected concepts. A clear effect there shows the intervention has the sensitivity to register causal involvement, which means the null and weaker effects that follow are informative about the model, not about the probe's sensitivity.

4.6.1 Consistent signals across layers: latent token

The latent-token features are that control; the model depends on the summary token by construction and the SAE isolates it almost perfectly ($\text{F1}_{\text{dom}} \approx 1$), so removing them should register an effect. At layers 4–8, ablating the top latent-token features degrades cross-entropy more than firing-rate-matched random ablation on both measures ($\text{mag}_z = 3.9, 5.0, 3.0$; $\text{sel}_z = 7.1, 5.6, 4.9$). At layer 2 the effect is in the same direction but sub-threshold on both ($\text{mag}_z = 1.8$, $\text{sel}_z = 1.0$). The absolute degradation is small ($\Delta\text{CE} \approx 0.004$–$0.007$), as befits a sanity check rather than a dominant feature, and it tracks the latent token's status as the encoder's most stable direction across depth (Section 4.5). With the intervention shown to detect this known effect, the substantive results below can be attributed to the features ablated rather than to a probe that cannot register causation.

4.6.2 The strongest causal signal: is_y_ion at layer 2

The most striking causal finding in the entire evaluation is the effect of ablating the top y-ion features at layer 2. The group-level result is $\text{sel}_z = +76.2$ and $\text{mag}_z = +179.4$, with a mean change in cross-entropy of $+0.102$; by far the largest ablation effect observed. This is a catastrophic degradation in model performance caused by removing ten features from a dictionary of 12,288. The single best y-ion feature at layer 2 (F7522, $\text{F1}_{\text{dom}} = 0.283$, lift $2.9\times$) alone produces $\Delta\text{CE} = +0.102$ when ablated, with a selectivity of $+0.070$ confirming that the effect is concentrated on spectra where y-ions are present.

The magnitude of this effect warrants careful interpretation. The F1_{dom} score of the best layer-2 y-ion feature is modest (0.283) relative to, for example, the immonium feature (0.777), because y-ions appear in nearly every spectrum, the concept base rate is high and the dominance penalty is relatively mild for features that also fire on background tokens. What the causal result reveals is that a small group of y-ion features at layer 2 is functionally essential to the model's sequencing, not merely correlated with y-ions but genuinely necessary for the decoder to produce accurate amino-acid predictions. The contrast with the layer-2 immonium features is instructive: a much sharper correlational feature does not produce a comparably strong causal signal.

4.6.3 Immonium ions: selective but not functionally critical

The immonium features provide an instructive contrast with the y-ion result. At layer 2, `is_I_ion` produces $\text{sel}_z = +3.25$ (significantly selective) but $\text{mag}_z = +0.02$ (indistinguishable from random in magnitude). The immonium detector feature (F2666) that achieves $\text{F1}_{\text{dom}} = 0.777$, the sharpest correlational result in the evaluation, has a negligible effect on the model's cross-entropy when ablated. This dissociation between correlational strength and causal importance is a key finding: a monosemantic feature can be a faithful detector of a concept without that concept being necessary for the model's core computation. Immonium ions identify residue presence but not position; their sequencing contribution is apparently secondary to the positional and ladder information carried by backbone ions.

4.6.4 `is_first_ion` at layer 8: selectivity and magnitude together

At layer 8, the strongest double-significant causal result is `is_first_ion`: $\text{sel}_z = +15.1$, $\text{mag}_z = +20.9$, and $\Delta\text{CE} = +0.006$. The concept labels a fragment as the terminal ion of its ladder, specifically b_1 or y_1 (index 1 from either terminus), regardless of ion series; in HCD data the y_1 ion dominates empirically, which explains why the best first-ion feature (F296, $\text{F1}_{\text{dom}} = 0.556$, lift $38.0\times$) is simultaneously the top feature for `position_Cterm` (Section 4.3). This is a different regime from the layer-2 y-ion effect: smaller in absolute CE terms but highly specific. Ablating the top first-ion features at layer 8 degrades the model's predictions in a way that is disproportionately concentrated on spectra where terminal ions are present, and the effect is far larger in magnitude than random ablation of the same number of features. This result suggests the model relies on a specific feature group at layer 8 to identify the anchor point of the ion ladder, precisely the information needed to begin autoregressive decoding.

4.6.5 Selective effects without large CE changes

Several concepts show statistically significant selectivity ($|\text{sel}_z| > 1.96$) without a correspondingly large change in overall CE. At layer 8, `is_matched_peak` produces $\text{sel}_z = -5.0$ with $\Delta\text{CE} = -0.004$ (CE decreases when the features are ablated), and `covers_R` produces $\text{sel}_z = +7.2$ with a negligible ΔCE. These effects are informationally real, the ablation is having a concept-specific effect, but the small absolute CE change suggests that the ablated features are part of a redundantly encoded concept: other features compensate for the intervention, limiting the overall impact on prediction quality.

4.6.6 Sign-reversed effects

A subset of concepts shows significant effects with a negative sign on ΔCE; ablating the top features for the concept *improves* the model's cross-entropy rather than degrading it. `cleaves_C_to_Asp` at layer 8 is the clearest example: $\text{sel}_z = +5.98$ (the effect is concept-specific) but $\text{mag}_z = -8.62$ ($\Delta\text{CE} = -0.017$, CE decreases). Similarly, `covers_N` at layer 8 shows $\text{sel}_z = +2.67$, $\text{mag}_z = -4.50$, $\Delta\text{CE} = -0.018$.

We interpret these effects as evidence of representational redundancy. The same chemical information is encoded along multiple pathways in the residual stream simultaneously; when one group of features is ablated, the concept-relevant information remains accessible through these other pathways, and the forward pass is unimpaired. The CE improvement is then attributable to the removal of a weakly redundant signal that was interfering with the pathways already encoding the concept more faithfully. A further domain-specific consideration applies to non-tryptic cleavage concepts; features specialised to rare fragmentation events are likely undertrained on the nine-species benchmark, which is dominated by tryptic peptides, and their representations may be noisier than those of well-sampled concepts. Removing

a noisy redundant copy of a concept that is already encoded elsewhere would produce exactly the pattern observed. We report these as genuine causal involvements, the ablation has a real and concept-specific effect on the model behaviour, without asserting that the standard necessity claim applies.

4.7 Summary of Findings

The results across all evaluation phases are consistent with a single organising principle: InstaNovo's encoder builds chemical representations in stages that closely track the logical demands of the sequencing problem. Physical spectral properties are most sharply encoded at layer 2; contextual chemical properties emerge and strengthen through layers 4 and 6; the most chemically specific and positionally grounded representations appear at layer 8 and are confirmed by the cross-layer analysis to be genuinely emergent rather than persistent from earlier depths. Two results sit outside this depth-progression pattern, immonium ion detection peaks at layer 2 and declines thereafter, and peptide length is best encoded at layer 2 and degrades with depth; both are discussed in Chapter 5.

The causal ablation results refine this picture in ways a correlational analysis alone cannot reveal. The dominant causal signal is the y-ion feature group at layer 2: ablating ten features from a dictionary of 12,288 produces a catastrophic $\Delta\text{CE} = +0.102$, establishing that backbone ion detection at the shallowest layer is functionally essential to decoding, not merely correlated with it. By contrast, the layer-2 immonium detector, the sharpest single-feature result in the entire evaluation ($\text{F1}_{\text{dom}} = 0.777$), has no detectable causal effect when ablated, demonstrating that correlational strength does not imply functional necessity. At layer 8, the first-ion feature group produces a smaller but highly specific effect ($\text{sel}_z = +15.1$, $\text{mag}_z = +20.9$), suggesting the model relies on a dedicated terminal-ion detector to anchor autoregressive decoding. Several layer-8 concepts show significant selectivity with negligible or negative ΔCE, consistent with redundant encoding of rare chemical events across multiple pathways in the residual stream.

Chapter 5 interprets the full set of findings, situates them within the broader SAE-for-scientific-models literature, and addresses the limitations of the evaluation framework.

5. Discussion and Conclusion

This chapter draws together the empirical results of Chapter 4 into a coherent interpretation, situates that interpretation within the broader research landscape, confronts the limitations of the analysis, and outlines the directions that would extend or strengthen the findings.

5.1 Interpretation of the Results

The central reading of Chapter 4 is that InstaNovo's encoder constructs its chemical vocabulary in stages whose order follows the logical demands of the sequencing task. The subsections below build this account, qualify it with two early-layer anomalies, and draw out its consequences for the relationship between feature detection and feature use.

5.1.1 A depth-staged construction of chemical representation

The concept associations are sorted cleanly by depth. Physical spectral properties that are available directly in the input embedding are represented most sharply at or before layer 4. Properties that require evidence to be integrated across peaks emerge and strengthen through the middle of the stack. The most chemically specific properties improve monotonically and peak at layer 8, with cross-layer analysis confirming that the dominant layer-8 features are emergent rather than inherited from layer 6 (Section 4.5).

This ordering is consistent with what the transformer architecture can compute at each depth. A layer-2 attention head can identify a single peak's m/z and intensity without integrating context across other peaks; identifying ion-series membership requires relating a peak's mass to the complementary ion of the other terminus, a computation that might require several rounds of attention to converge; and cleavage specificity requires knowing which residue occupies the C-terminal position of a fragment, a property that depends on having resolved positional identity along the ladder. The depth at which each concept peaks thus reflects the depth at which the necessary cross-peak evidence is first available, not merely a learned ordering convention.

The dictionary statistics are consistent with the same picture from a different angle: the dead feature rate falls from 44.1% at layer 2 to 29.0% at layer 8 and the concept-associated fraction rises from 54.8% to 70.9% (Table 4.2), indicating that deeper layers present more recurring chemical structure to a fixed-size dictionary. That an SAE trained without chemical supervision recovers this vocabulary at the depths fragmentation theory predicts is the principal positive result of the thesis. It places InstaNovo's encoder alongside the protein language models of InterPLM (Simon and Zou, 2025) and InterProt (Adams et al., 2025); the distinction is that first-principles per-token labels allow depth localisation of concepts, not merely confirmation of their existence.

5.1.2 The latent summary token stabilises early

The latent summary token follows the opposite depth profile from the per-peak features. Its cross-layer correlations are the highest in the evaluation (0.992–0.994 between layers 6 and 8; Section 4.5), indicating that its representation is effectively fixed by mid-stack, while it remains causally active at every depth (Section 4.6). The encoder therefore settles the decoder's primary input before it has finished computing per-peak chemical detail. A possible interpretation is that the latent token accumulates a coarse spectrum summary that is sufficient for the decoder to begin autoregressive generation, while

the deeper encoder layers continue refining the per-peak stream that the decoder cross-attends to at each decoding step. The fact that the layer-8 emergent features arise at the same depth at which the latent token has already stabilised is consistent with this separation of function: the deepest encoder computation is directed at per-peak refinement, not at the spectrum-level summary.

5.1.3 Two early-layer exceptions and what they reveal

Immonium detection ($F1_{dom} = 0.777$ at layer 2, the highest of any fragment concept) and peptide length (best $F1_{dom}$ at layer 2 across all length concepts) both peak at the shallowest layer and decline with depth. Both anomalies are consistent with, rather than counter to, the depth-staging account: each is a property the encoder can read almost directly from the input representation before contextual processing begins.

Immonium ions occupy fixed, low-m/z positions characteristic of each residue, directly accessible from the per-peak sinusoidal m/z embedding. Peptide length is a spectrum-level property fixed by the precursor before any peak-to-peak attention occurs. The encoder represents each property at the depth where its information is first available; early for inputs requiring no cross-peak integration, late for those that do. The two anomalies are therefore the sharpest evidence for the depth-staging principle, not exceptions to it: they identify the floor of the staging hierarchy, the properties whose computation is already complete at layer 2.

5.1.4 Detection versus use

Because the SAE preserves over 99.5% of the model's sequencing loss at every layer (Section 4.4), causal ablations are informative about the model rather than about reconstruction artefacts. The results reveal a dissociation that a correlational analysis cannot; the sharpest detector in the entire evaluation and the strongest causal signal correspond to different features and different concepts.

The layer-2 immonium feature (F2666, $F1_{dom} = 0.777$) has no detectable causal effect when ablated ($mag_z = +0.02$). The layer-2 y-ion feature group produces $\Delta CE = +0.102$ when ablated despite the best y-ion feature reaching only $F1_{dom} = 0.283$ at the same layer. The dissociation has a chemical explanation: immonium ions attest residue presence but carry no positional information for the decoder; y-ions define the mass ladder from which residue-by-residue sequence generation proceeds, and their positional information is what the decoder most immediately needs.

At layer 8, the `is_first_ion` group ($sel_z = +15.1$, $mag_z = +20.9$) provides the strongest double-significant causal result: a dedicated terminal-ion detector, where the best feature (F296) simultaneously leads on `position_Cterm`, consistent with a representation encoding the anchor point from which the decoder begins decoding. The three resulting categories; detected and causally used (`is_first_ion` at layer 8), detected but causally silent (immonium at layer 2), and used but distributed across many features so that no small group is individually necessary (Section 5.2) together show that feature interpretability for a scientific model must be assessed on both axes. A ranking by $F1_{dom}$ alone is silent about functional necessity.

5.1.5 Sign-reversed causal effects and redundant encoding

A subset of layer-8 concepts shows statistically significant selectivity but a negative ΔCE: ablating the top features for `cleaves_C_to_Asp` ($sel_z = +5.98$, $\Delta CE = -0.017$) and `covers_N` ($sel_z = +2.67$, $\Delta CE = -0.018$) improves cross-entropy rather than degrading it. These effects are real in the sense that they are concept-specific, but the standard necessity interpretation does not apply.

A plausible account is that the same chemical information is encoded along multiple pathways simultaneously. When one group of features is ablated, the concept-relevant signal remains accessible through other pathways, and the net effect of removing the ablated group is to eliminate a weakly redundant or noisier copy of a concept already encoded more faithfully elsewhere. This interpretation is consistent with the broader redundant-encoding picture described in Section 5.2. It carries a practical implication: a correlational atlas built on $F1_{dom}$ identifies which features encode a concept, but it cannot determine which copy is the faithful one and which the noisy redundant. Causal ablation discriminates the two, since only a noisy copy would produce a CE improvement upon removal.

5.1.6 A structural limitation of correlation-based ranking

The detection-versus-use dissociation is not anecdotal; it follows from how $F1_{dom}$ (Eq. 3.5.6) interacts with the chemistry of the domain. The dominance term penalises a feature for firing on concept-negative tokens. This is appropriate when concepts are near-independent, but fragment-ion concepts are structurally coupled by construction: a y-ion peak is simultaneously a matched peak, covers specific residues, and falls in a particular m/z band, so a genuine y-ion detector incurs a dominance penalty for exactly the firings its concept entails. The more a concept co-occurs with others, as backbone-ion concepts almost always do, the more its true detectors are demoted in the ranking, irrespective of their causal importance to the model. The under-ranking of y-ion features relative to the immonium feature is the visible symptom.

Two consequences follow. Within this evaluation, the causal ablation experiments are indispensable rather than confirmatory: an atlas built on $F1_{dom}$ alone would have ranked immonium features above y-ion features and inverted the true picture of the model's reliance. More broadly, any selectivity-based SAE evaluation in a domain whose observables are structurally coupled inherits the same bias. Correlational ranking should be treated as a source of hypotheses for causal testing rather than as a direct measure of feature function.

5.2 Limitations

5.2.1 Annotation coverage

The peak-match rate against the theoretical fragment ion set is 38% across the full evaluation corpus. The remaining 62% of peaks receive the `is_noise_peak` label and structural concept labels (mass region, intensity, precursor context, peptide length), but carry no ion-type, cleavage, or PTM localisation labels. Features that fire preferentially on these chemically unlabelled peaks are invisible to the F1 evaluation: they may be perfectly interpretable in terms of spectral physics or instrumental artefacts, but the current registry cannot confirm it. Chemically unlabelled peaks contribute to feature firing-rate statistics but cannot activate any ion-type or chemical concept, which means the dead-feature rate and the Gini coefficient are both inflated relative to what they would be if all peaks were chemically labelled. The practical consequence is that the fraction of features with at least one significant concept association is a lower bound on the true fraction of interpretable features, not an estimate of it.

5.2.2 Effect of redundant encoding on causal ablation

The causal ablation experiments test a specific and limited form of necessity: whether ablating the top-10 features for a concept, while leaving all other features intact, degrades the model's performance on concept-positive spectra. This design is susceptible to compensation. If the model represents a concept redundantly across many features, the ablation of the top-10 may be absorbed by the remaining features

with negligible effect on the output. The sign-reversed effects for some of the features at layer 8 suggest that compensation is occurring for at least some concepts.

5.2.3 Single model, single checkpoint

All results are reported for a single InstaNovo checkpoint trained on the ProteomeTools dataset. Whether the observed features generalise across model checkpoints during training or across models trained on different datasets is unknown. The SAE literature has shown that features can be substantially dataset-dependent (Bussmann et al., 2024), and some of the layer-8 cleavage and PTM features may reflect dataset-specific statistical patterns in the nine-species benchmark rather than general properties of the model's computation.

5.3 Future Directions

5.3.1 LLM-assisted automatic interpretation of unlabelled features

The 62% of peaks that carry no ion-type or chemical concept label from the current registry are not uninformative; they are peaks for which the current annotation pipeline has no chemical theory. An automated interpretation pipeline using a large language model, in the style of the approach developed for InterPLM (Simon and Zou, 2025), could provide qualitative descriptions of the unlabelled features by inspecting their top-activating tokens: the m/z values, intensities, peptide sequences, and precursor properties of the peaks that most strongly activate each feature. Features that systematically fire on high-m/z peaks in multiply-charged spectra, or on the noise peaks immediately following an intense y-ion, would be identifiable by inspection even if no formal concept label captures them. This would substantially increase the fraction of interpretable features and reduce the lower-bound bias in the concept association statistics.

5.3.2 Extending the annotation registry

The 50-concept registry covers the most common and chemically well-understood properties of tryptic fragment ions. Several extensions would increase its coverage without sacrificing the first-principles grounding that distinguishes it from database-dependent annotation.

Ion-series completeness concepts, labelling whether a peak represents a complete, unbroken run in the b- or y-ladder versus an isolated fragment, would test whether the encoder represents the structural coherence of the ion series, not just the identity of individual ions. Charge-site concepts, which identify ions whose fragmentation patterns reflect the position of the mobile proton, would connect the SAE features to the fragmentation chemistry literature (Steen and Mann, 2004). Each of these extensions is computable from first principles and would extend the coverage of the registry without introducing database dependence.

5.3.3 Causal steering and sequence generation control

The causal ablation experiments establish that certain feature groups are necessary for accurate sequencing. The complementary question is whether *activating* those features artificially, in the style of the feature steering experiments in InterProt (Adams et al., 2025), can control the model's predictions in interpretable ways. If the y-ion features at layer 2 are functionally essential for y-ion-based sequencing, then suppressing them should cause the model to rely more on b-ions; and if the cleavage-specificity features at layer 8 encode the model's knowledge of tryptic cleavage sites, then boosting them should

bias predictions toward sequences that end in lysine or arginine. Successful steering experiments of this kind would confirm the causal story established by ablation and would open a path toward controlled sequence generation, a capability that is of direct practical interest for antibody engineering and other protein design applications where the target sequence has known structural constraints.

5.3.4 Cross-model comparison

The InstaNovo encoder is one instance of a transformer architecture for mass spectrometry. Casanovo (Yilmaz et al., 2024) uses a similar encoder-decoder structure trained on a different dataset with a different training objective. Whether the features identified in this project are universal properties of transformer-based *de novo* sequencing models or are specific to InstaNovo is an open question. A cross-model comparison using the same SAE architecture and annotation registry on both models would determine which features are generic to the task and which are model-specific. Universal features would be candidates for the mechanistic building blocks of *de novo* sequencing; model-specific features would reveal how different training procedures and datasets lead to different internal solutions to the same problem.

5.4 Conclusion

Beyond InstaNovo, this work shows that the core tools of mechanistic interpretability generalise from language models to transformers used in *de novo* sequencing. The per-peak annotation pipeline, deriving all concept labels from first principles rather than from external databases, provides a template for applying SAE-based interpretability to other domain-specific scientific models where the token semantics are physically grounded and per-token causal structure can be computed analytically. The dissociation between correlational and causal feature importance is unlikely to be specific to mass spectrometry; it is a property of overparameterised models with redundant encoding, and it applies wherever SAEs are used to interpret features in models that have learned to solve a well-defined physical or chemical prediction task.

The broader promise of this work is the possibility of reading out what a scientific model has learned in terms that a domain expert can evaluate. A proteomics researcher who asks why InstaNovo failed to sequence a particular peptide can now point to a specific encoder layer, a specific feature group, and a specific chemical concept whose representation the model appears to have missed or miscalibrated. That level of mechanistic accountability, between the model's internal computation and the chemistry it is supposed to represent, is what interpretability for scientific discovery ultimately requires.

Acknowledgements

I thank my supervisors at InstaDeep; Mr Jeroen Van Goey, Ms Jemma Daniel, and Mr Konstantinos Kalogeropoulos for their patient and constructive guidance through every methodological shift of this research. I am grateful to the African Institute for Mathematical Sciences, South Africa and its funders for the academic environment in which this work was undertaken. I also extend my gratitude to Google DeepMind for their support through the Google DeepMind Scholarship.

References

Adams, E., Bai, L., Lee, M., Yu, Y., and AlQuraishi, M. From mechanistic interpretability to mechanistic biology: Training, evaluating, and interpreting sparse autoencoders on protein language models. bioRxiv preprint, 2025.

Aebersold, R. and Mann, M. Mass-spectrometric exploration of proteome structure and function. *Nature*, 537(7620):347–355, 2016. doi: 10.1038/nature19949.

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Bittremieux, W. spectrum_utils: A Python package for mass spectrometry data processing and visualization. *Analytical Chemistry*, 92(1):659–661, 2020. doi: 10.1021/acs.analchem.9b04884.

Bittremieux, W., Levitsky, L., Pilz, M., Sachsenberg, T., Huber, F., Wang, M., and Dorrestein, P. C. Unified and standardized mass spectrometry data processing in Python using spectrum_utils. *Journal of Proteome Research*, 22(2):625–631, 2023. doi: 10.1021/acs.jproteome.2c00632.

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.

Bricken, T., Templeton, A., Batson, J., Chen, B., Jermyn, A., Conerly, T., Turner, N., et al. Towards monosemanticity: Decomposing language models with dictionary learning. Transformer Circuits Thread, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Bussmann, B., Leask, P., and Nanda, N. BatchTopK sparse autoencoders. arXiv preprint, 2024. arXiv:2412.06410.

Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., Grosse, R., McCandlish, S., Kaplan, J., Amodei, D., Wattenberg, M., and Olah, C. Toy models of superposition. Transformer Circuits Thread, 2022. URL https://transformer-circuits.pub/2022/toy_model/index.html.

Elias, J. E. and Gygi, S. P. Target-decoy search strategy for increased confidence in large-scale protein identifications by mass spectrometry. *Nature Methods*, 4(3):207–214, 2007. doi: 10.1038/nmeth1019.

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

Frank, A. and Pevzner, P. PepNovo: de novo peptide sequencing via probabilistic network modeling. *Analytical Chemistry*, 77(4):964–973, 2005. doi: 10.1021/ac048788h.

Gao, L., la Tour, T. D., Tillman, H., Goh, G., Troll, R., Radford, A., Sutskever, I., Leike, J., and Wu, J. Scaling and evaluating sparse autoencoders. arXiv preprint, 2024. arXiv:2406.04093.

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.

Huben, R., Cunningham, H., Smith, L., Ewart, A., and Sharkey, L. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations*, volume 2024, pages 7827–7845, 2024.

Jain, S. and Wallace, B. C. Attention is not explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 3543–3556, 2019. doi: 10.18653/v1/N19-1357.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023. doi: 10.1126/science.ade2574.

Ma, B., Zhang, K., Hendrie, C., Liang, C., Li, M., Doherty-Kirby, A., and Lajoie, G. PEAKS: powerful software for peptide de novo sequencing by tandem mass spectrometry. *Rapid Communications in Mass Spectrometry*, 17(20):2337–2342, 2003. doi: 10.1002/rcm.1196.

Makhzani, A. and Frey, B. k-Sparse autoencoders. *arXiv preprint arXiv:1312.5663*, 2013.

Marks, S., Rager, C., Michaud, E. J., Belrose, N., Sharkey, L., and Mueller, A. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2403.19647.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022. arXiv:2202.05262.

Muth, T. and Renard, B. Y. Evaluating de novo sequencing in proteomics: already an accurate alternative to database-driven peptide identification? *Briefings in Bioinformatics*, 19(5):954–970, 2018. doi: 10.1093/bib/bbx033.

Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M., and Carter, S. Zoom in: An introduction to circuits. *Distill*, 2020. doi: 10.23915/distill.00024.001. https://distill.pub/2020/circuits/zoom-in.

Olshausen, B. A. and Field, D. J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996. doi: 10.1038/381607a0.

Park, K., Choe, Y. J., and Veitch, V. The linear representation hypothesis and the geometry of large language models. arXiv preprint, 2023. arXiv:2311.03658.

Rajamanoharan, S., Conmy, A., Smith, L., Lieberum, T., Varma, V., Kramár, J., Shah, R., and Nanda, N. Jumping ahead: Improving reconstruction fidelity with JumpReLU sparse autoencoders. arXiv preprint, 2024. arXiv:2407.14435.

Räuker, T., Ho, A., Casper, S., and Hadfield-Menell, D. Toward transparent ai: A survey on interpreting the inner structures of deep neural networks. In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 464–483, 2023. doi: 10.1109/SaTML54575.2023.00039.

Sadygov, R. G., Cociorva, D., and Yates, J. R. Large-scale database searching using tandem mass spectra: looking up the answer in the back of the book. *Nature Methods*, 1(3):195–202, 2004. doi: 10.1038/nmeth725.

Simon, E. and Zou, J. InterPLM: discovering interpretable features in protein language models via sparse autoencoders. *Nature Methods*, 22(10):2107–2117, 2025. doi: 10.1038/s41592-025-02836-7.

Steen, H. and Mann, M. The ABC's (and XYZ's) of peptide sequencing. *Nature Reviews Molecular Cell Biology*, 5(9):699–711, 2004. doi: 10.1038/nrm1468.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3319–3328. JMLR.org, 2017.

Templeton, A., Conerly, T., Marcus, J., Lindsey, J., Bricken, T., et al. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. Transformer Circuits Thread, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html.

Tran, N. H., Zhang, X., Xin, L., Shan, B., and Li, M. De novo peptide sequencing by deep learning. *Proceedings of the National Academy of Sciences*, 114(31):8247–8252, 2017. doi: 10.1073/pnas.1705691114.

Tran, N. H., Qiao, R., Xin, L., Chen, X., et al. Deep learning enables de novo peptide sequencing from data-independent-acquisition mass spectrometry. *Nature Methods*, 16(1):63–66, 2019. doi: 10.1038/s41592-018-0260-3.

Vig, J. A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–42, 2019.

Vig, J., Madani, A., Varshney, L. R., Xiong, C., Socher, R., and Rajani, N. F. Bertology meets biology: Interpreting attention in protein language models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YWtLZvLmud7.

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, 2022.

Wang, Y., Liang, Z., Ling, T., Chang, C., Yang, T., Xie, L., and He, Y. Transforming de novo peptide sequencing by explainable ai. ResearchSquare preprint, 2024.

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

Wiegreffe, S. and Pinter, Y. Attention is not not explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 11–20, 2019. doi: 10.18653/v1/D19-1002.

Yilmaz, M., Fondrie, W. E., Bittremieux, W., Melendez, C. F., Nelson, R., Ananth, V., Oh, S., and Noble, W. S. Sequence-to-sequence translation from mass spectra to peptides with a transformer model. *Nature Communications*, 15:6427, 2024. doi: 10.1038/s41467-024-49731-x.