MASARYK UNIVERSITY

FACULTY OF INFORMATICS

Predicting molecular structures from multi-stage MSn fragmentation trees using graph neural networks and DreaMS foundation model

Master's Thesis

FILIP JOZEFOV

Brno, Spring 2025

MASARYK UNIVERSITY

FACULTY OF INFORMATICS

Predicting molecular structures from multi-stage MSn fragmentation trees using graph neural networks and DreaMS foundation model

Master's Thesis

FILIP JOZEFOV

Advisor: Mgr. Aleš Křenek, Ph.D.

Department of Machine Learning and Data Processing

Brno, Spring 2025


FACULTAS ARTIS INFORMATICÆ
UNIVERSITAS MASARYKIANA

Declaration

Hereby I declare that this thesis is my original authorial work, which I have worked out on my own. All sources, references, and literature used or excerpted during elaboration of this work are properly cited and listed in complete reference to the due source.

During the preparation of this thesis, I used the AI tools to improve my writing style and for faster code writing. I declare that I used these tools in accordance with the principles of academic integrity. I checked the content and took full responsibility for it.

Filip Jozefov

Advisor: Mgr. Aleš Křenek, Ph.D.

Acknowledgements

I want to sincerely thank my supervisor, Mgr. Aleš Křenek, Ph.D., for trusting me with this diploma thesis topic and for his steady support.

I'm deeply grateful to the research group led by Mgr. Tomáš Pluskal, Ph.D., at the Institute of Organic Chemistry and Biochemistry of the CAS in Prague. Without their openness, guidance, and generosity in welcoming me into their lab, this work simply wouldn't have been possible.

I am also very thankful to Dr. Ing. Josef Šivic at the Czech Institute of Informatics, Robotics and Cybernetics. His insights into the machine learning aspects of the project were invaluable and gave me solid ground to stand on when I needed it most.

My sincere thanks go to Dr. Corinna Brungs from the University of Vienna. Her deep knowledge of mass spectrometry and our thoughtful discussions helped me grasp its fundamentals, and allowed me to understand the broader scientific context of the data I was working with.

I am deeply grateful to my family for their constant and unconditional support. In moments of difficulty, their encouragement gave me the strength to persist and pushed me to go even further.

Finally, I would like to thank my friends for being a source of balance and perspective. Their presence helped me unwind, recharge, and stay grounded throughout this journey.

This work was supported by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254), provided through projects OPEN-33-11 and OPEN-32-14.

Abstract

Metabolomics seeks to identify and understand the small molecules that drive biological systems, yet a large portion of these molecules remain unknown. Tandem mass spectrometry (MS/MS) is the primary technology for identifying metabolite structures at scale, but the process of interpreting spectra is often ambiguous and incomplete. As a result, the majority of MS/MS data remains unannotated, leaving much of the "dark metabolome" unexplored. While machine learning has made progress in molecular annotation, current models typically rely on a single stage of fragmentation (MS2), limiting their ability to capture deeper structural information.

We propose a new direction: incorporating multi-stage MSn fragmentation data, where molecules are fragmented in successive rounds to reveal deeper structural layers, to enhance molecular annotation. To investigate this, we develop the first neural network models trained on MSn spectra, combining graph neural networks with DreaMS, a foundation model for mass spectral embeddings. We benchmark these models on two key tasks: molecular structure retrieval from candidate sets and de novo structure generation.

Our results show that multi-stage fragmentation improves retrieval accuracy by up to 10x compared to MS2 alone. Furthermore, deeper MSn levels produce richer and more informative spectral representations, as confirmed through centered kernel alignment (CKA) analysis.

To support continued progress, we introduce MassSpecGymMSn, the first open benchmark for MSn-based molecular annotation. It includes 16,476 fragmentation trees (up to MS5), along with preprocessing tools, and will be made publicly available to advance research in this area.

Keywords

deep learning, mass spectrometry, benchmark, multi-stage mass spectrometry, dataset, metabolomics

Contents

List of Tables

List of Figures

1 Introduction

Metabolomics is the comprehensive study of small molecules (metabolites) present in biological and environmental samples. A primary goal of metabolomics is to accurately determine the molecular structures of these metabolites, as such information profoundly advances our understanding of biochemical pathways [1], disease mechanisms [2], environmental interactions [3], and drug development [4]. Mass spectrometry (MS), particularly tandem mass spectrometry (MS2), has become a foundational tool in metabolomics research due to its unmatched sensitivity and high-throughput capabilities [5, 6]. In MS2, molecules are fragmented and the resulting fragment ions are measured, producing characteristic spectral "fingerprints" that enable structural elucidation through comparison with reference databases or expert interpretation.

Despite the proven capabilities of MS2, accurately interpreting mass spectra remains highly challenging. Fragment ion patterns can be ambiguous, subtle structural differences between isomers are easily overlooked, and the majority of MS2 spectra remain uninterpreted due to the intrinsic complexity of the data. Recent studies estimate that only about 1.8% of spectra in untargeted metabolomics experiments can be confidently annotated [7], often referred to as the "dark metabolome". Furthermore, it is estimated that over 99% of the plant phytochemical space remains unexplored [8], highlighting the vast potential for discovery in metabolomics. Efforts over several decades to improve metabolite identification through computational methods, including both rule-based [9] approaches and, more recently, machine learning techniques [10, 11] have been significantly hindered by the lack of standardized datasets and common evaluation protocols. Researchers frequently rely on proprietary or fragmented spectral libraries, limiting the development and fair comparison of new methodologies. This shortage of publicly accessible benchmark datasets [12] has posed a substantial barrier to innovation within computational metabolomics.

Another significant limitation has been the historical reliance on single-stage fragmentation data (MS2). Standard metabolomics analyses rarely exploit the full potential of modern instrumentation, which allows multiple sequential rounds of fragmentation [5], termed multi-

stage MS MSn, extending beyond the typical MS2. MSn data offer deeper structural insights by hierarchically fragmenting molecules, where each stage generates fragments that can themselves be fragmented, producing detailed fragmentation trees. These trees can reveal molecular substructures that often remain hidden when relying solely on a single fragmentation event [13].

However, despite their theoretical advantages, multi-stage MS data have historically been scarce and underutilized in untargeted metabolomics [5]. The main barriers have been the increased complexity, time, and resource demands associated with acquiring MSn experiments. Moreover, public repositories predominantly contain MS2 spectra [14], with virtually no multi-stage (MS3 or higher) data available, limiting the scope of data-driven research. To address these challenges, in Pluskal lab, we developed a high-throughput, high-quality acquisition pipeline for MSn spectra, dramatically improving the efficiency and scalability of multi-stage fragmentation data collection.

To investigate the value of multi-stage mass spectrometry MSn data, this thesis introduces first deep learning approaches specifically tailored for multi-stage mass spectrometry. Neural networks have demonstrated significant potential across various scientific domains, including biology [15], chemistry [16], and computational sciences [17], due to their ability to capture intricate patterns within data. Central to this thesis is the development and deployment of graph neural networks [18], which model the hierarchical nature of MSn data. Additionally, the thesis employs DreaMS [19], a transformer-based foundation model pretrained on extensive MS2 spectra repositories, to generate learned embeddings for spectra at all fragmentation stages. Integrating DreaMS within multi-stage MS workflows marks a extension of foundation models beyond single-stage fragmentation data, demonstrating their significant potential in enriching structural information from mass spectrometry.

To facilitate neural network applications and foster reproducible research, this thesis introduces MassSpecGymMSn, the first large-scale, open-access benchmark dataset explicitly designed for multi-stage MSn analysis. MassSpecGymMSn contains 183,294 spectra spanning fragmentation levels up to MS5, covering 14,008 unique compounds, the majority of which have not been previously measured. This bench-

mark substantially expands the available mass spectrometry data, offering machine learning ready MSn data that were previously unavailable to researchers. In addition, MassSpecGymMSn defines standardized challenges for molecule identification and de novo molecular structure generation, with carefully designed dataset splits to prevent information leakage.

Finally, by combining benchmarking, statistical testing, and neural networks, this thesis provides the first study connecting multi-stage MSn data with deep learning models and quantitatively evaluates the value of multi-stage MSn data compared to conventional MS2 approaches.

1.1 Contributions

Our work presents the following key advancements toward establishing the combination of multi-stage MSn data and artificial intelligence as a new source of discovery in metabolomics:

- **Development of the first neural network for multi-stage MSn mass spectra:**
  We introduced graph neural networks (GNNs) designed specifically for multi-stage MSn data. These models represent spectra trees as hierarchical fragmentation structure, resulting in a dramatic improvement of nearly 10x in molecular identification performance compared to traditional MS2-based models.

- **Introduction of the first open, large-scale MSn dataset, MassSpecGymMSn:**
  This benchmark contains over 183,000 spectra spanning fragmentation stages up to MS5, covering ~14,000 unique compounds, greatly expanding publicly accessible mass spectrometry data previously limited primarily to MS2.

- **One of the first statistical analyses supporting MSn over MS2:**
  We provide some of the first quantitative and statistical evidence that deeper MSn stages offer distinct advantages over MS2. Statistical analyses show that each additional fragmentation stage uncovers significant structural relationships not present in MS2 alone. Further

examination of model internal representations reveals that models trained on full MSn trees develop richer, higher-dimensional feature spaces compared to those trained solely on MS2 data.

- **Foundation model integration beyond MS2:**
  For the first time, we showed that DreaMS, a foundation model trained solely on MS2 spectra, also produces meaningful embeddings for MSn data. Without ever seeing multi-stage spectra during training, its embeddings still cluster MSn fragmentation stages in a hierarchy that mirrors true substructure relationships. Downstream models built on these embeddings consistently outperformed conventional methods, underscoring both the surprising generalizability of foundation models and their ability to enhance structural insights in mass spectrometry.

- **Proven compatibility and extensibility of MSn Data:**
  In our MassSpecGymMSn, both MS2 and MSn data can be used together seamlessly. The benchmark is also designed for easy expansion, and we plan to add over two million spectra spanning multiple collision energies in both positive and negative ion modes in the near future.

- **Facilitating broad community adoption:**
  By releasing MassSpecGymMSn and its associated tools as standardized, reproducible open-source resources, we provide a benchmark for fair method comparisons. This significantly lowers the barrier to entry into computational metabolomics, promoting widespread adoption and fostering collaboration across multiple research communities at the intersection of mass spectrometry and machine learning.

2 Theoretical background

2.1 Fundamentals of mass spectrometry

2.1.1 From MS1 to MS2, foundations of mass spectrometry

Mass spectrometry (*MS*) is a powerful analytical technique that offers **high sensitivity** and **specificity**, making it a vital tool in areas such as drug discovery, environmental analysis, and proteomics. By accurately measuring the mass-to-charge ratio (m/z) and ion abundance, *MS* provides unique molecular fingerprints that are essential to identify and quantify compounds in complex mixtures. In this context, the value of m/z is defined as the mass of an ion divided by its charge, with ion masses reported in daltons (Da), where one dalton is 1/12 of the mass of a carbon-12 atom. This level of precision is complemented by the inherent simplicity and speed of mass spectrometry (see Figure 2.1). Unlike techniques such as X-ray crystallography, which often require extensive sample preparation and prolonged data collection [20], *MS* can be adapted for high-throughput analysis, providing rapid and exact molecular identification.

Before analysis, sample preparation is essential to ensure both high-quality and reliable results. Typically, the sample consists of a complex mixture of analytes, from which only a subset is of primary interest. Techniques such as chromatography are commonly used to separate these mixtures, enhance sensitivity and selectivity, and perform sample cleanup [22].

Following preparation, the analytes are introduced into a mass spectrometer (see Figure 2.2), which comprises three main components: an *ionization source*, a *mass analyzer*, and an *ion detection system*. The process in a mass spectrometer begins with converting analytes from their native state to gas-phase ions using soft ionization techniques such as *Electrospray Ionization* (ESI) and *Matrix-Assisted Laser Desorption/Ionization* (MALDI). These methods preserve the predominantly intact molecular ion by minimizing fragmentation [23], which is essential for subsequent analysis. In this thesis, our data were acquired using ESI ionization [5], whereby an electrical charge is im-

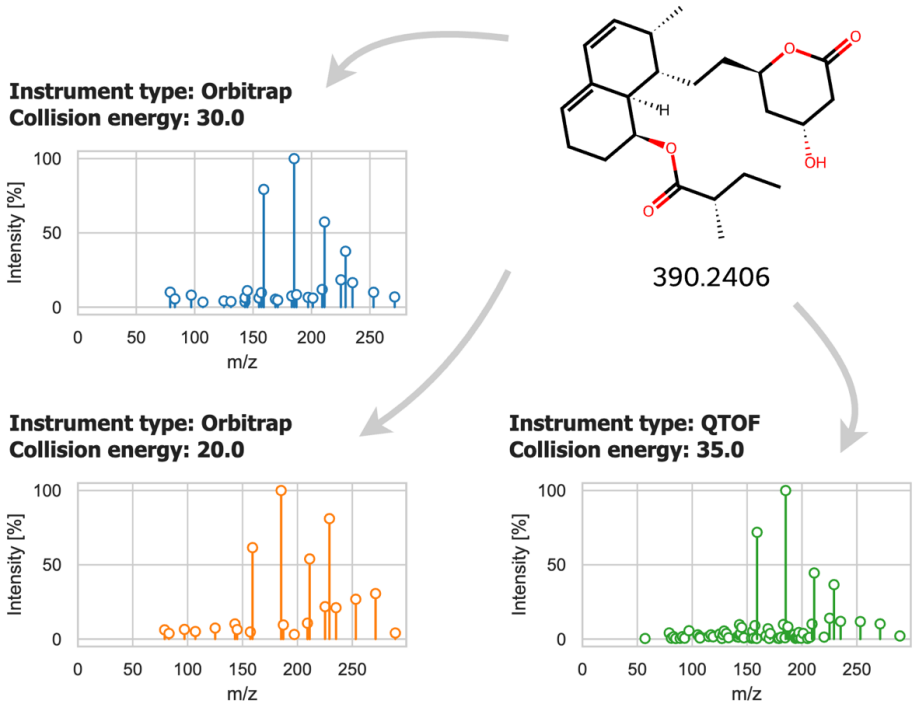

Figure 2.1: The figure shows the molecular structure of Mevastatin, a compound used in the prevention of cardiovascular diseases [21], along with its exact molecular weight. Adjacent to it are three distinct mass spectra, each generated from the same molecule under different experimental conditions. The spectra differ significantly, showing how experimental conditions affect fragmentation and spectral profiles.

parted to the analyte molecules, facilitating their manipulation within an electromagnetic field.

Following ionization, the ions are introduced into the first mass analyzer (MS1), where they are separated based on their m/z ratios. Instruments such as time-of-flight (TOF), Orbitrap, and quadrupole analyzers are commonly used at this stage. Although each analyzer has its own strengths and limitations [24, 25], they all rely on electromagnetic principles, which typically deflect ions [26] according to Fleming's left-hand rule, to achieve separation. The primary purpose of MS1 is to generate a spectrum of intact ions and select those

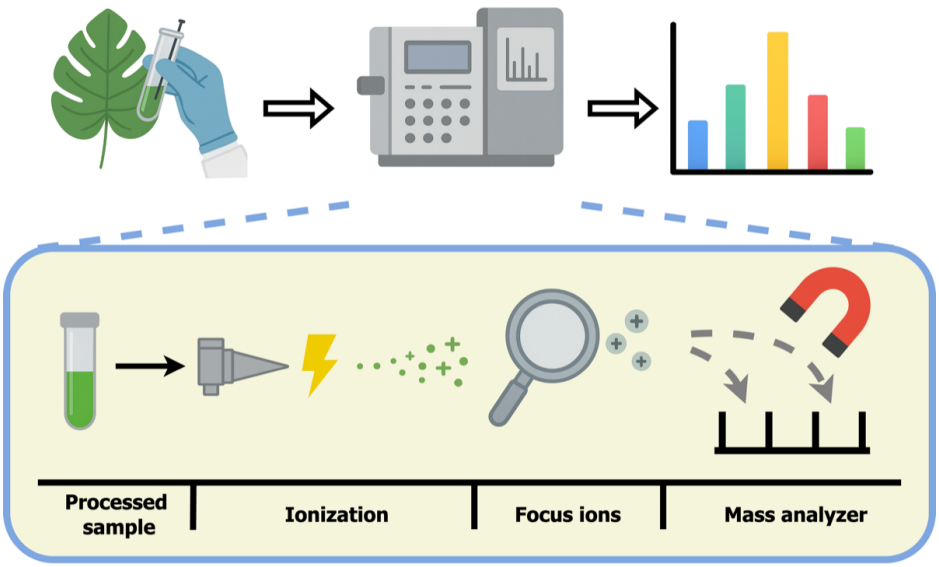

Figure 2.2: The upper part of the image: a sample is collected from an environmental source (e.g., a leaf) and introduced into the mass spectrometer, where its unique m/z is recorded as a mass spectrum. The lower part provides a magnified view of the instrument's core components. It depicts the ionization source, where the sample is transformed into gas-phase ions, the ion optics that focus and direct these ions, and the mass analyzer that separates them based on their m/z values before detection

precursor ions of interest. However, MS1 only provides mass measurements of intact ions and does not offer the structural details needed to resolve isomeric compounds or elucidate complex molecular architectures [27].

To address the limitations of mass measurement alone, the selected precursor ions are subsequently transferred to the second mass analyzer (MS2) for controlled fragmentation. In MS2, the isolated precursor ion is subjected to collision-induced dissociation (CID), where it collides with an inert gas (such as helium or nitrogen) within a collision cell. This collision converts kinetic energy into internal energy, causing the ion to break into smaller, characteristic fragments [28]. The reproducibility of CID ensures that the resulting fragmentation

patterns provide reliable fingerprints for structural elucidation and isomer differentiation.

The final mass spectrum, whether from MS1 or MS2, displays m/z values along the horizontal axis, serving as unique identifiers for the ions, and their corresponding intensities along the vertical axis, which reflect ion abundance. Achieving high resolution and sensitivity at each stage is crucial, as it allows for the detection of subtle differences in fragment ions [29]. This precision effectively narrows the range of potential structural candidates, consequently enhancing the reliability of compound identification.

For clarity, this chapter focuses on the core functions of the MS1 and MS2 stages in tandem MS workflows. While MS1 is responsible for the generation and selection of intact ions based on their m/z ratios, MS2 provides additional structural information through controlled fragmentation. Although a wide variety of mass analyzers, ionization methods, sample preparation techniques, and collision strategies exist, our discussion has been streamlined to emphasize these fundamental principles, recognizing that each component must be carefully optimized according to the specific analytical context [30].

2.1.2 Multi-stage fragmentation mass spectrometry

Multi-stage mass spectrometry (MSn^1) builds on traditional tandem mass spectrometry by iteratively fragmenting ions to reveal **increasingly detailed structural information**. In a typical workflow, a complex sample is first analyzed by MS1, which records the mass spectrum of intact molecular ions. Once sufficient ion signal is acquired, a precursor ion is isolated and fragmented, producing an MS2 spectrum that contains the initial set of product ions. In MSn, selected fragments from MS2 are subsequently isolated and subjected to further fragmentation, generating an MS3 spectrum; this process can be repeated through additional stages (see Figure 2.3). Each successive fragmentation stage provides deeper insights into the molecular fragmentation process and can subsequently lead to better compound identification [13, 31].

However, MS2 alone often leaves gaps in our understanding. Product ions may result from intermediate rearrangements rather than

1. Here, n indicates the number of successive fragmentation stages.

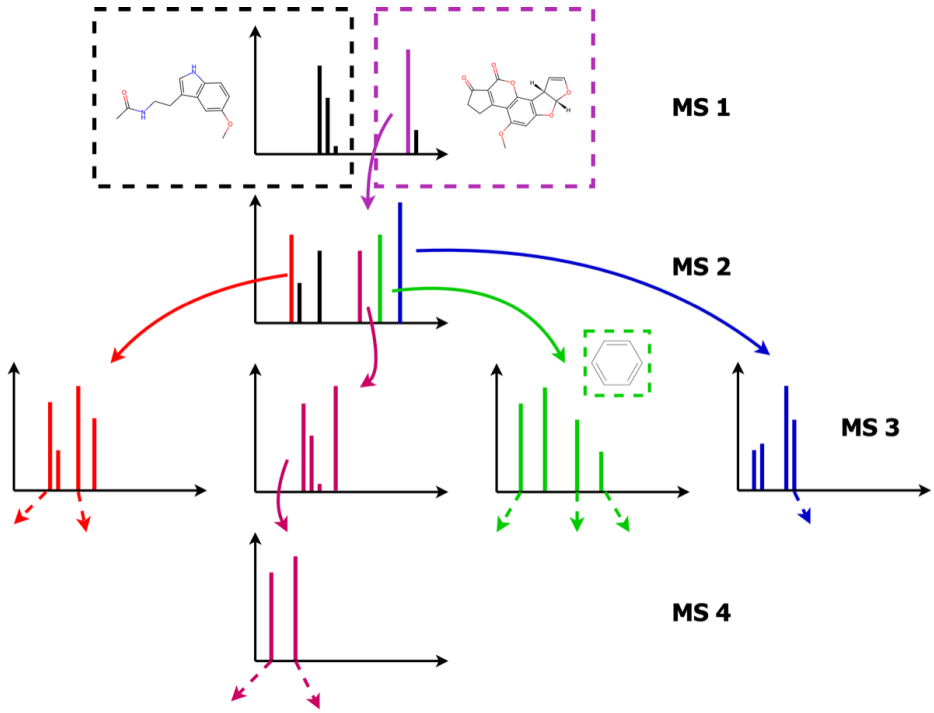

Figure 2.3: In multi-stage fragmentation, the selected precursor peaks are isolated and subjected to fragmentation, generating product ions that can serve as new precursor ions for further fragmentation.

direct fragmentation of the precursor, leading to **unexplained** signals even when the molecular structure is known. Subsequent MSn fragmentation of selected product ions enables the reconstruction of detailed fragmentation pathways by linking each fragment to its originating precursor [5, 32]. This helps establish clearer structural relationships within the molecule. These spectra can be integrated into hierarchical fragmentation trees that reveal how molecular substructures break down across levels. Crucially, MSn remains fully compatible with standard MS2 workflows, extending them by adding deeper layers of structural information. Despite these advantages, over 95% of LC/MS fragmentation studies remain limited to MS2 [13], underscoring the vast potential of broader MSn adoption.

2.1.3 Compound identifications with multi-stage mass spectra

Although MS2 has proven effective, yet MSn introduces an element of flexibility over conventional MS2 for resolving complex structural challenges. Importantly, MSn offers significant advantages for the dereplication of natural products[2][3] by distinguishing closely related isomers, a task that MS2 alone cannot reliably achieve [34]. In MS2, the absence of specific diagnostic ions prevents the clear differentiation between 6-C and 8-C glycosidic flavonoids, even when differences in ion-intensity ratios are observed [13, 31]. Establishing reliable differentiation rules using only MS2 would require the acquisition and computational analysis of an impractically large number of spectra.

In contrast, the MS3 stage reveals clear diagnostic ions that distinguish vitexin from isovitexin due to markedly different C-ring cleavage mechanisms [31]. Remarkably, MS4 data can even provide the precise position of the functional group. Besides facilitating the identification of isomers, MSn has demonstrated its capability in identifying larger, high-mass compounds, where assigning the correct structure becomes combinatorially more challenging [35]. For example, twenty-five citrus flavonoid O-diglycosides, complex molecules with masses above 1000–1500 Da, were accurately identified by comparing experimental MS3 spectra [13, 36]. This clearly demonstrates that relying solely on MS2 spectra can make high-confidence identification of such challenging molecules significantly more difficult.

Decide upon the definitive *MS* level for structural elucidation is often challenging. Fabre et al. [37] demonstrated the power of MSn to probe the fragmentation of flavonoid aglycones. Their study revealed that MS3 data could confirm proposed fragmentation pathways, distinguish characteristic neutral losses linked to specific substructures, and provide enough detail to infer plausible fragment structures. However, for some flavonoid aglycones, the information from MS3 was insufficient to conclusively elucidate the fragmentation mechanisms [13, 37]. In this diploma thesis, we construct of spectral trees up to the

2. Dereplication is the rapid identification of known compounds to avoid unnecessary characterization of previously described molecules.

3. These bioactive compounds, rich in structurally diverse secondary metabolites with significant pharmacological potential, are crucial leads in drug discovery [33]

MS5 level, with the aim of overcoming challenging cases that remain unresolved at lower fragmentation stages.

2.1.4 Algorithmic approach to spectra trees

Despite the relative scarcity of spectra trees, several algorithmic approaches have been developed to enhance spectral annotation by quantifying tree similarity. One prominent method is the construction of *fragmentation trees*, which assign molecular formulas to individual spectral peaks and link them in a hierarchy that reflects possible fragmentation pathways. Fragmentation trees have been extensively applied to MS2 data [9, 38], and extending them to MSn, where additional fragmentation stages offer more detailed insights, is expected to produce trees that are chemically and physically more accurate [39]. Böcker and colleagues further refined these methods for MSn data, showing that including deeper fragmentation levels can reorder up to **25%** of the fragments [40], demonstrating for the first time the significant benefit of MSn for improving tree quality [41].

Fragmentation trees also support *tree alignment algorithms* [6, 42, 43], allowing direct comparison of MS2 and MSn fragmentation patterns within a unified framework. In addition, they serve as crucial inputs for both forward tasks (predicting spectra from molecular structures [44–46]) and reverse tasks (such as de novo molecular generation [9, 10, 47, 48]), see Figure 2.4. This compatibility allows seamless comparison of MS2 and MSn data, while directly benefiting downstream structure prediction workflows.

In parallel, in silico generation approaches have been proposed to address the limited coverage of experimental spectra [49–51], but these are often hard to reproduce, outdated, or commercially restricted, and have not gained wide adoption.

2.1.5 Reference libraries

Spectral libraries are constructed to compile extensive collections of mass spectra that serve as references for identifying unknown compounds [34, 52]. Yet, library matching often results in **low annotation rates**, leaving many biomarkers and compounds uncharacterized, even when using modification-aware algorithms [5, 53]. This shortfall

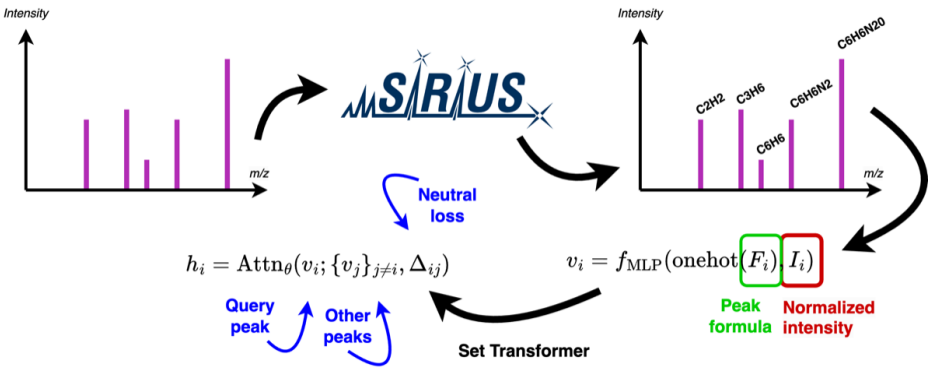

Figure 2.4: DiffMS [10] is the current state-of-the-art model for molecular structure prediction from MS2 data. It uses fragmentation trees from SIRIUS [9] tool to preprocess spectra, which improve predictive accuracy.

is largely due to the *limited availability* of high-quality, open-access databases, and the fact that over **95%** of LC/*MS* fragmentation studies are still performed at the MS2 level [13].

Open-access MS2 libraries such as GNPS [14], MoNA [54], HMDB [55], MassBank [56], NPLIB [57], and MassSpecGym [58] vary in both scope and quality. The largest, GNPS, contains up to **322,000** spectra and about **16,000** unique molecules. However, many spectra and compounds in these databases overlap or are of varying quality. To address this issue, MassSpecGym was developed to aggregate high-quality MS2 data from GNPS, MoNA, and MassBank, as well as a subset of the MS2 dataset used in this thesis. In contrast, proprietary libraries like those from the National Institute of Standards and Technology (NIST) [59], MzCloud [60], and METLIN [61], although more comprehensive (with NIST containing around **52,000** unique compounds), have restricted access, limiting their utility for machine learning applications and broader scientific research.

In this diploma thesis, we focus on MSn data, a challenging domain due to the more complex acquisition process and the limited number of compounds covered. Open-access repositories such as MassBank EU [56] and MoNA [54] offer fewer than **2,000** $MS^{(n>2)}$ spectra. Several smaller datasets have been generated in specific experimental contexts [62–64], but these are scattered, and remain difficult to con-

solidate. The largest closed database, mzCloud [60], includes more than **30,200** unique compounds [5], yet its restricted access and lack of standardization limit its usability for machine learning research.

The challenge of compound identification is magnified by the immense diversity of chemical space, estimated at up to 10^{60} small molecules under 500 Da [65]. While large structure databases like COCONUT [66], ChEMBL [67, 68], PubChem [69], and ZINC [70] cover millions to billions of compounds, most lack corresponding spectral data.

To address the significant shortage of high-quality, open-access MSn data and provide a reproducible, efficient method for generating such spectra, our colleagues led by Dr. Corinna Brungs assembled a database comprising **30,008** unique compounds and **2,350,646** MSn spectra (both merged and individual) [5]. This pioneering dataset was generated in just *23 days* using high-quality acquisitions at multiple collision energies and in both positive and negative ion modes (see Section 4.1). The goal was to expand open-access spectral libraries and demonstrate a fast, reproducible technique for MSn data generation, with all data and workflows made freely available to the community.

Building on this success, in this thesis we introduce **MassSpecGymMSn**, the first machine-learning–ready MSn database, which contains multi-stage fragmentation spectra for **14,008** unique compounds, totaling **183,294** spectra. We accompanied this resource with an open-source preprocessing pipeline and usage guidelines tailored for machine learning practitioners. We propose it as a foundation for in silico generation tasks, with the potential to bridge the gap between annotated spectra and compound identification, and to foster stronger integration between the data science community and metabolomics.

2.2 Deep learning

The perceptron, introduced in 1957 by Frank Rosenblatt, is a binary classifier that laid the groundwork for modern neural networks [71]. Unlike earlier binary models, the perceptron processes numerical inputs, with each connection assigned a weight that can be adjusted based on the data. Drawing inspiration from synaptic plasticity, the biological process where the strength of connections between neurons

changes with experience [72], the perceptron adapts its weights to effectively learn and classify patterns. This adaptive mechanism has been central to developing advanced neural network architectures and learning algorithms, leading to breakthroughs recognized by the Nobel Prize in 2024 [73–75].

Deep learning extends this idea by stacking multiple layers, forming multi-layer perceptron (MLP), and enabling the approximation of complex, non-linear functions [76]. Given a set of input-output pairs, the goal is to learn a parameterized function f_θ that maps inputs $x \in \mathbb{R}^n$ to outputs $y \in \mathbb{R}^m$, where n and m denote the dimensions of the input and output spaces, respectively. The model aims to predict outputs as close as possible to the true targets y, producing $f_\theta(x) = \hat{y} \approx y$ as an approximation.functions [76]. Given a set of input-output pairs, the goal is to learn a parameterized function f_θ that maps inputs $x \in \mathbb{R}^n$ to outputs $y \in \mathbb{R}^m$, where n and m denote the dimensions of the input and output spaces, respectively. The model aims to predict outputs as close as possible to the true targets y, producing $f_\theta(x) = \hat{y} \approx y$ as an approximation.

The function f_θ is typically composed of simpler layers, each applying a linear transformation followed by a non-linear activation function (e.g., ReLU). Formally:

$$\hat{y} = f_\theta(x) = f^{(L)}(\dots f^{(2)}(f^{(1)}(x))) \tag{2.1}$$

$$\text{where} \quad f^{(l)}(x) = \sigma(W^{(l)}x + b^{(l)}) \tag{2.2}$$

Here, L is the total number of layers, $f^{(l)}$ denotes the transformation at layer l. And $W^{(l)}$ and $b^{(l)}$ represent trainable parameters, and σ is the non-linear activation function. Those non-linear activation functions are essential, without them, the network would collapse into a linear model incapable of capturing complex patterns.

Training is necessary because neural networks initially start with random weights, leading to inaccurate predictions. The training process adjusts these parameters to minimize a defined loss function $\mathcal{L}(\theta)$, which quantifies the error between predicted and actual outputs. Optimization is typically performed using stochastic gradient descent, with the backpropagation algorithm efficiently computing gradients by propagating errors backward through the network [75]. Iteratively

updating the parameters allows the network to learn meaningful representations and improve predictive accuracy.

Beyond basic architectures, deep learning includes specialized models tailored for particular data structures and tasks. In this thesis, we utilize *graph neural networks* (*GNNs*) [18] and *Transformers* [77] due to their proven effectiveness for specific modalities relevant to our work.

Graph Neural Networks are a specialized class of neural networks designed to process data represented as graphs. At the core of GNNs is the message passing algorithm, which updates each node's features by aggregating information from its neighboring nodes [78]. This process generates contextually rich representations for nodes and edges, and consequently for the entire graph [79]. In our work, we focus on obtaining graph-level representations of mass spectra trees, where nodes correspond to measured spectra and edges represent the hierarchy of fragmentation events.

Transformers [77] have become a cornerstone of modern AI, powering state-of-the-art models like ChatGPT [80] and Gemini [81]. They have revolutionized sequence modeling by allowing every token in an input sequence to interact directly with every other token. Unlike earlier widely used approaches, such as recurrent or convolutional networks [82–84], that rely on local context or sequential processing, the Transformer employs a global self-attention mechanism. This design enables the efficient capture of long-range dependencies, originally developed for natural language processing tasks, but it has since been successfully applied to areas ranging from computer vision [85] to protein design [15, 86]. In this thesis, we implement Transformer models for molecular structure generation based on mass spectra trees.

3 Machine learning for mass spectrometry

3.1 Foundation models and DreaMS

Foundation models[1] refer to large-scale neural networks pre-trained on large, diverse datasets, which can then be fine-tuned for a wide array of downstream tasks. This paradigm shift moves from developing models tailored for a single specific problem to constructing **general-purpose data representations** that capture underlying patterns across domains. One well-known example today is in natural language processing, where models *BERT* [87] or *GPT* [88] are pre-trained using unsupervised or self-supervised techniques, enabling them to learn rich syntactic and semantic structures directly from raw text without explicit labels. These robust, high-dimensional representations can subsequently be adapted for specific, fine-grained applications with relatively limited additional training data [89] (see Figure 3.1).

Motivated by the success of *foundation models* in other fields, in our lab, Bushuiev et al. developed *DreaMS*, a transformer-based foundation model pre-trained on a vast corpus of unlabeled spectra. The key idea is to learn high-dimensional representations of mass spectra from millions of examples, enabling the model to capture generalizable patterns in fragmentation events. DreaMS is pre-trained to simultaneously predict masked spectral peaks and chromatographic retention orders from raw MS/MS data. Predicting missing peaks requires recognizing when a set of observed fragments implies the presence of a specific hidden fragment, while predicting retention order demands an understanding of molecular features that affect properties such as polarity and size [91]. Through these pre-training tasks, the model learns the underlying structural and fragmentation patterns, yielding rich spectral embeddings that implicitly encode molecular structure.

The final DreaMS model is optimized for spectral similarity, ensuring that spectra from **chemically similar compounds cluster closely** in the latent embedding space. This provides a robust alternative to traditional molecular networking approaches [92], which rely on heuristics and often struggle to resolve subtle structural differences. In

1. Neural networks pre-trained on large, diverse datasets to learn general-purpose representations.

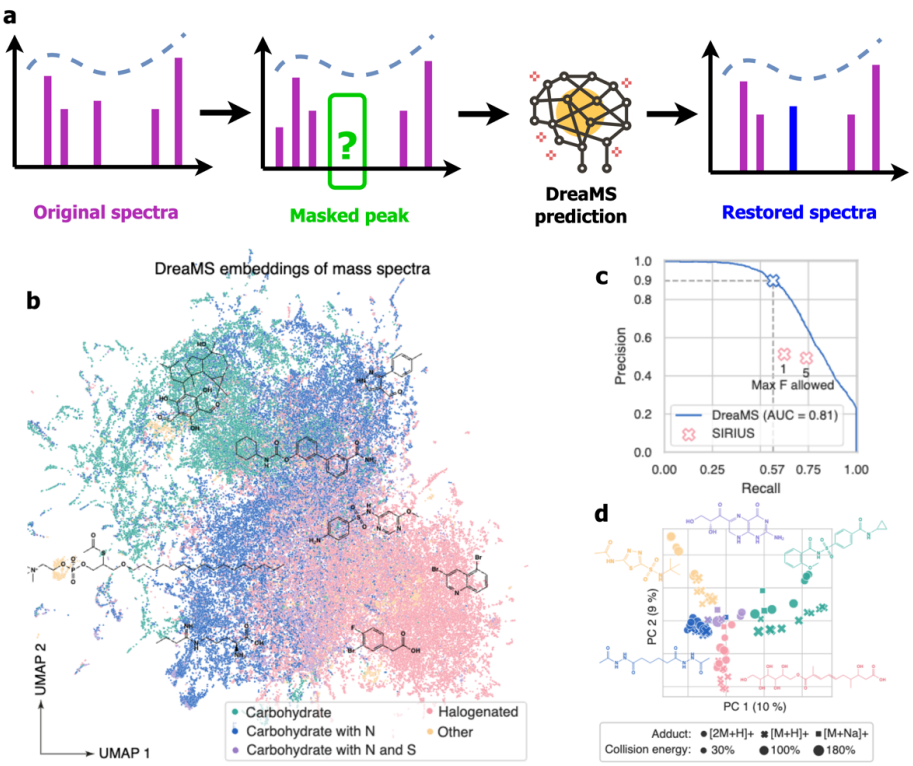

Figure 3.1: (**a**) Self-supervised learning on spectra, where masked spectra serve as templates for DreaMS training. (**b**) UMAP projection of DreaMS embeddings, illustrating the organization of the representation space by molecular formulas. (**c**) DreaMS (blue) outperforms SIRIUS, current state-of-the-art, (pink, under two settings) in detecting fluorinated molecules, achieving nearly two-fold higher precision. (**d**) PCA [90] of selected precursor embeddings demonstrates linear clustering by molecular structure, robust across different ionization adducts and normalized collision energies [19].

contrast, DreaMS leverages learned embeddings for more scalable and accurate untargeted analysis. A notable demonstration of its utility is in the detection of fluorinated compounds, DreaMS, when fine-tuned, outperforms specialized tools like SIRIUS [9] in identifying organofluorines[2], see Figure 3.1. This generalizable framework enhances compound annotation and structure discovery, positioning DreaMS as a strong foundation model for real-world mass spectrometry applications. In this thesis, we adopt it as one of the alternatives for mass spectra encoding.

3.2 Murcko histogram and data splitting

Generalization beyond the training data is essential in machine learning, making data splitting a critical step in model development [76]. In molecular machine learning, datasets are typically partitioned into training and validation sets using methods such as *structure-disjoint splitting*[3], *scaffold-disjoint splitting* [96, 97], or random sampling. However, when working with mass spectrometry data, these conventional splitting strategies may inadvertently lead to **information leakage** due to the *intrinsic* nature of fragmentation events [98].

Mass spectrometry datasets present unique challenges due to *collision-induced dissociation* (*CID*)[4]. During CID, molecules fragment into ions that often generate nearly identical spectra, even when subtle structural differences exist. Additionally, minor variations, such as differences in linker length, can yield similar fragmentation patterns [98]. These factors complicate the identification of truly novel structures in the validation set, potentially leading to models that perform well on validation data yet fail to generalize to new molecular structures (see Figure 3.2).

One prevalent technique for splitting molecular data is *scaffold-disjoint splitting*, typically implemented using *Murcko scaffolds* [99]. This approach reduces a molecule to its core structure by removing

2. Organofluorines a chemically important class that makes up over 30% of modern pharmaceuticals and agrochemicals [93, 94]
3. Based on the first 14 characters of the InChIKey [95], which does not resolve stereoisomers.
4. CID fragments ions by collisions with inert gas molecules, such as helium or nitrogen.

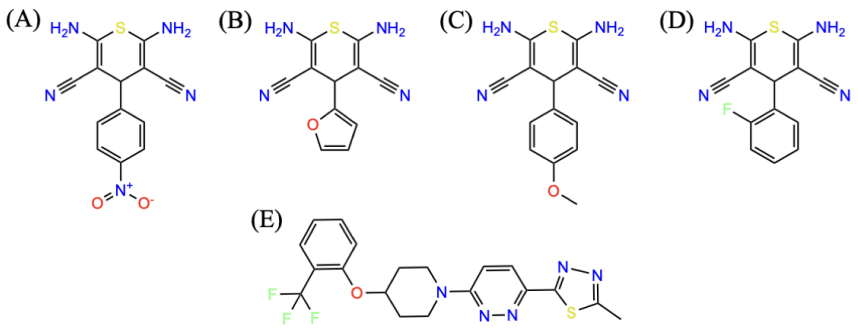

Figure 3.2: Adapted from Roman Bushuiev's thesis [98], this figure illustrates a critical challenge in mass spectrometry data splitting. Molecules **(a)**–**(d)** share similar fragmentation patterns, evidenced by peaks at m/z 177.0223 and 101.0163. Splitting fragments from these nearly identical molecules between training and validation sets may cause a model to learn trivial, non-generalizable features, leading to overfitting. Encountering a new compound **(e)**, which lacks the nitrogenated Thiane ring, the model fails to extrapolate despite shared spectral peaks.

side chains and retaining only the essential ring systems and linkers. While this method provides a binary classification, grouping molecules as either similar or distinct, it has notable limitations, especially in the context of mass spectrometry. For example, minor variations such as differences in linker length can lead to different scaffold assignments for otherwise highly similar molecules. This discrepancy is particularly problematic when fragmentation patterns are nearly identical despite structural variations [98].

To address these limitations, our lab introduced the concept of *Murcko histograms*[5]. Unlike traditional scaffold-based methods that assign a single label to a molecule's core structure, Murcko histograms provide a quantitative measure of molecular connectivity. In brief, after computing the Murcko scaffold, the method records two key metrics for each ring: the number of neighboring rings and the number of

5. Murcko histograms record, for each ring in the scaffold, the number of neighboring rings and the number of directly attached linker atoms, creating a connectivity histogram.

directly attached linker atoms. This process produces a histogram that exhibits two important properties. First, it is invariant to linker length, focusing instead on the connectivity between rings. Second, it offers a quantitative similarity measure with a transitive property: two molecules are considered similar if their histograms match or if one histogram is a subhistogram of the other [98].

By grouping molecules based on these relaxed similarity criteria, Murcko histograms effectively mitigate data leakage and reduce the risk of overlapping fragmentation patterns between training and validation sets. In this thesis, we employ Murcko histograms for data splitting due to their ability to accurately group similar compounds and their computational efficiency compared to alternatives such as Maximum Common Edge Subgraph (**MCES**) [100].

3.3 Mass spectra benchmarks

Benchmarking is crucial for advancing machine learning approaches to *MS*/*MS* spectrum annotation, as it establishes standardized datasets, evaluation metrics, and data splitting protocols that ensure models truly generalize rather than simply overfitting dataset-specific artifacts. Previous efforts, such as the *Critical Assessment of Small Molecule Identification* (CASMI) challenges [101] and the *MIST CANOPUS* [102] benchmark, have significantly contributed to molecular annotation but exhibit limitations in several important aspects. For example, the CASMI challenges comprise only a few **hundred** spectra and require extensive expert preprocessing, factors that limit reproducibility and rapid method development, with the most recent competition held three years ago. Similarly, although MIST CANOPUS curated a balanced dataset of approximately **9,000** molecules, its reliance on 2D InChIKey–based splits often permits chemically similar molecules to appear in both training and validation sets (see Section 3.2).

This initiative, led by our lab in collaboration with multiple international teams, addresses these limitations through the development of *MassSpecGym* [58]. MassSpecGym provides the largest curated collection of high-quality *MS*/*MS* spectra (**231,000** spectra linked to **29,000** unique molecules) and introduces a rigorous data split based on molecular MCES distances. This novel split minimizes the risk of near-

duplicate or highly similar molecules appearing in both training and test sets, as demonstrated by more realistic *Tanimoto similarity* [103] distributions compared to conventional splits. Moreover, MassSpecGym defines three distinct challenges: *de novo* molecule generation, molecule retrieval, and spectrum simulation, with tailored evaluation metrics such as *Tanimoto similarity*, *MCES distance*, and *cosine similarity* for spectral predictions. By standardizing these tasks and providing a machine learning–ready benchmark through user-friendly interfaces (e.g., integration with *PyTorch Lightning* [104] and *Hugging Face* [105]), MassSpecGym bridges the gap between mass spectrometry and AI.

Motivated by the success of the original MassSpecGym, this thesis introduces MassSpecGymMSn, the first benchmark explicitly designed for multi-stage MSn mass spectra, enabling next-generation AI-driven discoveries in mass spectrometry, see Fig. 3.3. Our approach retains the core functionality of MassSpecGym, and further improves previous bottlenecks.

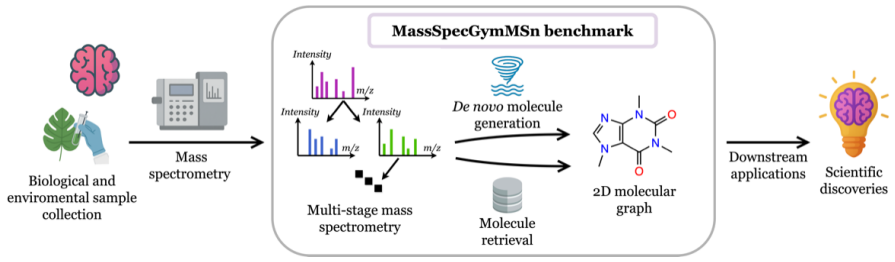

Figure 3.3: Standardized and reproducible benchmarks are essential for accelerating AI-driven discoveries. To support progress in metabolomics and mass spectrometry, we introduce MassSpecGymMSn.

4 MassSpacGymMSn: Dataset and benchmark construction

4.1 Data acquisition and setup

In our lab, colleagues Brungs et al. introduced the **first** high-throughput method for acquiring MSn trees along with an automated workflow for extracting and building open MSn libraries. Using this approach, we collected data from seven compound libraries, encompassing a total of **37,829** compounds. Based on their InChIKey strings[1] (which account for stereoisomers), **34,413** of these compounds represent unique structures, while **2,250** compounds appear in multiple libraries. By applying our acquisition pipeline to these **37,829** small molecules, we obtained MSn spectra for **30,008** unique compound structures within just **23** days in both positive and negative ion mode [5]. For comparison, previous high-throughput workflows acquired only MS2 spectra at speed of $\sim 1,000$ compounds per week [106].

In total, our new MS library comprises **357,065** cleaned and dereplicated MS2 spectra (**170,131** unmerged) and **2,350,646** MSn spectra (**1,366,639** unmerged). Merged spectra are pseudospectra that aggregate individual spectra across collision energies or fragmentation stages, see Figure 4.1b. The absence of spectra for some compounds is likely due to insufficient ionization or low MS1 signal intensities falling below the MSn threshold. In terms of chemical diversity, our dataset includes nearly **22,700** compounds that are not present in other major databases, and this helps expand the coverage of the mass spectrometry measured chemical space of small molecules.

In our workflow, library generation begins with the collection and preparation of samples from multiple compound libraries, followed by database querying [107] to compile detailed metadata. These data are then used to generate acquisition sequences for both positive and negative ionization modes. We optimized a high-throughput flow injection method that achieves a rectangular current intensity profile over 1.5 minutes within a total analysis time of 3 minutes per sample.

1. The InChIKey encodes molecular structure and stereochemistry in a fixed-length hashed string.

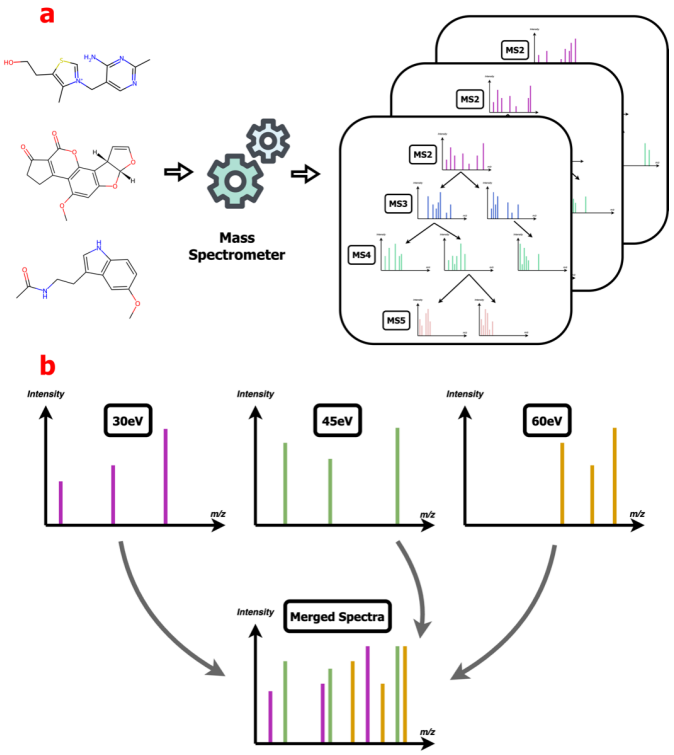

Figure 4.1: (**a**) Illustrates the generation of multi-stage MSn fragmentation trees from molecular inputs. For each compound, a full MSn tree is constructed by sequentially fragmenting ions across multiple stages, resulting in a library of MSn spectra. (**b**) Shows the process of spectra merging, where individual spectra acquired at different collision energies (e.g., 30 eV, 45 eV, and 60 eV) for the same compound are combined into a single merged, or "*pseudo*" spectrum.

This setup supports acquisition up to the MS5 level, selecting up to 25 precursor ions (distributed as 5 for MS3, 10 for MS4, and 10 for MS4) to yield a total of 75 spectra across three collision energies, see Figure 4.2. Notably, only a subset of precursors triggers a full MSn tree.

The mass spectrometer process is automated using robotic and *Echo* liquid handling systems, which mix and dilute up to ten compounds simultaneously. A data-dependent acquisition method is employed with a **50%** split between sampling and washout times. Data-dependent

acquisition is an automated method where, following an initial *MS1* scan, the instrument selects the most intense precursor ions, typically the top five, that exceed a set intensity threshold for further fragmentation. These selected ions are then isolated and fragmented at multiple collision energies to generate MS2 and subsequent MSn spectra, and the process is iteratively repeated for deeper fragmentation levels, ensuring that the most informative signals are systematically targeted.

Conventional liquid chromatography (LC) systems often fail to provide sufficient time during the elution peak to accumulate strong signals necessary for robust mass spectral tree acquisitions, which can exacerbate noise issues [108]. To overcome this, our workflow deliberately avoids the chromatography step, enhancing signal quality.

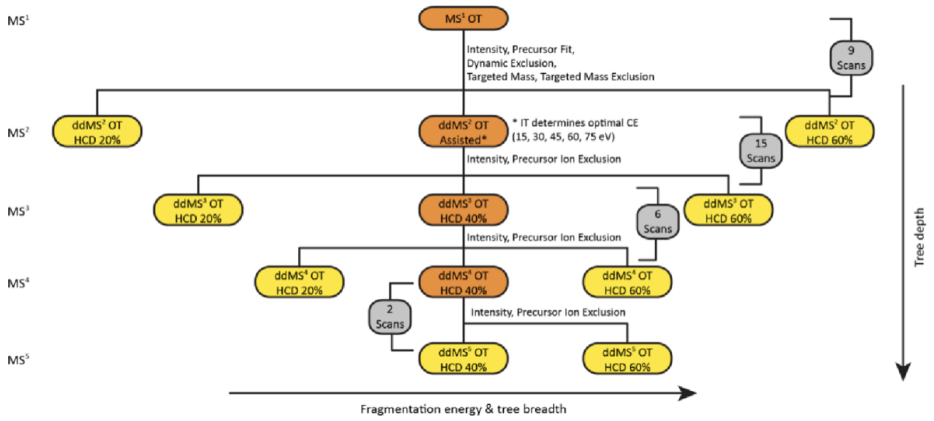

Figure 4.2: The figure illustrates the MSn tree data acquisition workflow as reported by Corrina et al. [5]. It shows that each MS2 precursor ion can trigger up to 75 scans. The legend: OT indicates Orbitrap, IT stands for Ion Trap, and dd denotes data-dependent acquisition.

Once the raw data are acquired, they are imported into our automated *mzMine* [109] workflow, where mass detection, fragmentation tree construction, and spectral annotation are performed. In this processing stage, each complete MSn tree is generated in approximately 13 seconds. Quality control measures ensure that only high-quality, matching spectra are exported along with their quality scores. All scripts, acquisition methods, and protocols are open source and freely available to the research community.

Given the rapid and automated nature of our data acquisition workflow, one might reasonably question whether the generated spectra are truly reliable, especially since traditional methods typically require much longer processing times. To address this concern, we performed a validation using feature-based molecular networking (FBMN) [110] on open-source libraries with GNPS [14]. Specifically, when comparing Tanimoto similarity scores, **88%** of all matches exhibited high similarity values (≥ 0.85). Furthermore, using the maximum common substructure metric (MCES) [100], where an edit distance of less than 4 indicates near-identical fragmentation patterns, we found that **94%** of matches were either identical or highly similar. A recent independent study further confirmed our library's quality by examining compound and spectral metadata across public repositories, which led to the removal of over **31,000** spectra each from GNPS, MoNA, and MassBankEU, while our MSn library failed quality checks for only **58** entries [111].

In conclusion, our solution represents a **paradigm shift** in spectral library construction. Every step of the process, from sample collection and preparation to data acquisition and automated processing, is fully reproducible and open source, with all scripts, methods, and workflows freely shared with the community. This is the *first* library of its kind, and all raw data, compound metadata, and processing parameters are transparently available. This resource is expected to expand the public repository of reference spectra and serve as a catalyst for developing innovative computational tools and machine learning models in metabolomics.

4.2 Implementation: Multi-stage MSn dataset

In this thesis, we extend the original *MassSpecGym* benchmark [58], which was focused on tandem MS2 spectra, into a new benchmark, *MassSpecGymMSn*, designed to handle multi-stage mass spectrometry MSn data (see Fig. 4.3). The MSn benchmark comprises **14,008** unique molecules, **16,476** distinct mass spectra trees, and a total of **183,365** mass spectra acquired in positive ion mode. Spectra in this dataset reach up to **five fragmentation levels**, offering richer structural insights. To ensure reproducibility, the dataset includes standardized

splits (see Section 3.2), defined challenges (see Section 4.4), and accompanying tutorials (see Section 6.4.6) for downstream use.

From an implementation perspective, we introduced several key components and architectural improvements to support this extension. Below, we summarize the main technical contributions that enable efficient handling and modeling of hierarchical MSn data:

- **Hierarchical graph representation**
  The core innovation is the reframing of MSn data as a hierarchical graph structure, where each spectrum tree captures successive fragmentation events. This representation includes the initial MS2 spectrum (*root*) and all subsequent stages.

- **Handling precursor m/z variations**
  Each node in the graph corresponds to a precursor ion with a defined m/z value and its associated spectrum. To address small deviations in precursor m/z values, common due to instrument limitations [5, 112], our implementation links child nodes based on the closest available m/z match, with predefined maximum allowed deviation.

- **Dealing with missing mass spectra**
  To maintain structural integrity when some precursor spectra are missing (due to low signal or shuffled entries in MGF[2]), we introduce *dummy* placeholder nodes during tree construction. These are updated once the corresponding spectra become available, while preserving the overall fragmentation hierarchy.

- **Efficient data handling and graph conversion**
  For efficient batching and model integration, we use PyTorch Lightning [114] for training orchestration. To enable seamless use in graph neural network workflows, spectrum trees are converted into PyTorch Geometric [115] data objects.

- **Caching and multi-GPU optimization**
  To address performance bottlenecks in retrieval tasks, particularly when processing large candidate sets of molecules, we implemented

2. Mascot Generic Format (MGF) is a common text-based format for storing mass spectra [113].

HDF5 [116]-based caching. Molecular transformations for all candidates are precomputed and stored. This optimization is especially critical when evaluating models against full candidate sets. Our pipeline is optimized for multi-GPU systems, achieving over 10× speed up on AMD MI250X and NVIDIA A100 hardware, see Figure A.1.

- **Configurable spectrum featurization**
  We developed a flexible *Spectrum Featurizer* that abstracts the complexity of mass spectra data processing (see Fig 4.3). This component allows users to define feature extraction parameters through a configuration file, specifying which attributes to extract and how they should be encoded for downstream models.

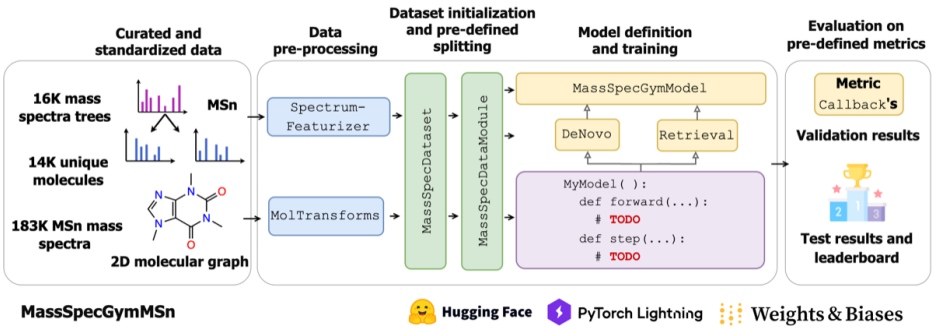

Figure 4.3: The codebase of the benchmark consists of fundamental bricks utilized for preprocessing, loading, training, and evaluation. The user will mainly interact with mass spectra and molecules processing pipelines, or in other words, input and output shaping. We provide a main class for seamless model training and interaction that handles all orchestrations.

4.3 Feature extraction

4.3.1 SMILES canonization

How best to represent molecules in a consistent and standardized way has long been a question in chemoinformatics. One widely adopted

option is the *SMILES* (Simplified Molecular Input Line Entry System) [117] format, which encodes chemical structures as ASCII strings. Because a single molecule can be depicted using various valid SMILES representations, depending on the starting point, atom ordering, stereochemistry, and other factors, a unified representation is a highly sought-after feature. Canonicalization processes convert diverse representations into a unique, standardized form known as **canonical SMILES**, (see Figure 4.4a).

To achieve uniformity across our pipeline, which includes candidate construction, *de novo* molecular generation, and other tasks, we decided to use the PubChem API [69] for canonicalization, as also used in MassSpecGym. In our evaluation, when comparing the SMILES produced by PubChem canonicalization with those generated using the RDKit[3] method, only **64 out of 14,008** molecules yielded matching representations.

During analysis of our current dataset, we discovered that only **6,427 out of 14,008** unique SMILES strings (based on exact string matching) were already canonicalized. After applying the canonicalization procedure to the entire dataset, the total number of unique SMILES remained unchanged at 14,008, and all molecular representations were successfully processed without any errors.

4.3.2 DreaMS embeddings

In this thesis, one of our objectives is to compare foundation model embeddings with raw spectral representations on predictive tasks. To achieve this, we employed the *DreaMS* foundation model developed in our laboratory (see Figure 4.4b). We selected DreaMS for its advanced model architecture and user-friendly interface, in addition to its rigorous data filtering and processing capabilities. The model leverages the extensive MassIVE [119] library and its GNPS [14] subset, a community standard for metabolite reference. This alignment is critical, as it ensures that the learned representations accurately capture the chemical coverage of our dataset [120]. These advantages are the primary reasons we chose this model over alternative options [121,

3. RDKit is an open-source library that provides a flexible framework for chemoinformatics [118].

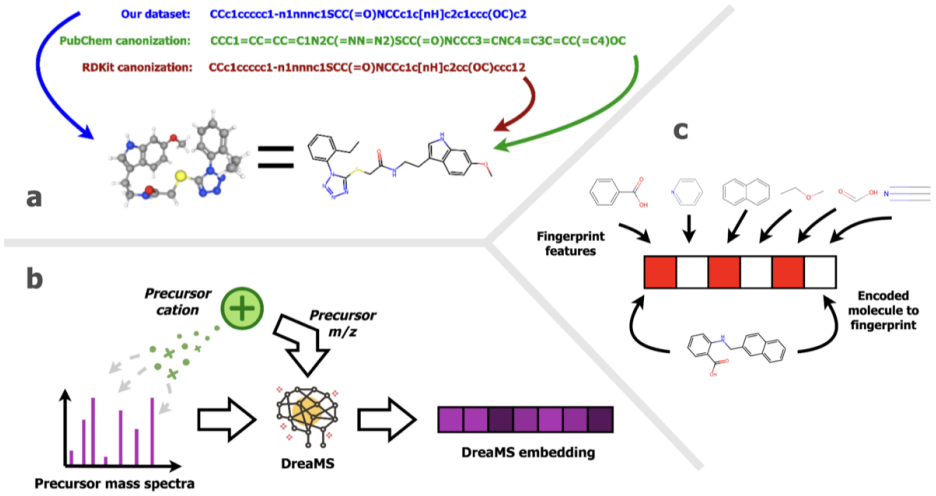

Figure 4.4: (**a**) Three SMILES representations: Original, PubChem canonical, and RDKit canonical, along with 3D conformer and 2D structure. (**b**) The DreaMS prediction pipeline: input precursor m/z and spectrum produce an embedding. (**c**) Molecular fingerprint visualization: structure (below) mapped to its 1024-dimensional fingerprint (middle), with activated features highlighted.

122]. The quality of the DreaMS model is further evidenced by its successful publication in *Nature Biotechnology*.

We processed 183,365 multi-stage fragmentation spectra using the DreaMS model. Although DreaMS was originally trained on pairs of precursor m/z values and corresponding mass spectra rather than on MSn data, we contend that its application remains valid because each spectrum is acquired independently; product ions, in isolation, do not inherently disclose their source [19]. While this approach may not be fully optimal, it provides a useful means of integrating DreaMS into our workflow, which we further evaluate in subsequent chapters. All processing was performed on the RunPod GPU cluster using a single NVIDIA H100 [123].

4.3.3 Molecular fingerprint

Molecular fingerprints are fixed-length vectors that encode the presence or frequency of chemical substructures within a molecule (see

Figure 4.4c). By transforming each molecule into a high-dimensional numeric representation, fingerprints capture subtle structural differences and enable rapid, quantitative comparison via similarity measures such as the Tanimoto similarity coefficient [103]. This makes them ideal for tasks like database search, compound ranking, and diversity analysis, where fast and meaningful measures of molecular likeness are required.

In this thesis, we follow MassSpecGym and employ *Extended-Connectivity (Morgan) fingerprints* [124, 125]. Each fingerprint is a 1024-bit vector generated with radius 2, where each atom's local neighborhood is iteratively hashed into integer identifiers. These vectors form the basis for all similarity-driven retrieval and analysis described in subsequent chapters.

4.4 Multi-stage spectra benchmark challenges

In this section, we define the main challenges addressed in our new *MassSpecGymMSn* benchmarks. The first challenge is **molecule retrieval**, which involves identifying the correct molecular graph from a chemical database based solely on a given multi-stage mass spectra (see Fig. 4.5b). This task is critical for real-world applications that require detecting specific compounds, such as pesticides, environmental pollutants, or other target substances, in a sample [58, 126].

The second challenge is **de novo molecule generation**, which involves predicting a complete molecular graph directly from an MSn spectra without relying on existing databases (see Fig. 4.5d). This task is considerably more complex, as it requires constructing novel molecular structures from inherently ambiguous spectral data, and can be compared to the goal of *AlphaFold* [15, 58].

4.4.1 Formal definition of challenges

In multi-stage fragmentation experiments, the acquired data is represented as a graph rather than a single spectrum. We define the *mass spectra graph* as $G_S = (V_S, E_S)$, where each node $v \in V_S$ is associated with a mass spectrum $X(v) \subset \mathbb{R}^+ \times (0, 1]$. Each spectrum $X(v)$ is a collection of two-dimensional points (m, I), where m denotes the mass-

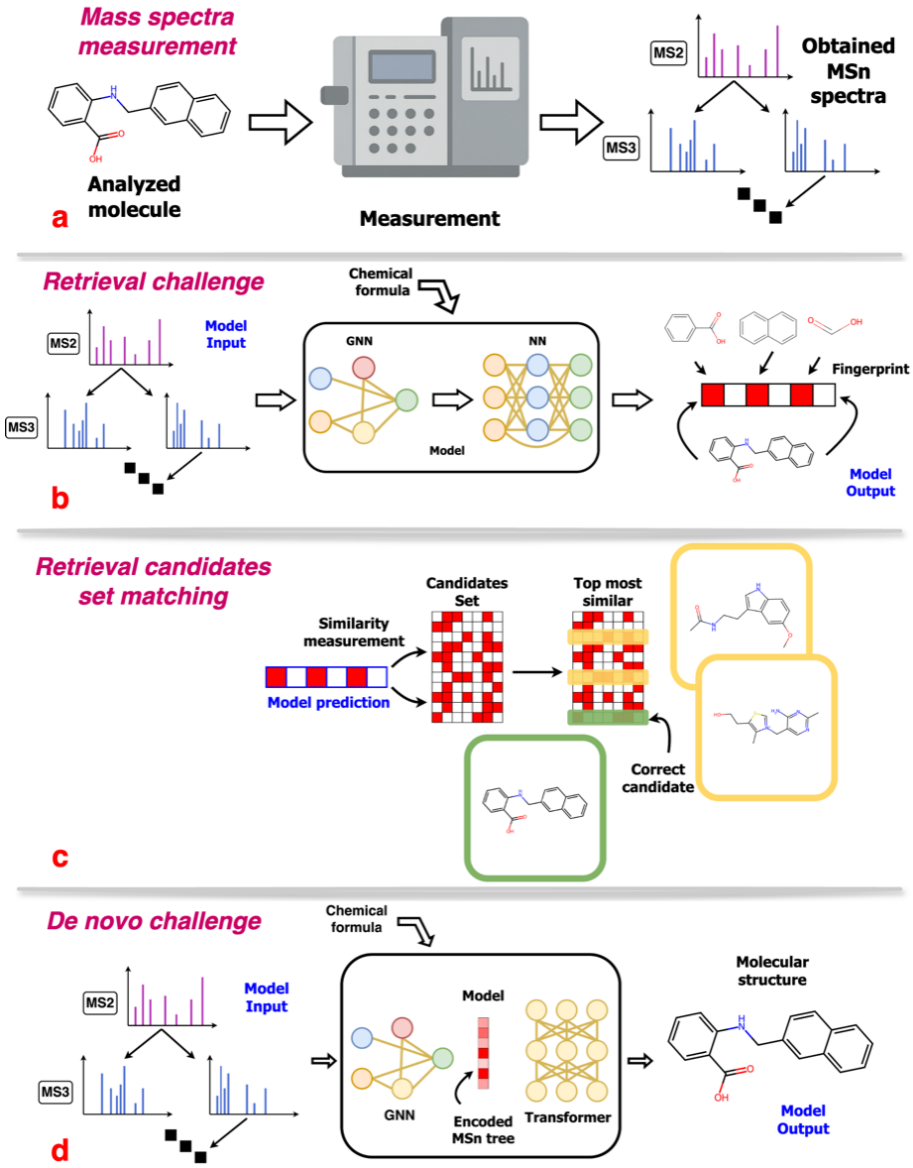

Figure 4.5: Panel (**a**) shows the structure of 2-(naphthalen-2-ylmethylamino)benzoic acid alongside its measured MSn spectrum. Panel (**b**) depicts the retrieval pipeline, which takes the measured MSn spectrum as input and outputs a predicted molecular fingerprint. Panel (**c**) illustrates the candidate-matching step: all database fingerprints are ranked by similarity to the prediction, with the top match ideally corresponding to the correct molecule. Panel (**d**) outlines the de novo prediction pipeline, where the measured MSn spectrum is directly converted into the molecule's SMILES representation.

to-charge (m/z) value and I the corresponding normalized intensity (with $\max\{I : (m, I) \in X(v)\} = 1$). A directed edge $(u, v) \in E_S$ indicates that the spectrum $X(v)$ is generated from a precursor peak in the parent spectrum $X(u)$ by means of a fragmentation event. In this structure, the root node corresponds to the initial precursor spectrum, and the branching pattern of the graph captures the sequential nature of multi-stage fragmentation.

Additionally, to spectra we define the *molecular graph* as $\hat{G}_M = (V_M, E_M)$, where the vertex set V_M is composed of atoms selected from a specified vocabulary of chemical elements[4]. Thus, $V_M \in \mathbb{V}^N$, where $|\mathbb{V}^N| = N$, and N is typically 118 or another number depending on the chosen subset. The edge set E_M represents the chemical bonds connecting these atoms, with each edge corresponding to one of four bond types: single, double, triple, and aromatic [127]. Hence, $E_M \in \mathbb{E}_M$, where $|\mathbb{E}_M| = 4$, and M is determined by the molecular connectivity. Note that the representation $\hat{G}_M = (V_M, E_M)$ models only the topological connectivity of the molecule. It deliberately omits three-dimensional coordinate information, since the data provided by MS2 spectra are generally insufficient for accurately predicting precise molecular conformations [29, 58].

4.4.2 Molecular retrieval benchmark definition

In retrieval tasks, we obtain a candidate set of molecular graphs and then rank them to identify the correct, or, at a minimum, the most similar structure. Formally, given an input derived from the spectra graph G_S, the task is to order a candidate set $C = \{\hat{G}_M^{(1)}, \hat{G}_M^{(2)}, \ldots, \hat{G}_M^{(n)}\}$ so that the correct molecular graph $\hat{G}_M \in C$ appears at the top of the ranking [58] (see Fig. 4.5c).

To standardize the evaluation, the candidate set is limited to at most **256** molecules per spectra graph. From the MS1 data or, equivalently, from the spectra graph G_S, we obtain the precursor mass of the molecule. Therefore, we narrow the candidate pool by considering only those molecules whose molecular masses fall within an

4. For example, the periodic table of 118 elements or a restricted subset such as the 10 most common ones [48].

acceptable experimental error range of the true precursor mass. This constitutes the standard retrieval challenge.

As an additional *bonus challenge*, we refine the candidate set based on molecular formula matching. In practice, chemical formulas can be accurately inferred from MS1 data [128–130], providing reliable information on the atomic composition. Consequently, the candidate set C is restricted to include only those candidate molecular graphs $\hat{G}_M^{(i)}$ whose vertex set V_M exactly corresponds to the known atomic composition of the true molecule $\hat{G}_M$ [58]. Although we present this as a bonus challenge, we acknowledge that accurately predicting chemical formulas remains only partially solved.

We evaluate molecule retrieval using standard information retrieval metrics along with a measure of structural similarity between the top retrieved candidate and the true molecule. Specifically, we consider the following metrics:

- **Hit Rate@k.** For each test example, let $C_k \subset C$ denote the set of the top-k candidate molecular graphs ranked by the retrieval model. The hit rate is defined as:

  $$\text{HitRate@}k = \mathbb{1}\{\hat{G}_M \in C_k\},$$

  where $\mathbb{1}$ is the indicator function that returns 1 if the true molecular graph $\hat{G}_M$ is present in C_k, and 0 otherwise. This value is averaged across all test examples, yielding an overall score in $[0, 1]$, where 1 indicates perfect retrieval performance.

- **MCES@1.** To further evaluate retrieval quality, we compute the *maximum common edge subgraph* (MCES) [100] distance between the top-1 retrieved candidate $\hat{G}_M^{(1)}$ and the true molecular graph $\hat{G}_M$:

  $$\text{MCES@1} = \text{MCES}(\hat{G}_M^{(1)}, \hat{G}_M).$$

  A value of 0 indicates that the top-1 candidate is structurally identical to the true molecule, while larger values correspond to lower structural similarity.

4.4.3 De novo molecule generation definition

The **de novo molecule generation** challenge aims to predict a molecular graph $\hat{G}_M = (V_M, E_M)$ from the acquired multi-stage fragmentation data encoded in the spectra graph G_S (see Fig. 4.5c). In this task, the spectra graph G_S encapsulates the acquired MSn data, where each node is associated with a mass spectrum $X(v) \subset \mathbb{R}^+ \times (0, 1]$ and directed edges denote fragmentation events.

Also, similar to the *retrieval* challenge, we define a *bonus challenge* in which the chemical formula of the true molecule is provided as input, thereby fixing the vertex set V_M as the true molecular graph $\hat{G}_M = (V_M, E_M)$.

While each mass spectrum captures a measurement from a specific compound, it provides only a partial view of the underlying molecular structure. Consequently, multiple molecular graphs $\hat{G}_M = (V_M, E_M)$ may be consistent with the observed data. To account for this uncertainty, we formulate the **de novo generation** task as predicting a set of k candidate graphs $\hat{G}_{M,k} = \{\hat{G}_M^{(1)}, \ldots, \hat{G}_M^{(k)}\}$, rather than a single solution. These candidates can either be sampled randomly from a model or selected as the top-k predictions from a larger set based on a scoring function.

To evaluate *de novo molecule generation*, we define metrics that assess the similarity between the predicted candidates and the true molecule. For each example, we first assess whether the true molecular graph $\hat{G}_M$ is among the top-k predictions in the candidate set $\hat{G}_{M,k}$. Given the inherent challenge of predicting an exact molecular graph, we further evaluate similarity using two complementary metrics:

- **Top-k MCES.** We compute the maximum common edge subgraph (MCES) [100] distance between the most similar candidate and the true molecular graph:

$$\text{Top-k MCES} = \min_{\hat{G}_M^{(i)} \in \hat{G}_{M,k}} \text{MCES}(\hat{G}_M^{(i)}, \hat{G}_M).$$

  A value of 0 indicates structural identity, while larger values indicate lower similarity.

- **Top-k Tanimoto Similarity.** We compute the Tanimoto similarity between the Morgan fingerprints of the predicted candidates

and the true molecule:

$$\text{Top-k Tanimoto} = \max_{\hat{G}_M^{(i)} \in \hat{G}_{M,k}} \text{Tanimoto}(\hat{G}_M^{(i)}, \hat{G}_M).$$

The Tanimoto score ranges from 0 to 1, with 1 indicating an exact match.

These metrics are averaged across all test examples and evaluated for $k \in \{1, 10\}$.

4.5 Preparation candidates set for retrieval task

In retrieval tasks, rather than generating an entire molecular graph *de novo*, we obtain a candidate set $C = \{\hat{G}_M^{(1)}, \hat{G}_M^{(2)}, \ldots, \hat{G}_M^{(n)}\}$ that is subsequently ranked so that the true molecular graph $\hat{G}_M$ appears at the top of the list (see Figure 4.6). Since chemical databases like PubChem contain over 118 million molecules [5], evaluating every candidate is impractical. Therefore, we restrict the candidate set to a maximum of 256 molecules per spectra graph G_S [58].

For constructing this candidate set, we adopt a **hierarchical sampling strategy** similar to that described in the original MassSpecGym paper [58]. At each sampling stage, candidates are filtered by enforcing exact precursor mass matching within the acceptable measurement error and, for the bonus challenge, by requiring an exact match of the molecular formula (i.e., the candidate's vertex set V_M must exactly correspond to the known atomic composition of the true molecule $\hat{G}_M$; see Algorithm 1 for details). Initially, candidates are drawn from a *primary pool* of one million biologically significant molecules [131]. If these criteria do not yield a complete set of 256 candidates, additional molecules are sampled from a *secondary pool* of four million entries [102] and, if necessary, further supplemented from the broader PubChem database. Importantly, the same filtering conditions are strictly applied at every stage, and the correct query molecule is always ensured to be included in the final candidate set [58].

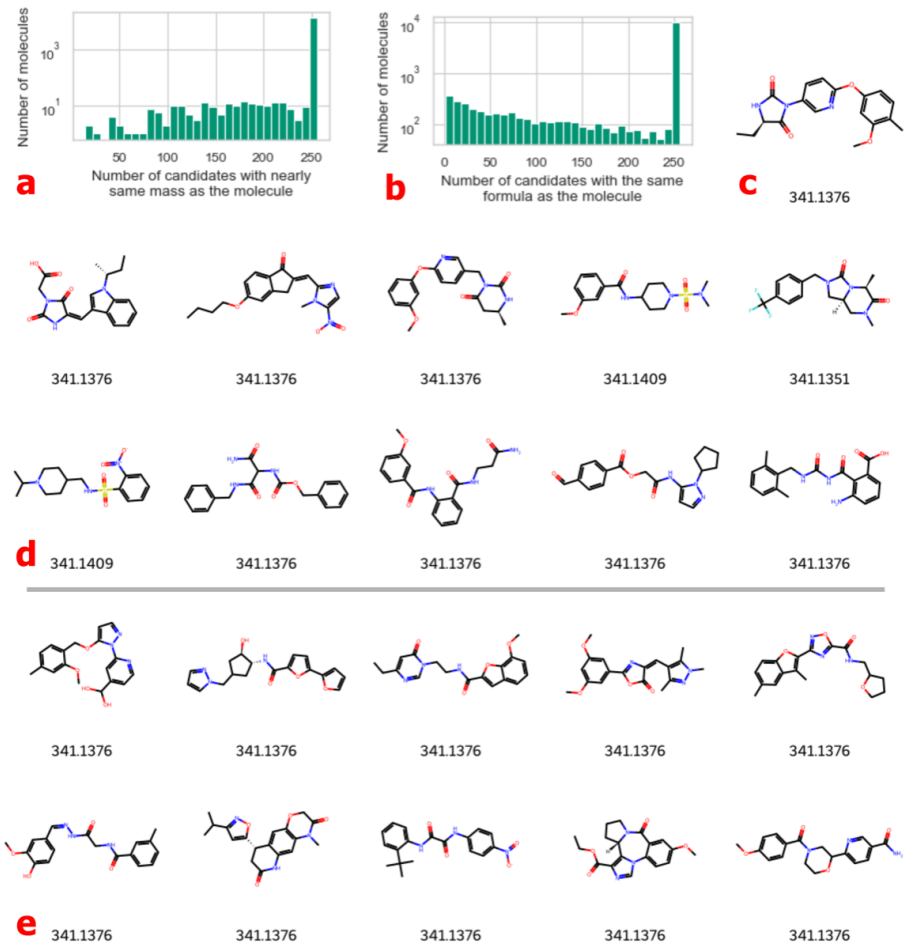

Figure 4.6: Candidate molecule distribution and examples. Distribution of candidate set sizes for two retrieval settings: standard challenge mass-based **a**) and formula challenge **b**). In our dataset, 72% of molecules reach the full set of 256 mass-based candidates, and 98.5% do so with formula-based filtering. **c**) An example molecule with 256 candidates in both settings. **d–e**) Ten randomly selected candidates from each retrieval type for the molecule in **c**), with molecular weights shown below.

Algorithm 1 Candidate selection algorithm (reimplemented from MassSpecGym [58])

Require: Query molecule q, kind of candidates $kind \in \{\texttt{mass}, \texttt{formula}\}$, ordered list of databases D, max candidates N
Ensure: Candidate set C

$C \leftarrow \{q\}$
if $|C| = N$ **then**
  return C
for all $D_i \in D$ **do**
  if $kind = \texttt{mass}$ **then**
    $\varepsilon \leftarrow \text{mass}(q) \times 10 \times 10^{-6}$ ▷ 10 ppm window
    $C \leftarrow C \cup \{\, c \in D_i \mid |\text{mass}(c) - \text{mass}(q)| < \varepsilon \wedge \text{inchi2d}(c) \neq \text{inchi2d}(q) \,\}$
  else if $kind = \texttt{formula}$ **then**
    $C \leftarrow C \cup \{\, c \in D_i \mid \text{formula}(c) = \text{formula}(q) \wedge \text{inchi2d}(c) \neq \text{inchi2d}(q) \,\}$
$C \leftarrow C[:N]$ ▷ truncate to first N
return C

4.6 Exploratory data analysis

Exploratory data analysis (*EDA*) is essential for assessing data quality, uncovering underlying structures, and identifying potential inconsistencies in our MSn fragmentation trees before proceeding to downstream tasks (see Figure 4.7). In our dataset, we identified **14,008** unique SMILES representations corresponding to **16,476** fragmentation trees. During the construction process, we observed that **10** unique trees contained nodes with missing mass spectra, see Section 4.2. To address these cases, we prunes branches containing such nodes, see Fig. 4.7.

Additionally, analysis of unassigned spectra to trees from input MGF files revealed that **71** spectra were not incorporated into any tree. Investigation showed that the primary cause was missing MS2 (root) spectra, which prevented the proper assignment of subsequent child nodes, an issue estimated to affect roughly **3** trees. A secondary issue involved a disordered MGF file where spectra were presented out of order, leading to incorrect tree construction. We decided to exclude such spectra from further analysis to avoid expensive processing workflow for these rare cases.

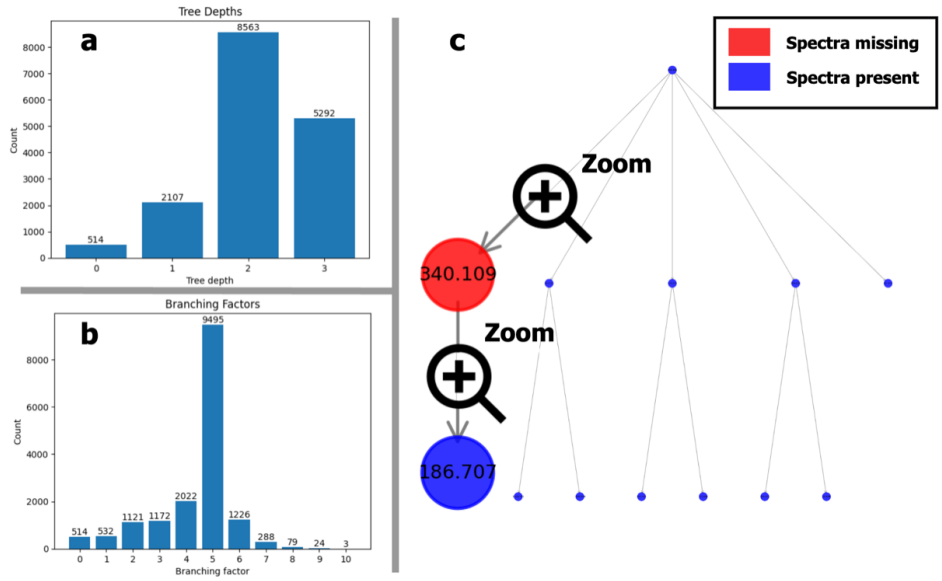

Figure 4.7: Panel (**a**) shows a histogram of tree depths, where depth 0 represents an MS2 spectrum and depth 3 corresponds to an MS5 spectrum; notably, **514** trees contain only an MS2 spectrum, indicating insufficient fragmentation. Panel (**b**) presents a histogram of the maximum branching factors in our MSn trees. For each tree, we recorded the node with the highest number of children and binned the results by branching factor; three trees include a node with 10 children. Panel (**c**) provides a visualization of a tree with missing spectra; the zoomed-in branch highlights a node that would be pruned if we remove missing mass spectra while keeping the mass spectra tree.

We further examined discrepancies between the expected precursor path masses and the actual measured ion masses [5, 112]; for more explanation, refer to section Implementation 4.2 and specifically the second and third points. The largest observed difference was **0.0036621**, within our established threshold of **0.005**, a value determined in consultation with mass spectrometry experts.

4.7 Reproducibility

4.7.1 Standardized split

Data splitting is one of the most critical factors affecting the performance of our algorithms in real-world scenarios. To achieve a *fragmentation-aware split*, we employ a Murcko histogram [98]-based approach (see Section 3.2). We set a distance threshold of 3 for small histograms and 4 for larger ones[5], and the resulting split is computed once and made available for reproducibility.

After constructing the histograms and assigning molecules to training, validation, and test folds with an intended ratio of 80%, 10%, and 10%, respectively, some deviations are inevitable because we assign whole sub-histogram groups. In our implementation, this approach resulted in 10,539 molecules in the training fold, 1,749 in the validation fold, and 1,708 in the test fold (see Table 4.1a).

To assess potential data leakage, we calculated the Tanimoto similarity for each molecule in one split against all molecules in a different split, selecting the closest match. As shown in Figure 4.8, the similarity distribution is centered around 0.3–0.4, indicating minimal overlap between the dataset splits.

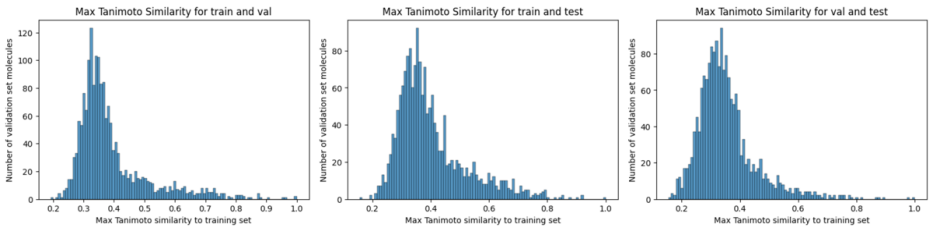

Figure 4.8: This figure presents three histograms showing the molecular Tanimoto similarity distributions between different dataset splits, indicating minimal data leakage and mirroring real-world scenarios where analytes come from distinct chemical distributions. The left panel compares the training and validation sets, the middle panel shows the training and test sets, and the right panel displays the validation and test sets.

5. Thresholds chosen based on empirical analysis of histogram size distributions.

Further to our examination of the data splits, it is essential to consider potential biases inherent to mass spectrometry. Variability in parameters such as ionization methods, adduct types, collision energies [132], and molecular classes can significantly influence fragmentation patterns and spectral quality. To mitigate these risks, we ensured that these key metadata are evenly distributed (see Tables 4.1b and 4.1c).

Table 4.1: Dataset splits: SMILES/spectra proportions, tree statistics, and adduct abundances.

(a) Distribution of SMILES and spectra

Fold	SMILES	Spectra
Train	0.753001	0.750667
Validation	0.124964	0.126240
Test	0.122035	0.123093

(b) Tree statistics for each dataset fold

Statistic	Train	Validatio
Number of trees	12,536	1,952
Average depth	2.12	2.20
Avg. branching factor	4.32	4.59
Avg. precursor m/z	433.05	409.73
Avg. retention time (s)	75.82	76.01
Avg. nodes per tree	10.97	11.85

(c) Relative abundances of adduct types (%)

Adduct	Train	Validation	Test
$[M + H]^+$	82.17	86.42	83.95
$[M + NH_4]^+$	8.06	8.25	5.53
$[M + H - H_2O]^+$	4.07	2.56	4.73
$[M + Na]^+$	2.43	1.64	2.82
$[M]^+$	2.21	0.77	2.36
$[M + H - 2H_2O]^+$	0.94	0.26	0.55
$[M - H_2O]^+$	0.12	0.10	0.05

To assess the distribution of chemical classes across our dataset folds, we employed *ClassyFire* [133], a web-based tool that assigns compounds to a detailed ChemOnt taxonomy. This taxonomy is organized hierarchically into Kingdom, SuperClass, Class, and SubClass levels. For our analysis, we focused on the Class level [134, 135], which comprises 764 distinct categories representing narrowly defined chemical types based on structural features, of which 260 categories are present in our dataset, see Fig. 4.9.

However, the official ClassyFire API proved unstable, frequently failing during batch classification and limiting throughput. To overcome these issues, as a subproject in this thesis, we developed *Py-*

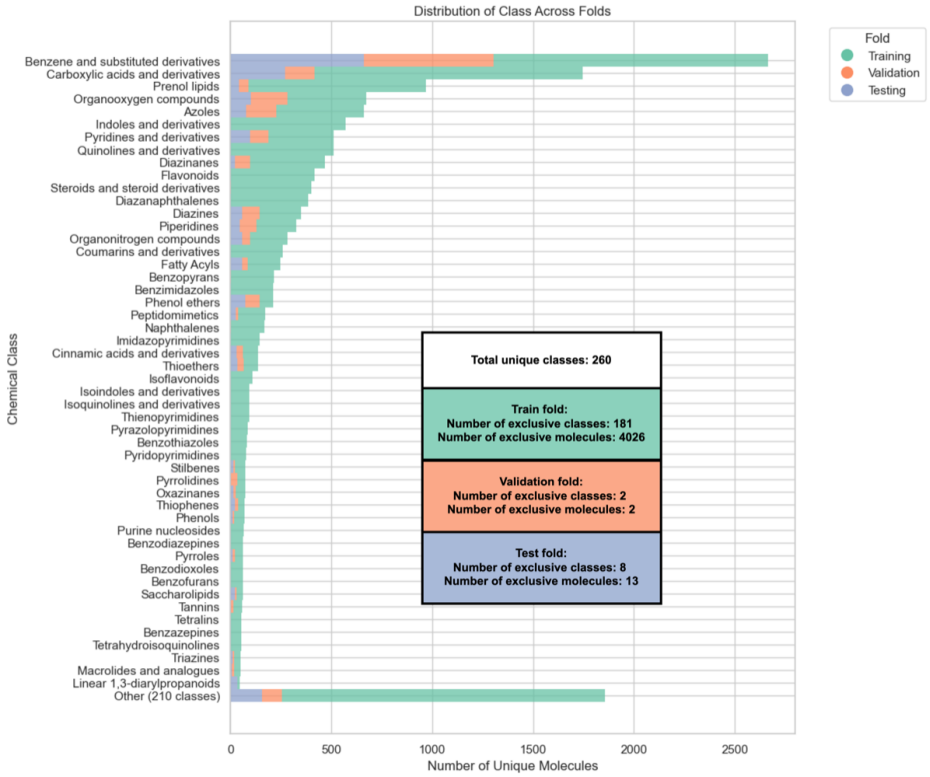

Figure 4.9: The figure illustrates that our data splitting achieves a balanced distribution of chemical classes across the folds. It presents a histogram of the 50 most common chemical classes as determined by ClassyFire in our dataset, along with an "Other" bar aggregating 210 less common classes. Additionally, the histogram shows, for each fold, the number of chemical classes that are exclusive to that fold and the corresponding counts of molecules in those exclusive classes.

ClassyFire[6], which builds upon the ClassyFire API while addressing its limitations, and we provided it as fully open source.

6. An improved wrapper around the ClassyFire API, available on GitHub with documentation, see Section 6.4.6.

5 Model architectures and experimental setup

5.1 Retrieval models architectural details

We propose two main architectures for the retrieval task: one for the **standard** and another for the **bonus** challenge, where the molecular formula is known. In both cases, the model takes a mass spectra tree as input and outputs a molecular fingerprint representing the target molecule (see Fig. 4.5b). The backbone of the model is a *graph neural network* (*GNN*), based on a *Graph Attention Network* (*GAT*) [136], which encodes the tree-structured input. Node embeddings are aggregated via mean pooling, and a final skip connection integrates the original features for enhanced prediction performance.

In other words, in our setup, each node of the tree is one measured spectrum (or its *DreaMS embedding*, see Fig. 5.1), and edges indicate which precursor peak gave rise to which fragment spectrum, see Fig. 2.3. A GNN processes this tree by letting each node gather information from its neighbors: the GAT layer specifically learns to weight messages from each neighbor according to their relevance ("attention") so that more informative fragments contribute more strongly to the node's new representation. After several such layers, each node's *GNN embedding* carries the local spectral features and context from the surrounding fragmentation events. Because we ultimately want a fixed-size fingerprint for the entire tree, we aggregate all node embeddings into one summary vector (via mean pooling); this ensures invariance to the number or order of nodes while preserving the overall mass spectra tree knowledge. Finally, a dense network refines this summary into the predicted molecular fingerprint (see Fig. 4.5).

In the bonus task, the architecture is extended to include an additional input branch that encodes the molecular formula. This formula is processed using a multi-layer perceptron (*MLP*), and its output is concatenated with the GNN-based spectral representation. The merged representation is then passed through a skip connection block to produce the final molecular fingerprint.

The formal description of both architectures is as follows:

1. **Graph attention encoding:**
   Each node $v \in V_S$ in the mass spectra tree $G_S = (V_S, E_S)$ is encoded using a GAT layer:

$$h_v = \mathrm{GAT}\big(X(v),\, G_S\big) \quad \in \mathbb{R}^d,$$

   where d is the shared hidden dimension preserved across model and is crucial for the internal representation analysis in Section 6.4.

2. **Mean aggregation (standard and bonus):**
   Node embeddings are aggregated to form a graph-level representation:

$$h_{G_S} = \frac{1}{|V_S|} \sum_{v \in V_S} h_v \quad \in \mathbb{R}^d.$$

3. **Formula branch (bonus task only):**
   The molecular formula, represented as a feature vector $f \in \mathbb{R}^{d_f}$, where each of the d_f dimensions counts a specific atom type, is encoded by a formula encoder:

$$h_f = E_{\text{form}}(f) \quad \in \mathbb{R}^{d_{\text{encoded formula}}}.$$

4. **Concatenation (bonus task only):**
   Spectra and formula representations are concatenated and linearly projected back to $\mathbb{R}^d$:

$$h_{\text{concat}} = \Pi\big[h_{G_S} \,\|\, h_f\big] \quad \in \mathbb{R}^d.$$

5. **Skip connection and fingerprint prediction:**
   We define

$$f_{\text{skip}} : \mathbb{R}^d \xrightarrow{\text{MLP}} \mathbb{R}^{d_F},$$

   The input vector in $\mathbb{R}^d$ is then passed through this MLP:

$$\hat{F} = \begin{cases} f_{\text{skip}}\big(h_{G_S}\big), & \text{(Standard)} \\ f_{\text{skip}}\big(h_{\text{concat}}\big), & \text{(Bonus)} \end{cases} \quad \in \mathbb{R}^{d_F}.$$

Table 5.1 summarizes the hyperparameter configurations for our retrieval task models, comparing two spectra processing variants, binned spectra [137] and DreaMS [19], each implemented in both standard and bonus settings. All hyperparameters were chosen based on standard recommendation practices [76] and to support model expressivity. Smaller models, or those trained with alternative loss functions, failed to learn effectively and plateaued in performance after just a few epochs.

In total, we evaluate four model variants. Our goal is to compare the performance of models that only differ in their input representation, using either the commonly used binned spectra approach [137] or the DreaMS [19] processed spectra, while keeping all other parameters constant. This isolates the impact of the input representation on retrieval performance and assesses the benefit of incorporating molecular formula information in the bonus models.

Table 5.1: Model configurations for the standard and bonus retrieval tasks. Each model maps spectra trees to molecular fingerprints using a GNN (GAT-based) backbone; bonus models include an extra formula branch.

Hyperparameter	Binned spectra Standard	Binned spectra Bonus	DreaMS spectra Standard	DreaMS spectra Bonus
`hidden_dim`	1024	1024	1024	1024
`fp_dim`	2048	2048	2048	2048
`input_dim`	4000	4000	1024	1024
`dropout_rate`	0.2	0.2	0.2	0.2
`bottleneck_factor`	1.0	1.0	1.0	1.0
`num_skipblocks`	6	6	6	6
`num_gnn_layers`	3	3	3	3
`gnn_layer_type`	GAT	GAT	GAT	GAT
`nheads`	4	4	4	4
`use_formula`	false	true	false	true
`formula_dim`	0	64	0	64
`loss`	cosine sim	cosine sim	cosine sim	cosine sim

Legend: `hidden_dim`: Width of hidden layers; `fp_dim`: Size of fingerprint vector; `input_dim`: Node feature dimension; `dropout_rate`: Dropout probability; `bottleneck_factor`: Compression ratio in skip connections; `num_skipblocks`: Number of skip blocks; `num_gnn_layers`: Number of GNN layers; `gnn_layer_type`: Type of graph conv.; `nheads`: Attention heads in GAT; `use_formula`: Formula branch used; `formula_dim`: Formula embedding size; `loss`: Training objective.

5.2 De novo models architectural details

In addition to the retrieval task, we implemented a **de novo molecular generation** model. Unlike the retrieval models that use an *MLP* to predict a molecular fingerprint, our de novo architecture employs an autoregressive framework based on the Transformer model from "*Attention Is All You Need*" [77] to generate molecules as *SMILES* strings.

In this architecture, the input spectra tree is first processed by a GAT [136] (see Section 5.1) to produce a conditional representation. This representation is then used to condition a Transformer decoder that predicts the SMILES sequence token by token. For the bonus variant, we incorporate a molecular formula branch, similar to the retrieval models, by concatenating its output with the spectra tree representation before feeding it to the decoder. To handle the sequential nature of SMILES generation, we apply standard *sinusoidal positional embeddings* [87, 138] and employ *byte pair encoding* [139].

We refined the decoder by pretraining it on 2.6 million SMILES strings drawn from two libraries of one million [131] and four million [102] natural products and biologically significant molecules. During this self-supervised [140] phase, we treated *SMILES* as a sequence prediction task: at each step, the decoder is fed the preceding tokens and trained to predict the next token, using *cross-entropy loss* [139] to measure errors. Further, we ensured no data leakage occurred throughout the pretraining process by verifying that the pretraining examples differed from those in the validation and test splits (see Figure A.2).

The formal structure of both variants is as follows:

1. **Graph attention encoding:**
   Each node $v \in V_S$ in the mass spectra tree $G_S = (V_S, E_S)$ is encoded using a GAT layer:

   $$h_v = \mathrm{GAT}\big(X(v),\, G_S\big) \quad \in \mathbb{R}^d,$$

   where d is the shared hidden dimension preserved across model. The graph-level embedding is obtained by mean pooling:

   $$h_{G_S} = \frac{1}{|V_S|} \sum_{v \in V_S} h_v \quad \in \mathbb{R}^d.$$

2. **Formula branch (bonus variant only):**
   The molecular formula is represented as a fixed-length vector $f \in \mathbb{R}^{d_f}$ and encoded by the same MLP:

$$h_f = E_{\text{form}}(f) \quad \in \mathbb{R}^{d_{\text{encoded formula}}}.$$

3. **Concatenation and encoder projection:**
   Concatenate and project back to $\mathbb{R}^d$:

$$u = \begin{cases} h_{G_S}, & \text{(Standard)} \\ \Pi\big[h_{G_S} \,\|\, h_f\big], & \text{(Bonus)} \end{cases} \quad \in \mathbb{R}^d.$$

   Then

$$z = f_{\text{enc}}(u) \quad \in \mathbb{R}^d.$$

4. **Transformer decoding:**
   The decoder generates the SMILES sequence autoregressively. At step t, given previous tokens $y_{<t}$ and embeddings $E(y_{<t})$, the decoder computes:

$$d_t = T\big(E(y_{<t}),\, z\big) \quad \in \mathbb{R}^d.$$

   A linear layer maps d_t to logits:

$$\ell_t = W_{\text{out}}\, d_t, \quad W_{\text{out}} \in \mathbb{R}^{|\mathcal{V}| \times d_{\text{model}}},$$

   where $|\mathcal{V}|$ is the token vocabulary size in our experiments (all set to 3000), and then we predict next–token:

$$p(y_t \mid y_{<t}, z) = \text{softmax}(\ell_t).$$

During training, we employ cross-entropy loss between predicted and target SMILES tokens. The pretrained decoder is frozen for the first 5 epochs to allow the encoder to adapt its latent space. To generate multiple candidates, we use *beam search decoding* [141].

Table 5.2 summarizes the hyperparameter configurations for our de novo models, comparing four variants, binned spectra, and DreaMS inputs, each in standard and bonus settings. This design isolates the impact of input representation and the addition of molecular formula information while keeping other parameters constant. We chose the parameters to match the dimensions of the retrieval model and the overall count of trainable parameters.

Table 5.2: Model configurations for the standard and bonus de novo molecular generation tasks. All models generate structures from spectra trees using a GNN encoder and transformer decoder; bonus models may integrate molecular formula.

Hyperparameter	Binned spectra Standard	Binned spectra Bonus	DreaMS spectra Standard	DreaMS spectra Bonus
`input_dim`	4000	4000	1024	1024
`hidden_dim`	1024	1024	1024	1024
`nhead`	4	4	4	4
`num_gat_layers`	3	3	3	3
`num_gat_heads`	4	4	4	4
`gat_dropout`	0.2	0.2	0.2	0.2
`decoder_layers`	4	4	4	4
`dropout_rate`	0.1	0.1	0.1	0.1
`max_tokens`	200	200	200	200
`temperature`	1.0	1.0	1.0	1.0
`use_formula`	false	true	false	true
`formula_dim`	0	0	0	0
`pretrained`	true	true	true	true
`freeze_epochs`	5	5	5	5
`loss`	cross entropy	cross entropy	cross entropy	cross entropy

Legend: `input_dim`: Dimensionality of node input features; `hidden_dim`: Width of internal hidden layers; `nhead`: Number of attention heads in transformer decoder; `num_gat_layers`: Number of GAT encoder layers; `num_gat_heads`: Attention heads per GAT layer; `gat_dropout`: Dropout rate in GAT encoder; `decoder_layers`: Number of transformer decoder layers; `dropout_rate`: Dropout applied in decoder; `max_tokens`: Maximum sequence length during generation; `temperature`: Sampling temperature for decoding; `use_formula`: Whether molecular formula branch is used; `formula_dim`: Dimensionality of formula embedding; `pretrained`: Decoder initialized from pretrained checkpoint; `freeze_epochs`: Number of epochs decoder is frozen; `loss`: Objective function for training.

5.3 Experimental design

After constructing the fragmentation trees, establishing a leakage-free split, and designing our model architectures, a natural question arises: does multi-stage mass spectrometry offer benefits over conventional MS1/MS2 approaches? While it may seem that all relevant ions are captured at the MS2 level, low-intensity signals, often hard to distinguish from noise, can carry valuable information that conventional analyses tend to underweight [19, 142, 143]. In contrast, multi-stage mass spectra use data-dependent acquisition to measure signals of interest, enhancing the visibility of these subtle yet potentially critical features [5].

In this section, we design experiments to assess whether incorporating additional levels of mass spectra in a hierarchical manner improves model performance. Our fragmentation trees consist of a

root spectrum at the MS2 level, with deeper levels extending to MS5. Accordingly, we define four distinct experiments: the first trains the model using only MS2 data, the second incorporates both MS2 and MS3 data, the third adds MS4, and the fourth includes MS5 as well (see Figure 5.1).

For each experiment, we train the model from scratch while keeping all experimental conditions, such as random weight initialization, batch loading, and shuffling, unchanged. We want to ensure that observed performance variations are predominantly attributable to differences in the input data. In addition to varying the tree depth (MS2 only; MS2–MS3; MS2–MS4; MS2–MS5), we consider two input representations: raw binned spectra and DreaMS-processed spectra (both in standard and bonus challenge). Overall, this framework yields 16 experiments for the retrieval task and 16 experiments for the de novo generation task.

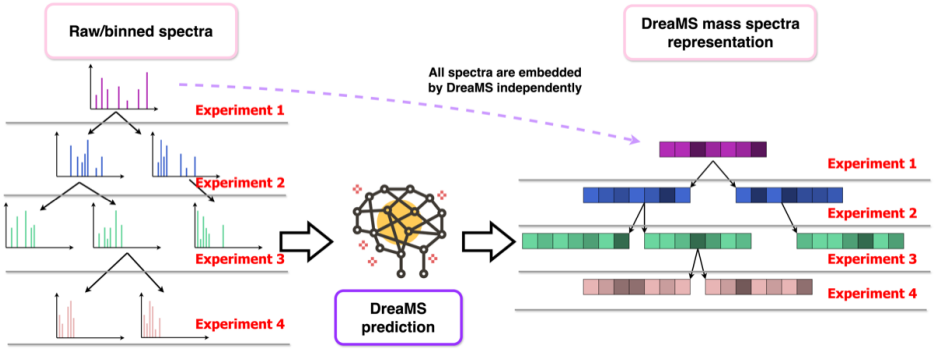

Figure 5.1: This figure demonstrates the transformation of raw spectra trees into embedding-based representations using the DreaMS model and the four hierarchical experimental conditions: training on MS2 only, progressively adding MS3, MS4, and MS5 levels. While DreaMS operates on raw spectra with precise peak resolution, the comparative experiments use **binned spectra** as an input representation.

5.4 Experimental setup and training environment

To ensure portability and reproducibility, we established a standardized environment in which all implementations operate on the same

software package. Training and data preparation are managed via a *YAML* configuration file that contains all parameters necessary to replicate the experimental conditions. All models were trained using identical settings, as summarized in Table 5.3.

Table 5.3: Environment and training settings used across all experiments.

Parameter	Value
Batch size	32 (effective: 256)
Number of epochs	30
Optimizer	Adam
Learning rate	0.0001
Weight decay	0.0001
GPU type	AMD MI250X
Number of GPUs	8
Multi-node training strategy	Distributed Data Parallel
Number of data loading workers	4

6 Experimental results and analysis

6.1 Retrieval models evaluations

6.1.1 Standard challenge

In this section, we evaluate our retrieval models on the standard challenge (see Section 4.4.2), comparing two spectral representations: binned spectra and DreaMS embeddings.

Table 6.1 displays two metrics: Hit Rate@1 is the fraction of examples where the correct molecule tops the ranked list, and MCES@1 measures structural similarity via the size of the maximum common edge subgraph, where 0 means identical molecule, for more detail, see Section 4.4.

As shown in that Table 6.1, the best results with **binned spectra** are achieved when the full fragmentation hierarchy (MS2–MS5) is used. In contrast, for **DreaMS embeddings**, optimal performance occurs with MS2-MS4, likely due to the fact that MS5 data is sparsely represented in the dataset (see Section 4.6).

Across all metrics and both validation and test sets, **DreaMS embeddings consistently outperform** binned spectra. On the test fold, the worst-performing model using binned spectra achieves a Hit Rate@1 of 0.012, whereas DreaMS embeddings yield up to a 10-fold improvement. Since binned spectra have been a dominant approach for years [144], this underscores the untapped potential of multi-stage MSn inputs. However, even these gains fall short of a ceiling, leaving substantial room for future innovation.

Finally, learning curves (Figure A.3 for binned spectra and Figure A.4 for DreaMS) confirm stable training across 30 epochs, with no signs of overfitting.

6.1.2 Bonus challenge

In this section, we evaluate the retrieval models on the **bonus task**, where candidate molecules are restricted to those sharing the same chemical formula as the query (see Section 4.4.2). As in the standard task, we compare two input representations, binned spectra and DreaMS embeddings.

Table 6.1: Retrieval performance (HR@1 ↑, HR@20 ↑, MCES@1 ↓) for binned spectra and DreaMS embeddings across MS2–MS5 on validation and test sets on standard challenge.

		Binned spectra			**DreaMS embeddings**		
Set	**Stage**	HR@1 ↑	HR@20 ↑	MCES@1 ↓	HR@1 ↑	HR@20 ↑	MCES@1 ↓
Validation	MS2	0.022	0.133	21.439	0.031*	0.171*	19.784*
	MS3	0.080	0.338	17.202	0.100*	0.436*	15.913*
	MS4	0.084	0.378	16.888	**0.116***†	**0.463***†	**15.810***†
	MS5	**0.088**	**0.388**	**16.747**	0.100*	0.433*	15.978*
Test	MS2	0.012	0.114	18.903	0.018*	0.153*	18.317*
	MS3	0.079	0.332	15.812	0.110*	0.410*	14.684*
	MS4	0.090	0.382	15.204	**0.112***†	**0.450***†	**14.502***†
	MS5	**0.091**	**0.385**	**15.160**	0.103*	0.413*	14.732*

Legend: HR@1 = Hit Rate@1; HR@20 = Hit Rate@20; ↑ higher is better; ↓ lower is better; Bold = best across MS2–MS5 for that representation; * = this representation outperforms the other at the same stage and set; † = global best among both representations and all stages in that set.

Table 6.2 reports performance across validation and test sets and highlights the best-performing configuration for each metric.

Table 6.2: Retrieval performance (HR@1 ↑, HR@20 ↑, MCES@1 ↓) for binned spectra and DreaMS embeddings across MS2–MS5 on validation and test sets on bonus challenge.

		Binned spectra			**DreaMS embeddings**		
Set	**Stage**	HR@1 ↑	HR@20 ↑	MCES@1 ↓	HR@1 ↑	HR@20 ↑	MCES@1 ↓
Validation	MS2	0.032*	0.227	13.690	0.031	0.235*	13.647*
	MS3	0.045	0.299	**13.043**	0.064*	0.366*	12.506*
	MS4	**0.046**	**0.300**	13.070	0.074*	0.384*	12.343*
	MS5	**0.046**	**0.300**	13.100	**0.075***†	**0.396***†	**12.211***†
Test	MS2	0.043*	0.248	12.723	0.042	0.251*	12.522*
	MS3	**0.068**	**0.323**	**11.857**	0.091*	0.383*	11.244*
	MS4	0.063	0.319	11.959	**0.104***†	**0.426***†	**10.933***†
	MS5	0.061	**0.323**	12.029	0.099*	0.424*	10.961*

Legend: HR@1 = Hit Rate@1; HR@20 = Hit Rate@20; ↑ higher is better; ↓ lower is better; Bold = best across MS2–MS5 for that representation; * = this representation outperforms the other at the same stage and set; † = global best among both representations and all stages in that set.

Overall, DreaMS embeddings outperform binned spectra on nearly all metrics and depths. The only exception is a marginal difference at Hit Rate@1 using MS2 data, where binned spectra slightly lead by less than one-thousandth. In both representations, increasing fragmentation depth improves models performance.

We also can spot distinctive patterns, models trained on fragmentation depths up to MS4 outperform those trained on MS5. This drop likely reflects GNN over-smoothing [145]: with only three layers and input as a sparsely connected spectra tree, deeper fragmentation likely yields homogenized node embeddings. Over-smoothing causes node features to converge, degrading discriminative power, an effect even stronger in the bonus task.

Finally, loss curves (Figure A.5 for binned spectra and Figure A.6 for DreaMS) show that training is stable over 30 epochs, with deeper fragmentation consistently reducing loss and with no evidence of overfitting.

6.2 De novo models evaluations

6.2.1 Standard challenge

In this section, we evaluate our **de novo** generation models under the standard challenge (see Section 4.4.3). As with the retrieval task, we compare two spectral representations, binned spectra and DreaMS embeddings.

In de novo models analysis, we complement MCES with Tanimoto similarity to capture broader structural agreement. It measures the proportion of shared bits in binary fingerprints (see Section 4.3.3), emphasizing substructure overlap over exact bond matches. Therefore, Tanimoto scores range from 0 to 1, where the lowest value means that there are no shared features and one means identical fingerprint. We computed Tanimoto on fingerprints of predicted versus ground-truth molecules.

Table 6.3 reports model performance using MCES and Tanimoto similarity for both top-1 and top-10 predicted structures.

The results show that moving from MS2 to deeper fragmentation levels (MS3 and above) yields clear performance improvements, often by several folds, across both metrics. However, in contrast to the retrieval task, the relative differences *between* models trained on deeper MSn levels (e.g., MS3 vs. MS4 or MS5) are less pronounced. To investigate this, we examine internal model representations in Section 6.4, where we identify an architecture bottleneck limiting how effectively the generation models leverage multi-stage spectral information.

Table 6.3: Performance of de novo generation models (MCES@1 ↓, MCES@10 ↓, Tanimoto@1 ↑, Tanimoto@10 ↑) for binned spectra and DreaMS embeddings across MS2–MS5 on validation and test sets on standard challenge.

	Binned spectra				DreaMS embeddings			
Stage	MCES@1 ↓	MCES@10 ↓	T@1 ↑	T@10 ↑	MCES@1 ↓	MCES@10 ↓	T@1 ↑	T@10 ↑
				Validation set				
MS2	99.965*	69.792*	0.001	0.050*	100.000	99.924	0.001	0.001
MS3	**51.100***	32.183*	**0.077**	0.121	53.802	33.097	0.085*	0.147*
MS4	54.138	**31.197** *†	0.074	**0.125**	**48.234***†	**31.237**	**0.097***†	**0.155***†
MS5	52.336*	31.484*	0.076	0.124	57.666	36.757	0.077*	0.139*
				Test set				
MS2	99.931*	75.623*	0.001	0.042*	100.000	99.054	0.001	0.002
MS3	**53.594**	32.287	**0.075**	0.130	40.428*	26.127*	0.110*	0.174*
MS4	57.002	**31.034**	0.072	**0.133**	**37.659***†	**24.727***†	**0.119***†	**0.184***†
MS5	55.145	31.385	**0.075**	0.132	43.898*	27.160*	0.106*	0.174*

Legend: T = Tanimoto similarity; ↓ lower is better; ↑ higher is better; Bold = best across MS2–MS5 for that representation; * = this representation outperforms the other at the same stage and set; † = global best across both representations and all stages in that set.

Moreover, training loss curves (see Figures 6.3 and 6.3) show that, unlike retrieval, deeper fragmentation does not significantly alter convergence dynamics. Nonetheless, the consistent gains over MS2-only baselines confirm the benefit of incorporating additional fragmentation stages for molecular structure generation.

6.2.2 Bonus challenge

In the **bonus challenge** for de novo generation, the molecular formula is provided (see Section 4.4.3). We again evaluate two spectral representations: binned spectra and DreaMS embeddings.

Across all fragmentation stages, DreaMS embeddings consistently outperform binned spectra. However, unlike in the standard challenge, the inclusion of deeper fragmentation levels (MS3-MS5) does not lead to pronounced gains. This behavior is examined in Section 6.4, where we show that generation models tend to focus heavily on the provided molecular formula and underutilize MSn data.

As shown in Figures A.10 and A.9, the training loss remains stable over 30 epochs, with no signs of overfitting. Including deeper fragmentation stages does not significantly impact convergence, in contrast to the retrieval setting.

Table 6.4: Performance of de novo generation models (MCES@1 ↓, MCES@10 ↓, Tanimoto@1 ↑, Tanimoto@10 ↑) for binned spectra and DreaMS embeddings across MS2–MS5 on validation and test sets on bonus challenge).

	Binned spectra				DreaMS embeddings			
Stage	MCES@1 ↓	MCES@10 ↓	T@1 ↑	T@10 ↑	MCES@1 ↓	MCES@10 ↓	T@1 ↑	T@10 ↑
Validation set								
MS2	57.054	31.019*	0.076	0.150	55.023*	31.131	0.089*	0.150
MS3	61.589	33.659	0.070	0.145	45.962*	29.345*	0.106*	0.171*
MS4	**54.420**	**30.774**	**0.083**	**0.156**	**44.916***†	**28.167***†	**0.110***†	**0.177***†
MS5	58.674	31.123	0.075	0.152	47.231*	29.095*	0.103*	0.171*
Test set								
MS2	45.001	25.321	0.099	0.167	39.203*	24.345*	0.110*	0.170*
MS3	51.953	26.844	0.088	0.164	37.001*	23.017*	0.123*	0.190*
MS4	**44.643**	**24.884**	**0.100**	**0.171**	**35.907***†	**20.923***†	**0.127***†	**0.199***†
MS5	46.830	25.550	0.096	0.169	36.140*	22.741*	**0.127***†	0.193*

Legend: T = Tanimoto similarity; ↓ lower is better; ↑ higher is better; Bold = best across MS2–MS5 for that representation; * = this representation outperforms the other at the same fragmentation stage and set; † = global best across both representations and all fragmentation stages in that set.

When evaluating the quality of generated molecular structures, we observe that deeper fragmentation consistently improves performance, confirming the potential of MSn data. However, there is still substantial room for improvement in absolute terms, with the highest Tanimoto similarity of $\sim$ 0.2, indicating limited structural overlap between predicted and true molecules. Likewise, MCES scores hover $\sim$ 20, reflecting considerable deviation in atomic connectivity. These observations hold across both the bonus and standard challenge, pointing to a general limitation for direct usage of models.

Still, these findings should be viewed as a promising first step. The results highlight the richness and complexity of the MSn dataset and offer a strong foundation for future improvements in model architecture and training strategies.

6.3 Spectral similarity analysis across MSn levels

In addition to training our models, our structured MSn tree representations provide a foundation for a deeper investigation of mass spectra relationships. In this section, we analyze the similarities between spectra across different MSn levels using *statistical tests* to assess potential hierarchical dependencies. Additionally, we compare the original mass

spectra with DreaMS-processed representations to estimate the extra value of foundation models in mass spectrometry.

For our analysis, we extract pairs of nodes from our multi-stage MSn trees. Each tree is structured such that the root is at the MS2 level (position 2), with subsequent nodes at MS3 (position 3) up to MS5 (position 5). We then compare node pairs within the same tree to assess the relationships between fragments originating from the same precursor; for example, a pair labeled (2, 3) represents a direct comparison between an MS2 spectrum and its immediate descendant at the MS3 level. When comparing nodes from different fragmentation levels, we require that the higher-level node is part of a direct lineage within the same branch. This approach preserves the inherent **hierarchical relationships** between fragments originating from the same precursor.

For same level comparisons, when a given level contains multiple nodes within the same tree (e.g., (3,3) pairs), we sample all possible intra-tree node pairs. However, since each tree contains only one MS2 node, (2,2) pairs must be collected from different trees[1]. For more details, refer to Algorithm 2 and Figure A.11.

6.3.1 Hungarian similarity on raw spectra

The cosine score is used to quantify the similarity between two mass spectra by optimally matching their peaks. For each pair of spectra, potential peak matches are identified when the m/z ratios fall within a tolerance of 0.1[2], and the Hungarian algorithm is then applied to solve the peak assignment problem, ensuring an optimal alignment [111]. Using the constructed pairs, we measured the spectral similarity, and the resulting cosine similarity distributions exhibit considerable heterogeneity (see Figure 6.1).

Notably, the frequency of node pairs varies substantially: the most abundant are (4,4) pairs with 250,039 examples, while the least frequent are (2–4,5) pairs with only 14,511 examples. To mitigate the influence of more prevalent pairs on our statistical analysis, we randomly downsampled each pair type to 14,511 examples.

1. To avoid bias, (2,2) comparisons are drawn randomly across trees while preserving sample size parity.
2. Based on typical instrument resolution.

Algorithm 2 Constructing intra tree pairs

```
1: procedure ConstructHierarchicalPairs(T)
2:     N ← BFS(T)                                   ▷ Collect all nodes from tree T
3:     for all n ∈ N do
4:         D(n) ← ComputeDescendants(n)   ▷ Precompute descendants of n via BFS
5:     for i ← 1 to |N| − 1 do
6:         n_A ← N[i]; ms_A ← GetMSLevel(n_A)
7:         for j ← i + 1 to |N| do
8:             n_B ← N[j]; ms_B ← GetMSLevel(n_B)
9:             if ms_A = ms_B then                  ▷ Same-level: include all unique pairs
10:                RecordPair(n_A, n_B)
11:            else if ms_A < ms_B then ▷ Lower→higher: check descendant relation
12:                if n_B ∈ D(n_A) then
13:                    RecordPair(n_A, n_B)
```

Helper functions: BFS(T): breadth-first search on tree T; ComputeDescendants(n): Returns the set of descendants of node n; GetMSLevel(n): retrieve MS level from metadata; RecordPair(n_A, n_B): store the pair (n_A, n_B).

All hierarchical pairs distribution comparisons

After downsampling, we performed a *Kolmogorov–Smirnov* (KS) test [146] to evaluate the normality of the cosine similarity distributions. The KS test compared each empirical distribution with a normal distribution. The results indicate that none of the fragmentation levels relationships exhibit a normal distribution.

We then performed *Mann–Whitney U* tests [147] to determine whether the cosine similarity distributions differ significantly between node pairs. The resulting p-values were minimal, and even after applying a strict Bonferroni correction [148] (adjusted $\alpha \approx 0.001111$ from an initial 0.05), every pairwise comparison was statistically significant. This indicates that each fragmentation level comparison yields a distinctly different cosine similarity distribution.

Next, we computed the *Rank-Biserial Correlation* [149] to quantify the effect size between the groups. The analysis yielded predominantly moderate to high correlation values, indicating a strong effect and confirming that the distributions of cosine similarity scores differ markedly between the compared node pairs.

We further evaluated whether a parametric distribution could model the cosine similarity scores. Using both the KS statistic and

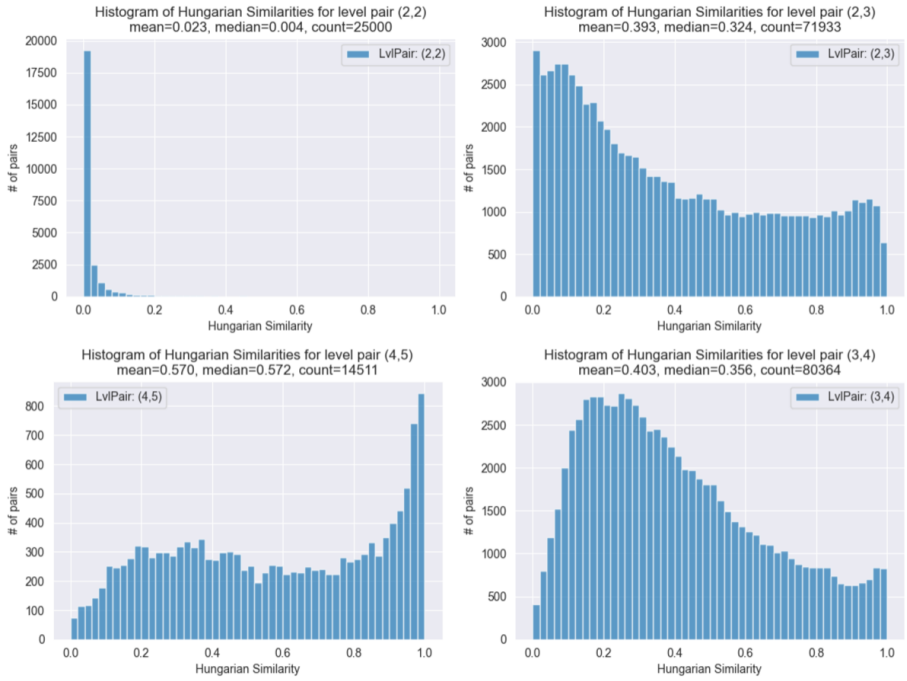

Figure 6.1: Cosine similarity distributions for node pairs (2,2), (2,3), (3,4), and (4,5) extracted from MSn trees. Notably, (2,2) pairs have a mean similarity of 0.023, while (4,5) pairs have a mean of 0.570. The number of pairs varies significantly across these categories.

Akaike Information Criterion [150] (AIC), we compared candidate distributions: Normal, Exponential, Beta, Gamma, Log-Normal, Uniform, Weibull (min), Weibull (max), Pareto, Student's t, and Cauchy, but found that none provided a dominant fit to the data.

Intra-group investigation

In our intra-group comparisons, we analyze the cosine similarity distributions for node pairs at the same fragmentation level, specifically, (2,2), (3,3), (4,4), and (5,5) pairs. We employ the *Kruskal–Wallis H-test* [151], a non-parametric method, to assess whether the similarity score distributions differ significantly among these groups. The test results confirm that not all groups share the same distribution.

We computed the *Spearman rank correlation* [152] between hierarchy levels and similarity scores to assess the presence of a monotonic relationship. The analysis yielded a correlation coefficient of 0.368 ($p < 0.001$), indicating that higher hierarchy levels are generally associated with increased similarity (see Fig. 6.2).

Inter-group investigation

In our inter-group analysis, we examined node pairs that span different fragmentation levels, namely (2,3), (2,4), (2,5), (3,4), (3,5), and (4,5). We introduced a variable, the *level difference*, which quantifies the absolute difference between the hierarchical positions of node pairs (e.g., a (2,5) pair has a level difference of 3). We then examined how this level difference relates to the cosine similarity scores between spectra. Using Spearman's Rank Correlation, we obtained a coefficient of -0.639 ($p < 0.001$), indicating that as the level difference increases, the spectral similarity tends to decrease significantly.

For further insight, we subdivided the inter-group comparisons into two subsets based on the lower tree level. Subset 1, consisting of pairs (2,4) and (3,4), exhibits a moderate negative correlation of -0.550, while Subset 2, comprising pairs (2,5), (3,5), and (4,5), shows a stronger negative correlation of -0.731. This analysis indicates that as the level difference increases, the decline in cosine similarity becomes more pronounced, suggesting that greater differences in fragmentation depth result in more distinct mass spectra profiles (see Fig. 6.2).

6.3.2 Cosine similarity on DreaMS embeddings

We conducted the same multi-stage hierarchical investigation of DreaMS embeddings, although originally trained on MS2 spectra, product ions are acquired independently, making its application to multi-stage data meaningful [13]. In contrast to the raw mass spectra, cosine similarities on DreaMS embeddings are more concentrated and exhibit a bell-shaped distribution across all levels of comparison (see Figures 6.3 and A.13).

As before, each pair type was randomly downsampled to 14,511 examples to mitigate frequency biases.

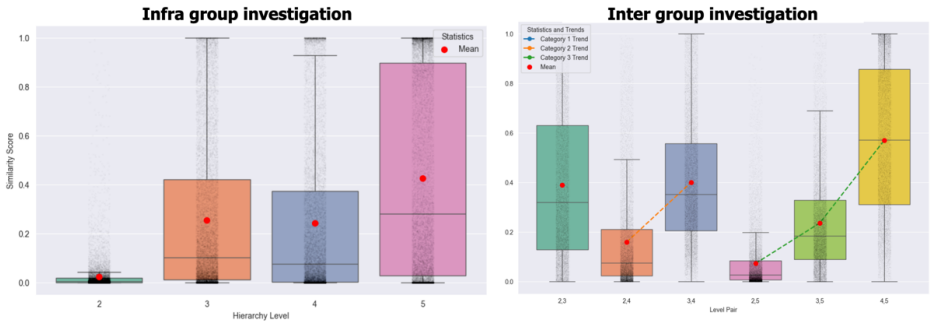

Figure 6.2: Similarity measurements for both intra- and inter-group comparisons. The left panel shows that spectra at deeper fragmentation levels exhibit higher cosine similarity, while the right panel indicates that as the difference between fragmentation levels increases, cosine similarity declines more sharply. Specifically, spectra at fragmentation depths 4 and 5 maintain a cosine similarity of ≈ 0.55, whereas comparisons to the same compound's spectra at stage 2 to 5 yield only ≈ 0.1.

All hierarchical pairs distribution comparisons

To mirror our analysis on raw spectra, we evaluated the distributional characteristics of cosine similarity scores computed from DreaMS embeddings.

First, the *Kolmogorov–Smirnov* (KS) test [146] revealed that none of the fragmentation levels exhibited a normally distributed pattern in their cosine similarity scores.

Next, *Mann–Whitney U* tests [147] were performed to compare the distributions across different node pairs. The resulting p values were minimal and almost all pairwise comparisons were statistically significant. The only exceptions were the comparisons between $(2,4)$ vs. $(2,5)$ and $(3,3)$ vs. $(4,4)$, which did not show significant differences.

Furthermore, the *Rank-Biserial Correlation* [149] analysis indicated moderate to strong effect sizes, suggesting that distributions of cosine similarities differ considerably.

Lastly, assessments using both the KS statistic and the *Akaike Information Criterion* (AIC) [150] showed that none of the candidate parametric distributions (including Normal, Exponential, Beta, Gamma,

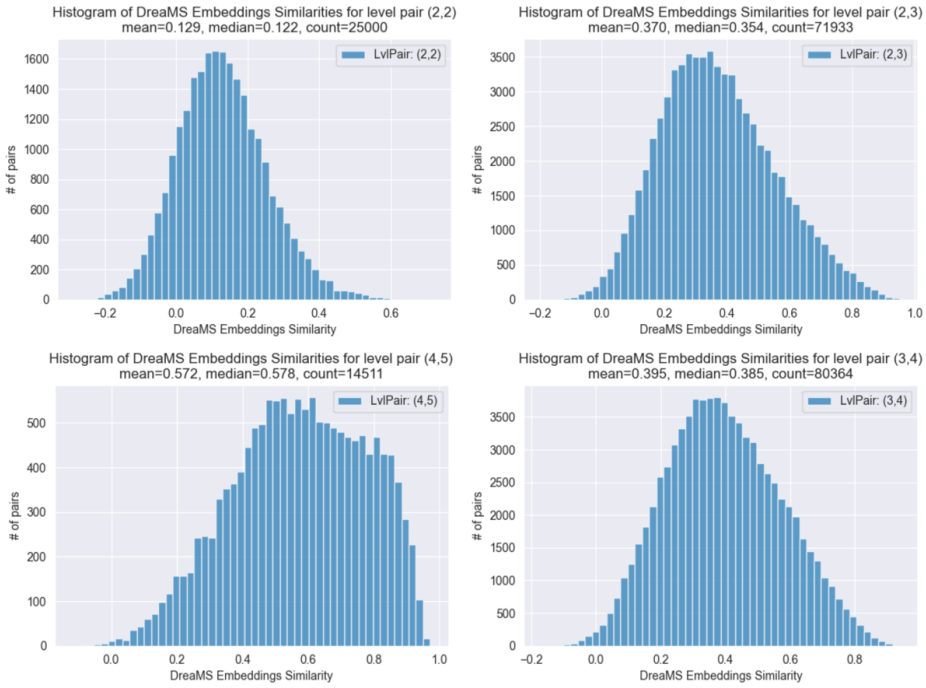

Figure 6.3: The figure illustrates the cosine similarity distributions for node pairs based on DreaMS embeddings. Unlike the raw spectra distributions, these curves are more centered, exhibiting a bell-shaped form. Similar to raw mass spectra, the skew toward higher similarity with higher fragmentation depth is also observed.

Log-Normal, Uniform, Weibull, Pareto, Student's t, and Cauchy) provided a dominant fit for the data.

Intra-group investigation

In our intra-group analysis using *DreaMS embeddings*, the *Kruskal–Wallis H-test* [151] confirmed significant differences among groups, mirroring the findings from analysis of raw spectra.

Furthermore, *Spearman rank correlation* [152] yielded a coefficient of **0.529** ($p < 0.001$), compared to **0.368** ($p < 0.001$) with raw spectra. As before, higher fragmentation levels are associated with increased cosine similarity, an effect more pronounced in DreaMS embeddings.

Inter-group investigation

Parallel to raw spectra, we applied *Spearman's Rank Correlation* to the DreaMS embeddings. The analysis produced a coefficient of -0.472 ($p < 0.001$), indicating that as the level difference increases, cosine similarity tends to decrease. This negative correlation is slightly weaker than the -0.639 observed with raw spectra.

For further insight with DreaMS embeddings, we subdivided the inter-group comparisons into two subsets. Subset 1, comprising pairs (2,4) and (3,4), exhibits a moderate negative correlation of -0.490, whereas Subset 2, consisting of pairs (2,5), (3,5), and (4,5), shows a stronger negative correlation of -0.659. The cosine similarity declines more sharply in Subset 2 as the level difference increases, consistent with the pattern observed on raw spectra.

6.3.3 Comparison of raw spectra vs DreaMS spectra representation

To assess the ability of our spectral representations to capture molecular similarity, we constructed triplets for each molecule comprising its *raw MS2 spectrum*, the corresponding *DreaMS embedding*, and the *molecular fingerprint*.

We computed the cosine similarity for both the raw spectra and the DreaMS embeddings and compared these distributions to the **Tanimoto similarity** similarity distribution as a ground truth measure. The results demonstrate that the cosine similarities obtained from DreaMS embeddings align more closely with Tanimoto similarity, whereas similarities on raw spectra are heavily skewed toward zero (see Figure 6.4).

6.3.4 DreaMS MSn clustering

We further investigated the behavior of *DreaMS embeddings* on MSn data using *UMAP*. UMAP projects high-dimensional data into a low-dimensional space while preserving both local and global structures. Surprisingly, the visualization reveals one dominant cluster alongside several smaller peripheral clusters, which is notable given that DreaMS was not explicitly trained on such spectra. When we colored the clusters by fragmentation depth, we observed that most of the

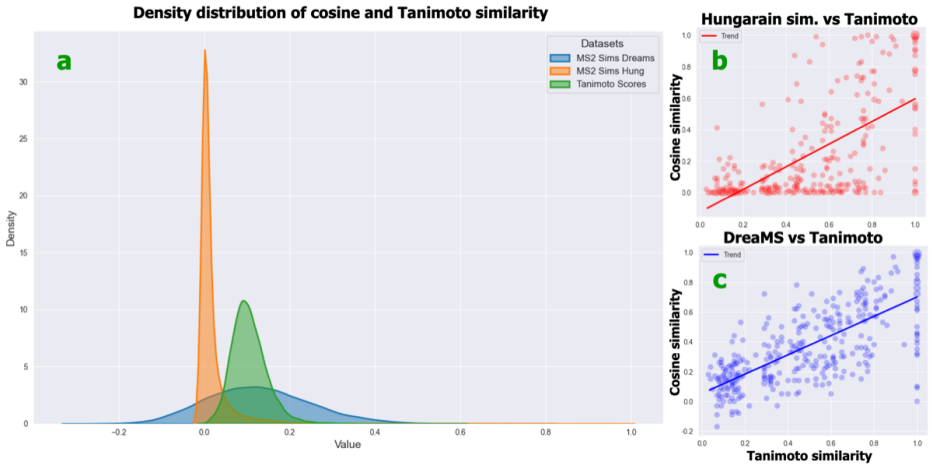

Figure 6.4: Comparison of raw spectra and DreaMS embeddings for molecular similarity. Panel (a) shows the density distributions for Tanimoto similarity (ground truth), Hungarian cosine similarity computed from raw spectra, and cosine similarity based on DreaMS embeddings across the entire dataset. Panel (b) displays scatter plots of eight uniformly distributed Tanimoto similarity bins (each consisting of 50 randomly sampled examples) with cosine similarity from raw spectra on the vertical axis. Panel (c) presents the same binned sampling but with cosine similarity based on DreaMS embeddings. Notably, while the raw spectra cosine similarities in Panel (b) are heavily skewed toward 0, the distribution in Panel (c) aligns much more closely with the true Tanimoto similarity, highlighting the better performance of DreaMS embeddings in capturing molecular similarity.

peripheral clusters predominantly consist of spectra from MS4 and higher levels (see Figure 6.5).

Upon further analysis, we found that the peripheral clusters could largely be distinguished by their spectral characteristics, specifically, these spectra exhibit fewer peaks (≈ 5) and a lower average ion mass (≈ 100 Da).

When we filtered the data to include only spectra with fewer than 5 peaks and an average ion mass below 100 m/z, the clustering became noticeably more granular. Each resulting cluster exhibited a homogeneous precursor mass, even though the mean precursor mass differed

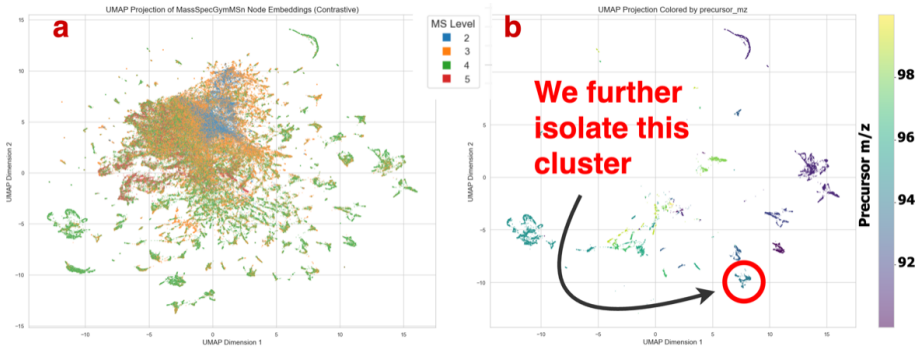

Figure 6.5: UMAP visualizations of MSn data using DreaMS embeddings. Panel (a) shows the UMAP projection for all data points, colored by fragmentation level. Panel (b) presents the UMAP projection for spectra filtered to include only those with at most 5 peaks, an average m/z below 100, and a precursor m/z below 100, with points colored by precursor m/z. This filtered view reveals distinct clusters with homogeneous precursor masses, and we pick one for further analysis.

between clusters. This consistency within clusters suggests that these groups likely represent small fragments with similar properties, potentially acting as fundamental building blocks of the original molecules (see Figure 6.5).

To further test this hypothesis, we isolated a single cluster. The spectra in this cluster demonstrated exceptionally high similarity: using the Hungarian cosine measure on raw spectra, similarity scores ranged from 0.9 to 1.0, while DreaMS cosine similarity values were between 0.8 and 1.0. Randomly selected examples within the cluster showed near-identical spectral features (see Figure 6.6). These findings strongly support that DreaMS is capable of capturing the subtle nuances in MSn data that enable the formation of distinct, homogeneous clusters, a feature that is much more challenging to achieve using raw spectra.

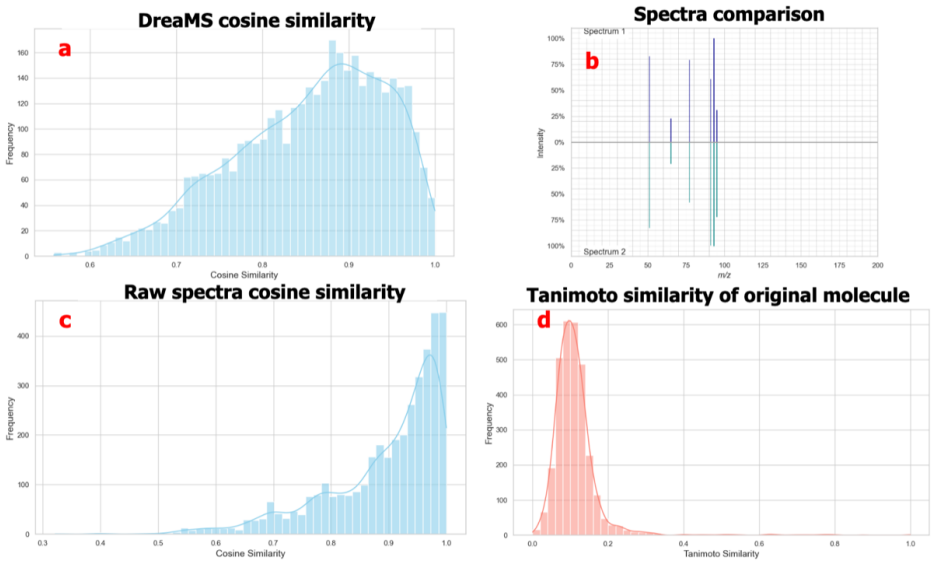

Figure 6.6: Investigated cluster analysis from Figure 6.5. Panel (a) displays the cosine similarity distribution computed on DreaMS embeddings, with scores predominantly above 0.8. Panel (b) shows representative spectra randomly selected from the cluster, which are nearly identical. Panel (c) presents the cosine similarity distribution based on raw spectra using the Hungarian method, again with scores mostly exceeding 0.8. In contrast, Panel (d) illustrates the Tanimoto similarity distribution of the underlying molecules, with values falling below 0.2. These findings indicate that, although the spectra within the cluster exhibit high similarity, the corresponding molecules are chemically diverse.

6.4 Internal representations and explainability

Throughout our experiments, we consistently observed two key trends: (1) incorporating deeper MS levels as input enhances predictive accuracy, and (2) spectra from higher fragmentation stages exhibit increasing similarity. These findings suggest that the models may be leveraging different internal mechanisms depending on the input structure. Therefore, we employ representational similarity analysis to investigate how different MS-level inputs shape the hidden space of the model. To isolate the effect of input types, all models were initialized

identically and trained under the same conditions (see Section 5.1), with input being the only varying factor.

Standard probes like *Canonical Correlation Analysis* (*CCA*) [153] (and its orthogonally or scale-limited variants [154]) detect only linear overlaps and can miss critical geometric distinctions when layer widths exceed sample sizes. As a result, these methods flatten critical distinctions, making them unsuitable for detailed representation analysis in deep models [155].

To address the shortcomings of previous methods, we use **Centered Kernel Alignment** (**CKA**) [155], which uses representational-similarity matrices to compare the geometry of activations rather than raw feature vectors. CKA remains stable when feature dimensions exceed the sample size[3], and remain sensitive to principal variance directions. Importantly, it is invariant only to orthogonal transforms and uniform scaling.

We apply CKA to compare models trained with four distinct input encodings across varying MSn tree depths. CKA is particularly well suited for this setting: it enables us to quantify and visualize how different spectral tree representations shape the internal representation of learned features. To complement these insights, we also perform eigenvalue analyses on the representation matrices to further understand how variance is distributed within the learned spaces.

6.4.1 Analysis of CKA heatmaps on Retrieval challenge

To visualize how internal representations evolve across network depth and input encoding, we re-implemented the original CKA framework [157] and generated layer-by-layer heatmaps of representational similarity (see Figure 6.7). For each model, we extracted activations from every layer and applied *global mean pooling* (*max pooling*, which produced nearly identical results) to reduce each graph-structured output into a fixed-size feature vector per example, necessary because graph sizes vary across inputs. Then for each layer, we computed a *Radial Basis Function* (RBF) kernel Gram matrix [158] to capture nonlinear relationships between examples, then applied CKA to assess similarity between layers. We exclusively use examples from the test

3. This is achieved by normalization with the Hilbert–Schmidt independence criterion, also known as the kernel-based conditional dependence measure [156].

fold of our dataset. A CKA score close to 1 indicates that two layers organize the data similarly, CKA$\approx$ 0 means orthogonal representations.

The resulting heatmaps (see Figure 6.7) are symmetric matrices where each cell reflects the similarity between two layers. Off-diagonal cells reveal how long feature subspaces persist through the network.

Similarity persists in the skip-connected dense block, whereas graph layers remain largely unaligned (CKA ≈ 0). Within the dense block, CKA decays with layer distance, but skip-connected pairs remain noticeably more similar than non-residual neighbors. This suggests that skip connections help preserve feature subspaces across depth, even as the network continues to refine them.

Notably, in the Standard challenge, models trained with only the MS2 level (maximum fragmentation stage 0) produced significantly more similar upper-layer representations, with CKA scores reaching up to 0.7. In contrast, when deeper fragmentation trees were included, maximum similarity dropped to approximately 0.25.

This pattern was much less pronounced in the Bonus challenge, where similarity across depths remained more uniform.

In the GNN block, representation similarity was generally low and inconsistent, perhaps influenced by the lack of soft pooling operations. However, the first GNN layer stood out as relatively more stable across input encodings, consistently showing higher similarity than the subsequent two layers.

Unlike prior vision-model studies [159, 160], our retrieval task yields more diffuse CKA maps, reflecting the inherent complexity of mass-spectral annotation and strictly unsolved problems.

6.4.2 Analysis of CKA heatmaps on De Novo challenge

We applied the same CKA pipeline (Section 6.4.1) to de novo models (Section 5.2).

In de novo, decoder layers exhibit uniform CKA $\approx$ 0.7˜0.8 across fragmentation depths, unlike Retrieval's, where a high similarity value was observed only on a model trained on MS2 (see Figure 6.7. We also observe parallel similarity bands: heads at the same position across layers align more strongly, though overall CKA decays with inter-layer distance

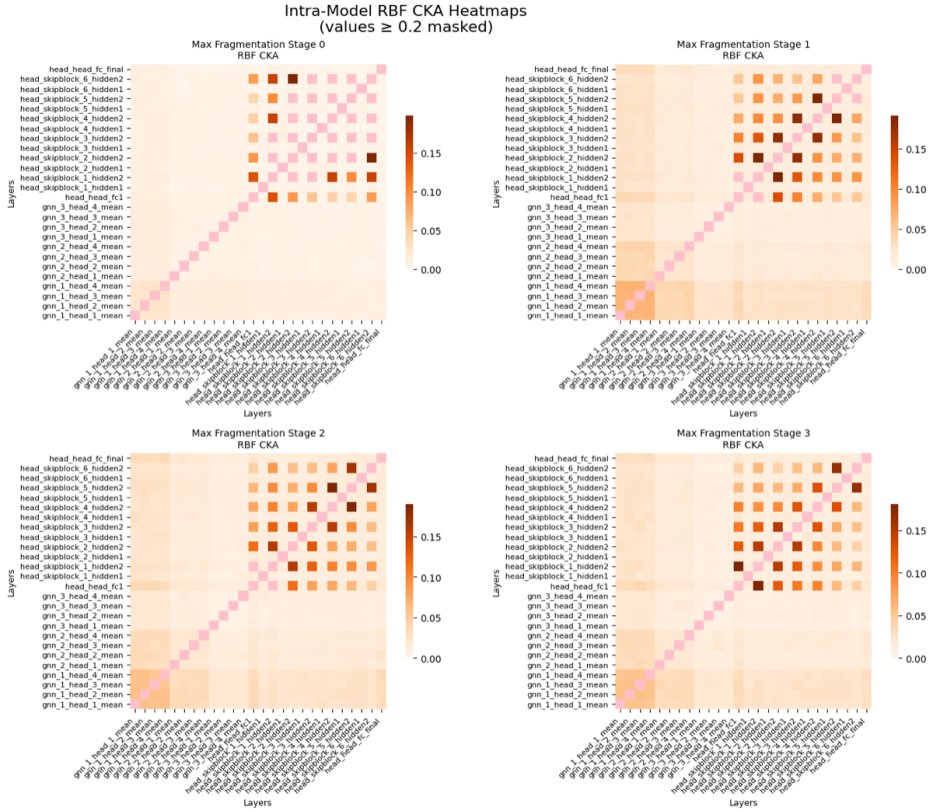

Figure 6.7: Intra-model CKA heatmaps for four models, each trained on a different maximum fragmentation stage using the DreaMS spectrum representation for the Standard challenge. Each heatmap shows pairwise representational similarity across layers, from input (bottom-left) to output (top-right). To ensure comparability, each heatmap is independently scaled to its own maximum CKA value, preventing models with higher absolute similarity from overshadowing subtler patterns in others. Pink regions highlight layer pairs with similarity above 0.2, emphasizing moderately aligned representations while still visualizing more nuanced differences that would otherwise be obscured.

In the graph attention encoder, we observe similar trends as in the Retrieval challenge: the first GNN layer consistently shows the highest similarity across input encodings, with CKA values reaching up to 0.2, lightly higher than the ≈ 0.1 observed previously. Interestingly, we also detect a modest degree of similarity in the deeper GNN layers, which was largely absent in the Retrieval models.

As in Retrieval, the first GNN layer leads (CKA ≈ 0.2 vs. ≈ 0.1 before), but unlike Retrieval, deeper GNN layers now show a modest residual similarity.

Similarly to retrieval, de novo CKA heatmaps remain low and diffuse, with no apparent differences across the four fragmentation stages (see Figure 6.8). The lack of pronounced similarity structure may reflect the current task complexity and room for neural network architectural improvements, which we further investigate in the following sections.

6.4.3 Retrieval model representation comparison with effective rank

To probe how fragmentation depth and input encoding affect model complexity, we computed each layer's effective rank (See Figure 6.9), revealing whether deeper spectra expand representational subspaces or induce bottlenecks.

We formed per-layer activation matrices (rows representing test examples and columns representing feature dimensions), subtracted column means, and performed principal component analysis (PCA) [90] to obtain eigenvalues. We then summarized dimensionality via the *effective rank* [161], where, e.g., a rank ≈ 2 indicates that two principal components capture nearly all variance from a 1,024-dimensional layer.

First, we observe (see Figure 6.9) that GNN encoder leads in dimensionality: binned-spectra models peak at ~ 150 effective dimensions in layer 1 and then collapse. DreaMS-based models decline gradually, implying richer input subspaces that the graph layers retain better.

Second, adding any deeper stage (MS3–MS5) boosts effective rank several-fold over MS2-only models, beyond MS3, the rank profiles converge showing that the leap from two to three stages unlocks most

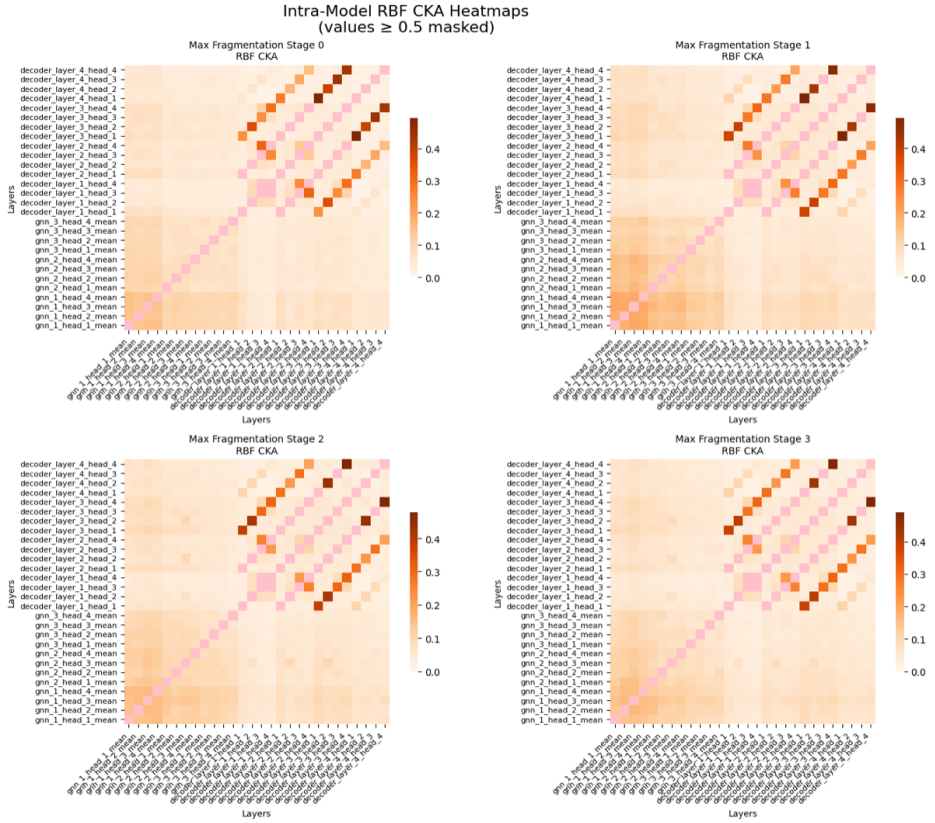

Figure 6.8: Intra-model CKA heatmaps for four models trained on different maximum fragmentation stages using the DreaMS spectrum representation in the De Novo challenge. Each heatmap depicts pairwise representational similarity across layers, from input (bottom-left corner) to output (top-right corner). To maintain comparability, each heatmap is individually scaled to its own maximum CKA value. Pink regions highlight layer pairs with similarity above 0.5, allowing visualization of more nuanced differences that would otherwise be obscured.

complexity (noting uneven tree depths representation in the dataset, see Section 4.6).

Finally, models trained on DreaMS embeddings consistently outrank binned spectra, and incorporating molecular formula in the Bonus task further elevates effective rank in upper dense layers (Figure 6.9). These findings show that richer input encodings and deeper fragmentation improve retrieval task metrics and drive networks to occupy larger, dimensional feature subspaces.

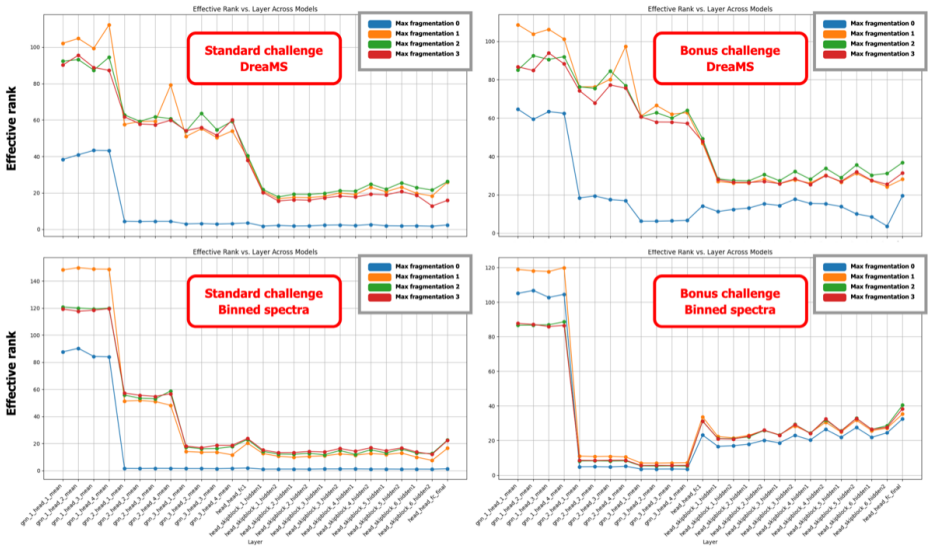

Figure 6.9: Effective rank across model layers for four input representations. Each subplot corresponds to a different combination of task (Standard, right plots vs. Bonus challenge, left plots) and input encoding (DreaMS spectra representation on top, binned spectra on bottom). Within each plot, the x-axis represents network layers from left to right, starting with the GNN input layers and ending with the dense output layers. The y-axis shows the average effective rank for each layer, indicating the intrinsic dimensionality of the learned representation. Each line represents one of four models trained on a different maximum fragmentation stage (MS2, MS3, MS4, MS5), with individual points showing the average effective rank at each layer.

6.4.4 De novo model representation comparison with effective rank

We applied the same effective-rank analysis to our de novo models and trends in de novo are far more muted than in retrieval. DreaMS models still outrank binned spectra, with a gentler decay through the GNN encoder, but any gap between MS2 and deeper stage models disappears.

All De Novo variants exhibit a *severe bottleneck* at the GNN encoder to the Transformer decoder boundary (see Figure 6.10): effective rank falls below ~ 9 of 1,024 dimensions, compressing almost all information into a tiny subspace before decoding. Such extreme compression implies the decoder leans more on the decoder's language modeling and past tokens than on the encoded MSn structure.

6.4.5 Retrieval model representation comparison with top eigenvectors similarity

Building on our effective-rank analysis (Section 6.4.3), we assessed whether principal feature directions align across models by comparing the top 30 eigenvectors per layer. For each model pair, we project one set of eigenvectors into the other's space and compute *cosine similarities* (Fig. 6.11), revealing how input encoding and fragmentation depth affect principal subspace preservation. Specifically, for each input type: raw spectra $\pm$ bonus and DreaMS $\pm$ bonus, we pooled MS2-MS5 variants to obtain group level alignment profiles.

Our top-eigenvector analysis reveals several consistent trends. First, the initial GNN layer aligns strongly (cosine ≈ 0.8) across all benchmark tasks, indicating a fragmentation depth agnostic feature extractor. Beyond layer 1, alignment decays, but DreaMS models consistently preserve higher similarity than binned spectra variants.

Second, the bonus challenge (incorporating molecular formula) boosts alignment upon entering the upper dense network, especially for binned-spectra models, which show the most significant gain. The top eigenvalues of models trained on DreaMS embeddings gain less (Fig. 6.11), suggesting they continue to be more influenced by including more fragmentation stages.

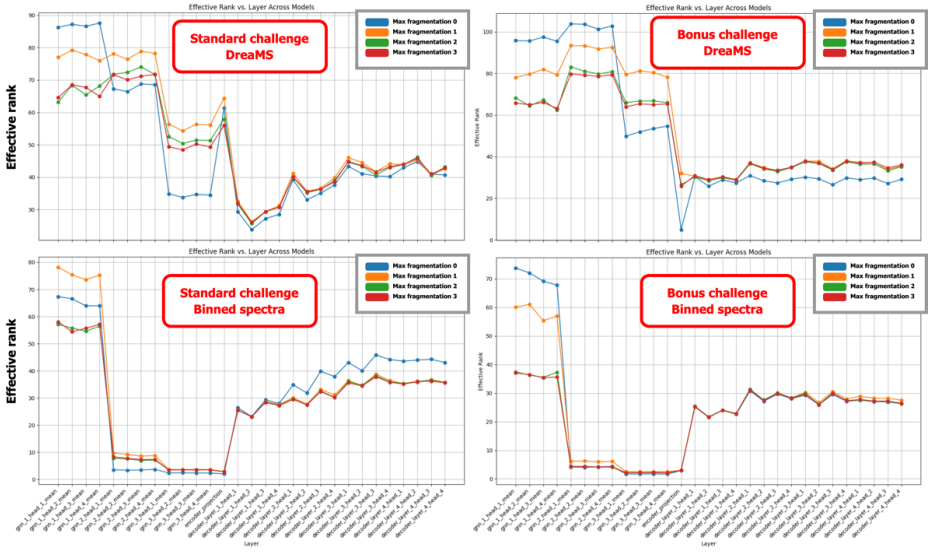

Figure 6.10: Effective rank across layers of the De Novo models for both Standard (right) and Bonus (left) challenges, comparing DreaMS (top) and binned-spectra (bottom) encodings. The x-axis spans layers from the GNN encoder (left) through to the Transformer decoder output (right), while the y-axis plots the average effective rank at each stage. Each curve corresponds to a model trained on a different maximum fragmentation depth (MS2–MS5). Note the steep collapse in dimensionality at the encoder–decoder boundary for all fragmentation levels and encodings, a bottleneck that forces nearly all information into a minimal space before sequence generation.

6.4.6 De novo model representation comparison with top eigenvectors similarity

Applying our top-eigenvector alignment analysis to the De Novo models (see Figure 6.11) reveals both familiar and distinct patterns compared to retrieval. In the GNN encoder, layer 1 aligns at ≈ 0.7 across inputs. Alignment then plummets, most sharply for binned spectra, while DreaMS models sustain higher consistency deeper in the encoder.

However, at the encoder–decoder boundary, all models exhibit a rebound, with mean cosine scores climbing above **0.8** and remaining flat throughout the Transformer decoder. This effect is most pronounced

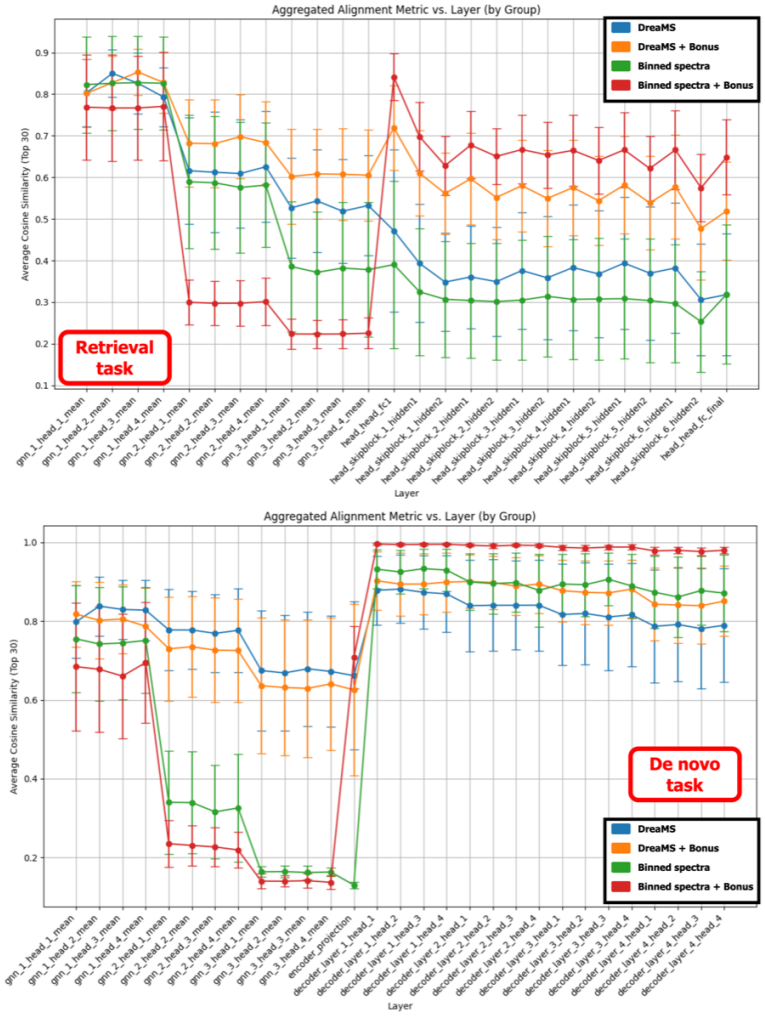

Figure 6.11: Group-wise alignment of the top-30 eigenvectors across network layers. The figure consists of two panels: the top panel shows Retrieval models, and the bottom panel shows De Novo models. Each curve represents the average cosine similarity along with its standard deviation, pooled across all fragmentation stages (MS2–MS5) for each input type (raw spectra, raw + bonus, DreaMS embeddings, and DreaMS + bonus). The x-axis indexes the network layers, progressing from the GNN encoder (left) to the dense output layers (right) for Retrieval models, and from the GNN encoder through the Transformer decoder for De Novo models. The y-axis reports the mean alignment within each group. Continuous curves illustrate how principal subspaces are preserved, or collapse, through the network under different input conditions.

in the binned spectra representation in the bonus challenge, where alignment reaches nearly **1.0** across decoder layers, indicating that the top-30 eigenvectors are virtually unchanged once decoding begins.

This rebound underscores the severe *information bottleneck*[4] from the effective-rank analysis: de novo models lean on the generative decoder and, in bonus challenge, also on molecular formula knowledge, over exploiting the MSn tree encoding. These findings highlight opportunities for future improvement, particularly by encouraging more effective use of MSn information within the de novo model architecture.

Code, Data and Reproducibility

All code, data, and subprojects developed during this thesis are open-source and freely available. The MassSpecGymMSn benchmark supports fully reproducible pipelines and is publicly accessible:

- **Benchmark code & documentation:**
  github.com/Jozefov/MassSpecGymMSn
- **Benchmark dataset on HuggingFace:**
  huggingface.co/datasets/Jozefov/MassSpecGymMSn
- **Models & model pipelines examples:**
  github.com/Jozefov/PhantoMS
- **Molecular class annotation with PyClassyFire:**
  github.com/Jozefov/PyClassyFire

4. As defined in Section 6.4.4

7 Conclusions and Future work

In this work, we introduce MassSpecGymMSn, the first open, machine learning ready benchmark built on multi-stage fragmentation MSn data. Comprising 183,294 spectra across 14,008 unique compounds, our resource fills a critical gap in computational mass spectrometry. This unprecedented dataset captures fragmentation stages up to MS5, whereas existing datasets are almost exclusively restricted to MS2. Moreover, it predominantly covers compounds absent from any current open or proprietary MS2 or MSn repository. Importantly, it supports reproducible machine learning research in mass spectrometry by providing standardized retrieval and de novo molecular generation challenges, with carefully controlled data splits to prevent information leakage. Prior to this work, no publicly available MSn datasets of comparable scale and quality existed.

Alongside this benchmark, we developed the first neural network models specifically trained to operate on MSn spectra trees. Our Graph Neural Network (GNN) encoders treat spectra not as flat collections of peaks, but as true hierarchical structures, mirroring the sequential fragmentation processes inherent to MSn experiments. To our knowledge, this is the first time neural networks have been designed and trained to exploit the full depth of multi-stage mass spectrometry data, opening new analytical possibilities that were previously inaccessible.

In our study, we explored two distinct approaches to representing MSn spectra. The first used a classical method, encoding each spectrum as a simple binned peak vector. The second applied the DreaMS foundation model to generate dense, learned embeddings for each spectrum. This setup enabled a direct comparison between traditional handcrafted feature spaces and modern learned representations in the context of MSn data. Importantly, our work marks a novel step in extending foundation models beyond MS2, demonstrating their potential to capture the deeper structure revealed by multi-stage fragmentation.

Our experiments show that adding even a single additional fragmentation stage, moving from MS2 to MS3, yields dramatic gains in both retrieval and generation performance. In the standard retrieval task, the Hit Rate@1 jumps from 0.012 at MS2 to 0.079 at MS3, a 6.6-fold

increase, and continues rising to 0.091 at MS5. When using DreaMS foundation model embeddings, the improvement is even more striking: the Hit Rate@1 reaches 0.112 at MS4, representing a nearly 10x increase over the MS2 baseline achieved with simple binned spectra.

Moreover, our detailed analysis of spectral similarities within MSn trees confirms that each additional fragmentation stage reveals previously hidden, rich relationships. Comparing spectra similarities across fragmentation stages using cosine similarity, we observe that the average similarity rises from just 0.02 for MS2–MS2 spectra pairs to 0.57 for MS4–MS5 comparisons, highlighting a dramatic gain in shared structural information. Nonparametric tests, including the Mann-Whitney U test and Kruskal-Wallis H test, further validate that each fragmentation stage contributes statistically distinct patterns, confirming that deeper fragmentation adds new information. Spearman correlation analyses additionally confirm a strong positive trend between fragmentation depth and spectral similarity. DreaMS embeddings mirror these patterns, with similarity distributions becoming more tightly centered and mean values rising from 0.13 at MS2 to 0.57 at MS5, further capturing the unique and orthogonal information each deeper fragmentation stage contributes.

Quantitative internal model analyses further reinforce the value of MSn data. Centered Kernel Alignment (CKA) heatmaps, effective rank measurements, and top eigenvector alignments all confirm that deeper fragmentation stages enrich the internal feature spaces of neural networks. Models trained on MSn trees maintain richer, higher-dimensional representations compared to models trained solely on MS2 data.

Across both the standard and bonus retrieval challenges, models built on DreaMS embeddings consistently outperformed those using traditional binned spectra. At the MS3 level, for example, DreaMS achieved a Hit Rate@20 of 0.410 compared to 0.338 for binned spectra, representing a 23.5% relative improvement, along with lower molecular structural distances. Critically, internal representation analyses showed that models trained on DreaMS embeddings maintain high internal consistency and dimensional richness across network layers, whereas binned-spectra models degrade substantially after the initial stages.

Remarkably, we found that DreaMS embeddings, despite the foundation model being trained only on MS2 data, naturally organize MSn spectra into meaningful fragmentation-stage clusters without supervision. UMAP projections revealed a distinct central cluster of spectra from early fragmentation stages, with peripheral clusters dominated by MS4 and MS5, indicating that DreaMS embeddings inherently capture hierarchical substructures present in MSn data. This unexpected generalization underscores the power of foundation models and highlights their potential for enabling deeper chemical understanding.

Altogether, this work establishes a new standard for computational mass spectrometry. By providing the first large-scale, public MSn benchmark and pioneering neural architectures designed to exploit its full richness, we open new avenues for data-driven metabolomics. In parallel, our laboratory developed a high-throughput experimental pipeline for multi-stage fragmentation measurements, achieving unprecedented speed and quality in compound characterization. This platform not only enabled the creation of MassSpecGymMSn but also positions us to dramatically expand the availability of high-quality MSn data in the near future. We aim to lower the barriers to entry for the machine learning community, reducing the need for deep domain expertise in mass spectrometry, and hope to spark broader interest in MSn over MS2 data. Given that nearly 98% of currently measured mass spectra remain unannotated [7] and often require extensive expert curation, MSn holds the promise to unlock new potential. With MassSpecGymMSn, machine learning models, and experimental innovations developed alongside it, we lay the groundwork that may catalyze the next generation of tools, tools as transformative for mass spectrometry as AlphaFold has been for structural biology, reshaping how mass spectra data are interpreted, modeled, and ultimately driving the discovery of new metabolites.

A An appendix

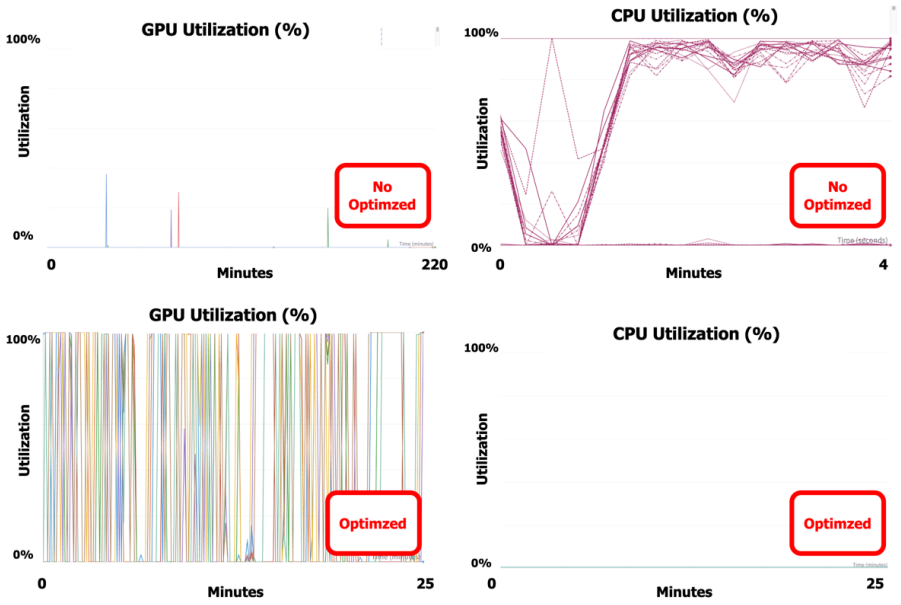

Figure A.1: The upper images illustrate an unoptimized computation scenario for the retrieval task. The upper-left image shows a 30-epoch run on 8 AMD MI250X GPUs, which took 220 minutes to complete; the GPUs are frequently idle due to CPU-bound batch processing, often showing 0% utilization. The upper-right image displays the CPU usage measured during a separate 4-minute run under the same conditions, demonstrating that 7 CPUs were constantly hitting 100% utilization. In contrast, the lower images demonstrate the optimized version, where GPU utilization consistently reaches 100% while CPU usage remains minimal.

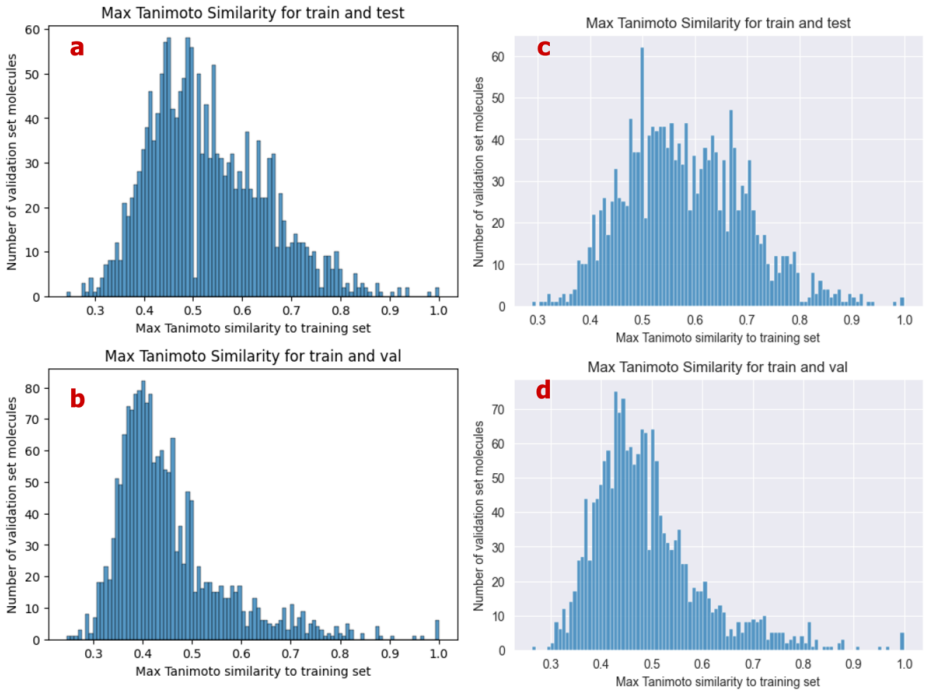

Figure A.2: This figure presents the Tanimoto similarity distributions for molecules used to train the de novo decoder, compared with the validation and test folds obtained via a Murcko histogram stratified split. Panels (a) and (b) represent molecules from a 1-million-compound database, with (a) comparing train set to test set and (b) to the validation set, while panels (c) and (d) show similar comparisons for molecules from a 4-million-compound database. In total, 2.6 million molecules form the training set, and here we confirm minimal data leakage between the splits.

Figure A.3: Training loss for retrieval, standard challenge with binned spectra

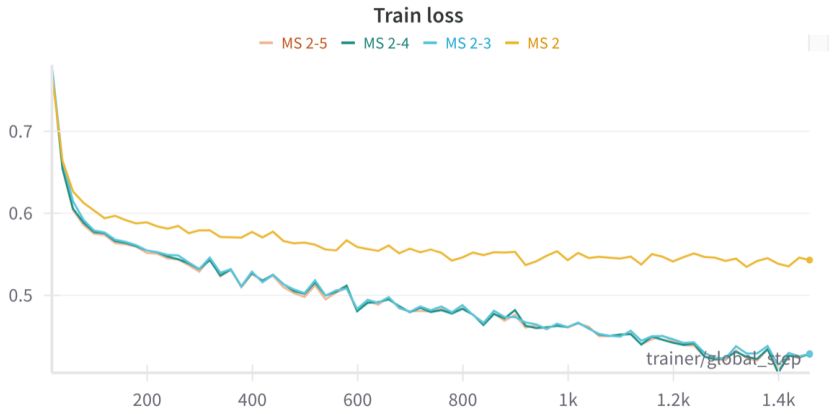

Figure A.4: Training loss for retrieval, standard challenge with DreaMS

Figure A.5: Training loss for retrieval, bonus challenge with binned spectra

Figure A.6: Training loss for retrieval, bonus challenge with DreaMS

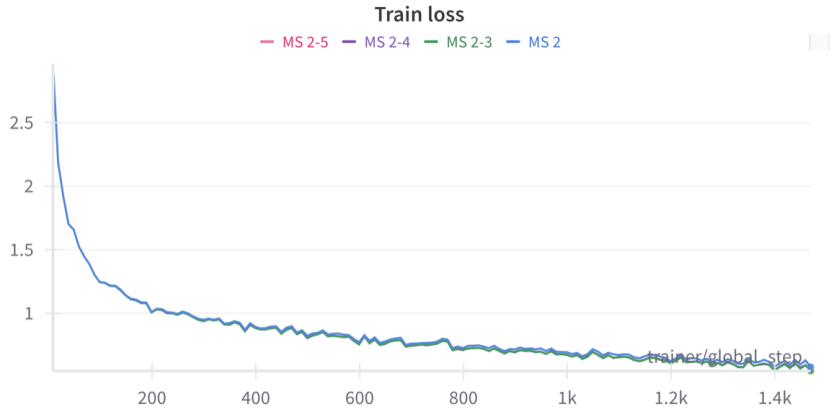

Figure A.7: Training loss for de novo, standard challenge with binned spectra

Figure A.8: Training loss for de novo, standard challenge with DreaMS

Figure A.9: Training loss for de novo, bonus challenge with binned spectra

Figure A.10: Training loss for de novo, bonus challenge with DreaMS

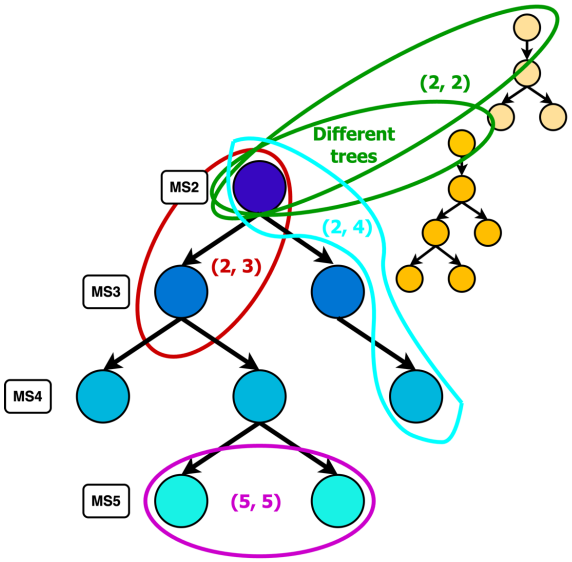

Figure A.11: Illustration of the node pair extraction process from multi-stage MSn trees. Circled pairs indicate comparisons constructed from the same tree between different fragmentation levels (e.g., a $(2, 3)$ pair) or within the same level (e.g., $(5, 5)$ pairs). Note that, since each tree contains only one MS2 node, $(2, 2)$ pairs are assembled from nodes in different trees.

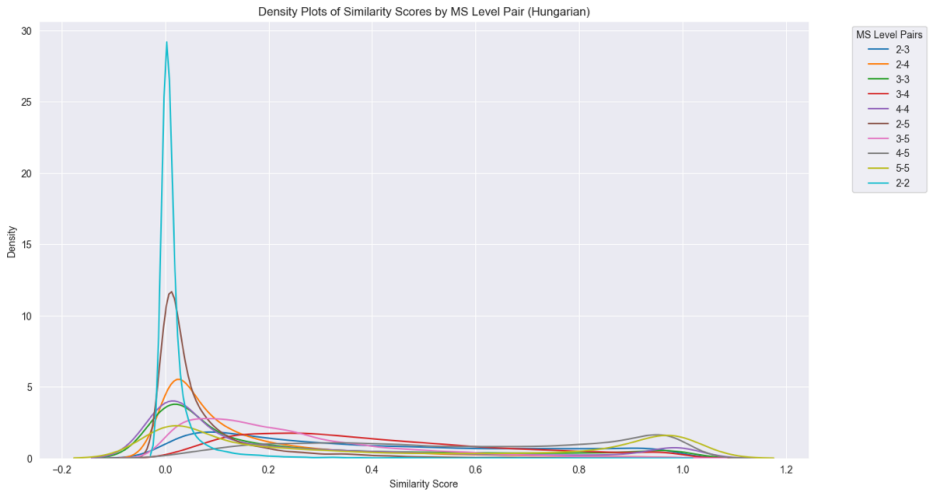

Figure A.12: Hungarian cosine similarity distribution on hierarchical pairs.

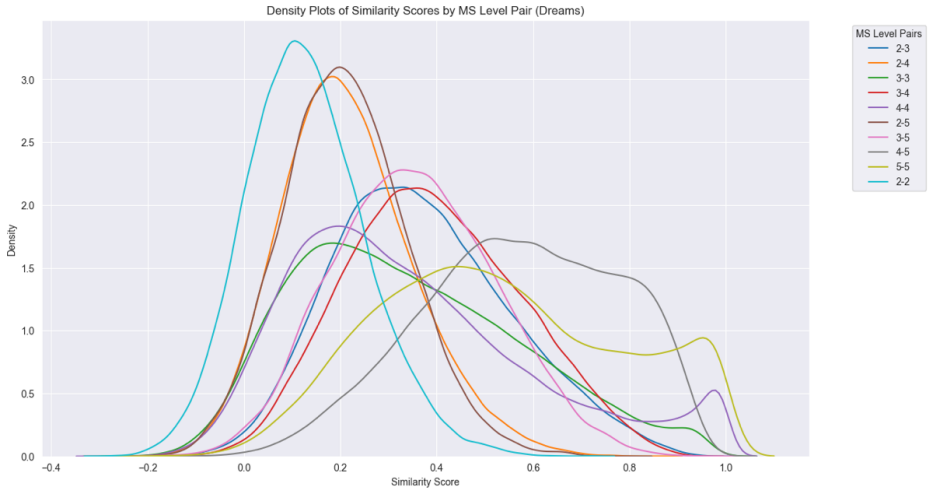

Figure A.13: DreaMS cosine similarity distribution on hierarchical pairs.

Bibliography

[1] Dinesh K Barupal et al. "MetaMapp: mapping and visualizing metabolomic data by integrating information from biochemical pathways and chemical and mass spectral similarity". en. In: *BMC Bioinformatics* 13.1 (May 2012), p. 99.

[2] Rima Kaddurah-Daouk and K Ranga Rama Krishnan. "Metabolomics: a global biochemical approach to the study of central nervous system diseases". en. In: *Neuropsychopharmacology* 34.1 (Jan. 2009), pp. 173–186.

[3] Mark R Viant and Ulf Sommer. "Mass spectrometry based environmental metabolomics: a primer and review". en. In: *Metabolomics* 9.S1 (Mar. 2013), pp. 144–158.

[4] Aihua Zhang et al. "Mass spectrometry-driven drug discovery for development of herbal medicine". en. In: *Mass Spectrom. Rev.* 37.3 (May 2018), pp. 307–320.

[5] Corinna Brungs et al. "Efficient generation of open multi-stage fragmentation mass spectral libraries". In: *ChemRxiv* (Oct. 2024).

[6] Ying Jin et al. "A new strategy for the discovery of epimedium metabolites using high-performance liquid chromatography with high resolution mass spectrometry". en. In: *Anal. Chim. Acta* 768 (Mar. 2013), pp. 111–117.

[7] Ricardo R da Silva et al. "Illuminating the dark matter in metabolomics". en. In: *Proc. Natl. Acad. Sci. U. S. A.* 112.41 (Oct. 2015), pp. 12549–12550.

[8] Chloe Engler Hart et al. "Defining the limits of plant chemical space: challenges and estimations". en. In: *Gigascience* 14 (Jan. 2025), giaf033.

[9] Kai Dührkop et al. "SIRIUS 4: a rapid tool for turning tandem mass spectra into metabolite structure information". en. In: *Nat. Methods* 16.4 (Apr. 2019), pp. 299–302.

[10] Montgomery Bohde et al. "DiffMS: Diffusion generation of molecules conditioned on mass spectra". In: *arXiv* [*cs.LG*] (Feb. 2025).

[11] Adam Hájek et al. "SpecTUS: Spectral Translator for Unknown Structures annotation from EI-MS spectra". In: *arXiv* [*cs.LG*] (Feb. 2025).

[12] *mzCloud – Advanced Mass Spectral Database*. https://www.mzcloud.org/. Accessed: 2025-3-23.

[13] Arpana Vaniya and Oliver Fiehn. "Using fragmentation trees and mass spectral trees for identifying unknown compounds in metabolomics". en. In: *Trends Analyt. Chem.* 69 (June 2015), pp. 52–61.

[14] Mingxun Wang et al. "Sharing and community curation of mass spectrometry data with Global Natural Products Social Molecular Networking". en. In: *Nat. Biotechnol.* 34.8 (Aug. 2016), pp. 828–837.

[15] John Jumper et al. "Highly accurate protein structure prediction with AlphaFold". en. In: *Nature* 596.7873 (Aug. 2021), pp. 583–589.

[16] Justin Gilmer et al. "Neural Message Passing for Quantum Chemistry". In: *arXiv* [*cs.LG*] (Apr. 2017).

[17] Kurt Hornik et al. "Multilayer feedforward networks are universal approximators". en. In: *Neural Netw.* 2.5 (Jan. 1989), pp. 359–366.

[18] William L Hamilton. *Graph representation learning*. en. Synthesis lectures on artificial intelligence and machine learning. Cham, Switzerland: Springer International Publishing, Sept. 2020.

[19] Roman Bushuiev et al. "Emergence of molecular structures from repository-scale self-supervised learning on tandem mass spectra". In: *ChemRxiv* (Apr. 2024).

[20] John H Beale et al. "Successful sample preparation for serial crystallography experiments". en. In: *J. Appl. Crystallogr.* 52.Pt 6 (Dec. 2019), pp. 1385–1396.

[21] Sepideh Amin-Hanjani et al. "Mevastatin, an HMG-CoA Reductase Inhibitor, Reduces Stroke Damage and Upregulates Endothelial Nitric Oxide Synthase in Mice". In: *Stroke* 32.4 (Apr. 2001), pp. 980–986.

[22] Gary Siuzdak. *Activity Metabolomics and mass spectrometry 2024 edition*. 2025.

[23] Yan Wang et al. "A "soft" and "hard" ionization method for comprehensive studies of molecules". en. In: *Anal. Chem.* 90.24 (Dec. 2018), pp. 14095–14099.

[24] *Mass Analyzers* (*Mass Spectrometry*). en. https://chem.libretexts.org/Bookshelves/Analytical_Chemistry/Supplemental_Modules_(Analytical_Chemistry)/Instrumentation_and_Analysis/Mass_Spectrometry/Mass_Spectrometers_(Instrumentation)/Mass_Analyzers_(Mass_Spectrometry). Accessed: 2025-3-30. Oct. 2013.

[25] Johan Viaene et al. “Comparison of a triple-quadrupole and a quadrupole time-of-flight mass analyzer to quantify 16 opioids in human plasma”. en. In: *J. Pharm. Biomed. Anal.* 127 (Aug. 2016), pp. 49–59.

[26] Jeffrey S Gaffney and Nancy A Marley. “Chemical Measurements and Instrumentation”. In: *General Chemistry for Engineers*. Ed. by Jeffrey S Gaffney and Nancy A Marley. Elsevier, 2018, pp. 493–532.

[27] Youzhong Liu et al. “Mass spectrometry-based structure elucidation of small molecule impurities and degradation products in pharmaceutical development”. en. In: *Trends Analyt. Chem.* 121.115686 (Dec. 2019), p. 115686.

[28] J L Holmes. “Metastable Ions”. In: *Encyclopedia of Spectroscopy and Spectrometry*. Ed. by John C Lindon et al. Elsevier, 2017, pp. 797–802.

[29] Emma L Schymanski et al. “Identifying small molecules via high resolution mass spectrometry: communicating confidence”. en. In: *Environ. Sci. Technol.* 48.4 (Feb. 2014), pp. 2097–2098.

[30] Jurgen H Gross. *Mass Spectrometry: A Textbook*. en. 3rd ed. Cham, Switzerland: Springer International Publishing, June 2017.

[31] P Waridel et al. “Evaluation of quadrupole time-of-flight tandem mass spectrometry and ion-trap multiple-stage mass spectrometry for the differentiation of C-glycosidic flavonoid isomers”. en. In: *J. Chromatogr. A* 926.1 (Aug. 2001), pp. 29–41.

[32] Ying S Ting et al. “Automated lipid A structure assignment from hierarchical tandem mass spectrometry data”. en. In: *J. Am. Soc. Mass Spectrom.* 22.5 (May 2011), pp. 856–866.

[33] Mohamed A Salem et al. “Metabolomics in the context of plant natural products research: From sample preparation to metabolite analysis”. en. In: *Metabolites* 10.1 (Jan. 2020), p. 37.

[34] Niek F de Jonge et al. “MS2Query: reliable and scalable MS2 mass spectra-based analogue search”. en. In: *Nat. Commun.* 14.1 (Mar. 2023), p. 1752.

[35] Sebastian Böcker and Kai Dührkop. “Fragmentation trees reloaded”. en. In: *J. Cheminform.* 8.1 (Feb. 2016), p. 5.

[36] Peiying Shi et al. “Characterization and identification of isomeric flavonoid O-diglycosides from genus Citrus in negative electrospray ionization by ion trap mass spectrometry and time-of-flight mass

spectrometry". en. In: *Anal. Chim. Acta* 598.1 (Aug. 2007), pp. 110–118.

[37] Nicolas Fabre et al. "Determination of flavone, flavonol, and flavanone aglycones by negative ion liquid chromatography electrospray ion trap mass spectrometry". en. In: *J. Am. Soc. Mass Spectrom.* 12.6 (June 2001), pp. 707–715.

[38] Lars Ridder et al. "Automatic compound annotation from mass spectrometry data using MAGMa". en. In: *Mass Spectrom. (Tokyo)* 3.Spec Iss 2 (July 2014), S0033.

[39] Florian Rasche et al. "Computing fragmentation trees from tandem mass spectrometry data". en. In: *Anal. Chem.* 83.4 (Feb. 2011), pp. 1243–1251.

[40] Kerstin Scheubert et al. "Computing fragmentation trees from metabolite multiple mass spectrometry data". en. In: *J. Comput. Biol.* 18.11 (Nov. 2011), pp. 1383–1397.

[41] Kerstin Scheubert et al. "Multiple mass spectrometry fragmentation trees revisited: Boosting performance and quality". In: *Lecture Notes in Computer Science*. Lecture notes in computer science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 217–231.

[42] Florian Rasche et al. "Identifying the unknowns by aligning fragmentation trees". en. In: *Anal. Chem.* 84.7 (Apr. 2012), pp. 3417–3426.

[43] Haiying Zhang et al. "Mass defect filter technique and its applications to drug metabolite identification by high-resolution mass spectrometry". en. In: *J. Mass Spectrom.* 44.7 (July 2009), pp. 999–1016.

[44] Samuel Goldman et al. "Generating molecular fragmentation graphs with autoregressive neural networks". en. In: *Anal. Chem.* 96.8 (Feb. 2024), pp. 3419–3428.

[45] Runzhong Wang et al. "Neural graph matching improves retrieval augmented generation in molecular machine learning". In: *arXiv* [*cs.LG*] (Feb. 2025).

[46] Adamo Young et al. "FraGNNet: A deep probabilistic model for mass spectrum prediction". In: *arXiv* [*cs.LG*] (Apr. 2024).

[47] Richard Licheng Zhu and Eric Jonas. "Rapid approximate subset-based spectra prediction for electron ionization-mass spectrometry". en. In: *Anal. Chem.* 95.5 (Feb. 2023), pp. 2653–2663.

[48] Michael A Stravs et al. "MSNovelist: de novo structure generation from mass spectra". en. In: *Nat. Methods* 19.7 (July 2022), pp. 865–870.

[49] Michelle T Sheldon et al. "Determination of ion structures in structurally related compounds using precursor ion fingerprinting". en. In: *J. Am. Soc. Mass Spectrom.* 20.3 (Mar. 2009), pp. 370–376.

[50] Jiarui Zhou et al. "HAMMER: automated operation of mass frontier to construct in silico mass spectral fragmentation libraries". en. In: *Bioinformatics* 30.4 (Feb. 2014), pp. 581–583.

[51] Brandon Y Lieng et al. "Computational expansion of high-resolution-MSn spectral libraries". en. In: *Anal. Chem.* 95.47 (Nov. 2023), pp. 17284–17291.

[52] Jennifer N Wei et al. "Rapid prediction of electron-ionization mass spectrometry using neural networks". en. In: *ACS Cent. Sci.* 5.4 (Apr. 2019), pp. 700–708.

[53] Wout Bittremieux et al. "Comparison of cosine, modified cosine, and neutral loss based spectrum alignment for discovery of structurally related molecules". en. In: *J. Am. Soc. Mass Spectrom.* 33.9 (Sept. 2022), pp. 1733–1744.

[54] *MassBank of North America*. en. `https://mona.fiehnlab.ucdavis.edu/`. Accessed: 2025-3-23.

[55] David S Wishart et al. "HMDB: the Human Metabolome Database". en. In: *Nucleic Acids Res.* 35.Database issue (Jan. 2007), pp. D521–6.

[56] Hisayuki Horai et al. "MassBank: a public repository for sharing mass spectral data for life sciences". en. In: *J. Mass Spectrom.* 45.7 (July 2010), pp. 703–714.

[57] Kai Dührkop et al. "Systematic classification of unknown metabolites using high-resolution fragmentation mass spectra". en. In: *Nat. Biotechnol.* 39.4 (Apr. 2021), pp. 462–471.

[58] Roman Bushuiev et al. "MassSpecGym: A benchmark for the discovery and identification of molecules". In: *Neural Inf Process Syst* (Oct. 2024).

[59] Xi-Wu Zhang et al. "Mass spectrometry-based metabolomics in health and medical science: a systematic review". en. In: *RSC Adv.* 10.6 (Jan. 2020), pp. 3092–3104.

[60] *mzCloud – Advanced Mass Spectral Database*. `https://www.mzcloud.org/`. Accessed: 2025-3-23.

[61] Carlos Guijas et al. “METLIN: A technology platform for identifying knowns and unknowns”. en. In: *Anal. Chem.* 90.5 (Mar. 2018), pp. 3156–3164.

[62] Hailong Zhang et al. “Congruent strategies for carbohydrate sequencing. 2. FragLib: an MSn spectral library”. en. In: *Anal. Chem.* 77.19 (Oct. 2005), pp. 6263–6270.

[63] Julio E Peironcely et al. “Automated pipeline for de novo metabolite identification using mass-spectrometry-based metabolomics”. en. In: *Anal. Chem.* 85.7 (Apr. 2013), pp. 3576–3583.

[64] Hiromi Ito et al. “In vitro and in vivo enzymatic syntheses and mass spectrometric database for N-glycans and o-glycans”. en. In: *Methods Enzymol.* 478 (2010). Ed. by Minoru Fukuda, pp. 127–149.

[65] Kim Kultima et al. “Development and evaluation of normalization methods for label-free relative quantification of endogenous peptides”. en. In: *Mol. Cell. Proteomics* 8.10 (Oct. 2009), pp. 2285–2295.

[66] Maria Sorokina et al. “COCONUT online: Collection of Open Natural Products database”. en. In: *J. Cheminform.* 13.1 (Jan. 2021), p. 2.

[67] Mark Davies et al. “ChEMBL web services: streamlining access to drug discovery data and utilities”. en. In: *Nucleic Acids Res.* 43.W1 (July 2015), W612–20.

[68] Barbara Zdrazil et al. “The ChEMBL Database in 2023: a drug discovery platform spanning multiple bioactivity data types and time periods”. en. In: *Nucleic Acids Res.* 52.D1 (Jan. 2024), pp. D1180–D1192.

[69] Sunghwan Kim et al. “PubChem 2023 update”. en. In: *Nucleic Acids Res.* 51.D1 (Jan. 2023), pp. D1373–D1380.

[70] John J Irwin et al. “ZINC20-A free ultralarge-scale chemical database for ligand discovery”. en. In: *J. Chem. Inf. Model.* 60.12 (Dec. 2020), pp. 6065–6073.

[71] Cornell Aeronautical. *The perceptron: A probabilistic model for information storage and organization in the brain*. `https://www.ling.upenn.edu/courses/cogs501/Rosenblatt1958.pdf`. Accessed: 2025-4-3.

[72] S J Martin et al. “Synaptic plasticity and memory: an evaluation of the hypothesis”. en. In: *Annu. Rev. Neurosci.* 23.1 (2000), pp. 649–711.

[73] Ben Li and Stephen Gilbert. "Artificial Intelligence awarded two Nobel Prizes for innovations that will shape the future of medicine". en. In: *NPJ Digit. Med.* 7.1 (Nov. 2024), p. 336.

[74] J J Hopfield. "Neural networks and physical systems with emergent collective computational abilities". en. In: *Proc. Natl. Acad. Sci. U. S. A.* 79.8 (Apr. 1982), pp. 2554–2558.

[75] David E Rumelhart et al. "Learning representations by back-propagating errors". en. In: *Nature* 323.6088 (Oct. 1986), pp. 533–536.

[76] Ian Goodfellow et al. *Deep Learning*. en. MIT Press, Nov. 2016.

[77] Ashish Vaswani et al. *Attention is all you need*. `https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`. Accessed: 2025-3-30.

[78] Jie Zhou et al. "Graph neural networks: A review of methods and applications". In: *arXiv* [*cs.LG*] (Dec. 2018).

[79] Federico Errica et al. "A fair comparison of graph neural networks for graph classification". In: *arXiv* [*cs.LG*] (Dec. 2019).

[80] Tom B Brown et al. "Language Models are Few-Shot Learners". In: *arXiv* [*cs.CL*] (May 2020).

[81] Gemini Team et al. "Gemini: A family of highly capable multimodal models". In: *arXiv* [*cs.CL*] (Dec. 2023).

[82] S Hochreiter and J Schmidhuber. "Long short-term memory". en. In: *Neural Comput.* 9.8 (Nov. 1997), pp. 1735–1780.

[83] Junyoung Chung et al. "Empirical evaluation of gated recurrent neural networks on sequence modeling". In: *arXiv* [*cs.NE*] (Dec. 2014).

[84] Alex Krizhevsky et al. "ImageNet classification with deep convolutional neural networks". en. In: *Commun. ACM* 60.6 (May 2017), pp. 84–90.

[85] Alexey Dosovitskiy et al. "An image is worth 16x16 words: Transformers for image recognition at scale". In: *arXiv* [*cs.CV*] (Oct. 2020).

[86] Zeming Lin et al. "Evolutionary-scale prediction of atomic-level protein structure with a language model". en. In: *Science* 379.6637 (Mar. 2023), pp. 1123–1130.

[87] Jacob Devlin et al. "BERT: Pre-training of deep bidirectional Transformers for language understanding". In: *arXiv* [*cs.CL*] (Oct. 2018).

[88] Alec Radford and Karthik Narasimhan. "Improving language understanding by generative pre-training". In: (2018).

[89] Marius Mosbach et al. "On the stability of fine-tuning BERT: Misconceptions, explanations, and strong baselines". In: *arXiv* [*cs.LG*] (June 2020).

[90] Hervé Abdi and Lynne J Williams. "Principal component analysis: Principal component analysis". en. In: *Wiley Interdiscip. Rev. Comput. Stat.* 2.4 (July 2010), pp. 433–459.

[91] Marie-Claire Hennion et al. "Retention behaviour of polar compounds using porous graphitic carbon with water-rich mobile phases". en. In: *J. Chromatogr. A* 712.2 (Oct. 1995), pp. 287–301.

[92] Allegra T Aron et al. "Reproducible molecular networking of untargeted mass spectrometry data using GNPS". en. In: *Nat. Protoc.* 15.6 (June 2020), pp. 1954–1991.

[93] Yuta Ogawa et al. "Current contributions of organofluorine compounds to the agrochemical industry". en. In: *iScience* 23.9 (Sept. 2020), p. 101467.

[94] Rafal Mulka et al. "FluoBase: a fluorinated agents database". en. In: *J. Cheminform.* 17.1 (Feb. 2025), p. 19.

[95] Stephen R Heller et al. "InChI, the IUPAC International Chemical Identifier". en. In: *J. Cheminform.* 7.1 (May 2015), p. 23.

[96] Matthew C Robinson et al. "Validating the validation: reanalyzing a large-scale comparison of deep learning and machine learning models for bioactivity prediction". en. In: *J. Comput. Aided Mol. Des.* 34.7 (July 2020), pp. 717–730.

[97] Samuel Goldman et al. "Prefix-tree decoding for predicting mass spectra from molecules". In: *arXiv* [*q-bio.QM*] (Mar. 2023).

[98] Roman Bushuiev. "Samořízené strojové učení pro interpretaci molekulárních dat z hmotnostní spektrometrie". PhD thesis. June 2023.

[99] G W Bemis and M A Murcko. "The properties of known drugs. 1. Molecular frameworks". en. In: *J. Med. Chem.* 39.15 (July 1996), pp. 2887–2893.

[100] Fleming Kretschmer et al. "Small molecule machine learning: All models are wrong, some may not even be useful". In: *Bioinformatics* biorxiv;2023.03.27.534311v2 (Mar. 2023).

[101] Emma L Schymanski et al. "Critical Assessment of Small Molecule Identification 2016: automated methods". en. In: *J. Cheminform.* 9.1 (Mar. 2017), p. 22.

[102] Samuel Goldman et al. "Annotating metabolite mass spectra with domain-inspired chemical formula transformers". In: *Bioinformatics* biorxiv;2022.12.30.522318v1 (Dec. 2022).

[103] Darko Butina. "Unsupervised data base clustering based on daylight's fingerprint and tanimoto similarity: A fast and automated way to cluster small and large data sets". en. In: *J. Chem. Inf. Comput. Sci.* 39.4 (July 1999), pp. 747–750.

[104] William Falcon et al. *PyTorchLightning/pytorch-lightning: 0.7.6 release*. 2020.

[105] Thomas Wolf et al. "HuggingFace's transformers: State-of-the-art natural language processing". In: *arXiv* [*cs.CL*] (Oct. 2019).

[106] Manfred Beckmann et al. "High-throughput, nontargeted metabolite fingerprinting using nominal mass flow injection electrospray mass spectrometry". en. In: *Nat. Protoc.* 3.3 (2008), pp. 486–504.

[107] Adriano Rutz et al. "The LOTUS initiative for open knowledge management in natural products research". en. In: *Elife* 11 (May 2022).

[108] David J Ashline et al. "Carbohydrate structural isomers analyzed by sequential mass spectrometry". en. In: *Anal. Chem.* 79.10 (May 2007), pp. 3830–3842.

[109] Robin Schmid et al. "Integrative analysis of multimodal mass spectrometry data in MZmine 3". en. In: *Nat. Biotechnol.* 41.4 (Apr. 2023), pp. 447–449.

[110] Louis-Félix Nothias et al. "Feature-based molecular networking in the GNPS analysis environment". en. In: *Nat. Methods* 17.9 (Sept. 2020), pp. 905–908.

[111] Niek F de Jonge et al. "Reproducible MS/MS library cleaning pipeline in matchms". en. In: *J. Cheminform.* 16.1 (July 2024), p. 88.

[112] Kermit K Murray. "Resolution and resolving power in Mass Spectrometry". en. In: *J. Am. Soc. Mass Spectrom.* 33.12 (Dec. 2022), pp. 2342–2347.

[113] Eric W Deutsch. "File formats commonly used in mass spectrometry proteomics". en. In: *Mol. Cell. Proteomics* 11.12 (Dec. 2012), pp. 1612–1621.

[114] Adam Paszke et al. “PyTorch: An imperative style, high-performance deep learning library”. In: *arXiv* [*cs.LG*] (Dec. 2019).

[115] Matthias Fey and Jan Eric Lenssen. “Fast graph representation learning with PyTorch Geometric”. In: *arXiv* [*cs.LG*] (Mar. 2019).

[116] Mike Folk et al. “An overview of the HDF5 technology suite and its applications”. In: *Proceedings of the EDBT/ICDT 2011 Workshop on Array Databases*. New York, NY, USA: ACM, Mar. 2011.

[117] David Weininger. “SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules”. en. In: *J. Chem. Inf. Comput. Sci.* 28.1 (Feb. 1988), pp. 31–36.

[118] Greg Landrum. *RDKit*. en. `https://www.rdkit.org`. Accessed: 2025-4-8.

[119] *CCMS ProteoSAFe Workflow Input Form*. `https://massive.ucsd.edu`. Accessed: 2025-4-8.

[120] Fleming Kretschmer et al. “Coverage bias in small molecule machine learning”. en. In: *Nat. Commun.* 16.1 (Jan. 2025), p. 554.

[121] Gabriel Asher et al. “LSM1-MS2: A foundation model for MS/MS, encompassing chemical property predictions, search and de novo generation”. In: *ChemRxiv* (June 2024).

[122] *PRISM: A foundation model for life's chemistry*. en. `https://enveda.com/prism-a-foundation-model-for-lifes-chemistry/`. Accessed: 2025-4-8.

[123] RunPod. *RunPod - The Cloud Built for AI*. en. `https://www.runpod.io/`. Accessed: 2025-4-8.

[124] David Rogers and Mathew Hahn. “Extended-connectivity fingerprints”. en. In: *J. Chem. Inf. Model.* 50.5 (May 2010), pp. 742–754.

[125] H L Morgan. “The generation of a unique machine description for chemical structures-A technique developed at chemical abstracts service”. en. In: *J. Chem. Doc.* 5.2 (May 1965), pp. 107–113.

[126] Beate I Escher et al. “Tracking complex mixtures of chemicals in our changing environment”. en. In: *Science* 367.6476 (Jan. 2020), pp. 388–392.

[127] Weihua Hu et al. “Open Graph Benchmark: Datasets for machine learning on graphs”. In: *arXiv* [*cs.LG*] (May 2020).

[128] Kai Dührkop and Sebastian Böcker. "Fragmentation trees reloaded". In: *arXiv* [*q-bio.QM*] (Dec. 2014).

[129] Shipei Xing et al. "BUDDY: molecular formula discovery via bottom-up MS/MS interrogation". en. In: *Nat. Methods* 20.6 (June 2023), pp. 881–890.

[130] Tomáš Pluskal et al. "Highly accurate chemical formula prediction tool utilizing high-resolution mass spectra, MS/MS fragmentation, heuristic rules, and isotope pattern matching". en. In: *Anal. Chem.* 84.10 (May 2012), pp. 4396–4403.

[131] Juan Carlos Alarcon-Barrera et al. "Recent advances in metabolomics analysis for early drug development". en. In: *Drug Discov. Today* 27.6 (June 2022), pp. 1763–1773.

[132] Seung Ha Lee and Dal Woong Choi. "Comparison between source-induced dissociation and collision-induced dissociation of ampicillin, chloramphenicol, ciprofloxacin, and oxytetracycline via mass spectrometry". en. In: *Toxicol. Res.* 29.2 (June 2013), pp. 107–114.

[133] Yannick Djoumbou Feunang et al. "ClassyFire: automated chemical classification with a comprehensive, computable taxonomy". en. In: *J. Cheminform.* 8.1 (Nov. 2016), p. 61.

[134] Kirill Degtyarenko et al. "ChEBI: a database and ontology for chemical entities of biological interest". en. In: *Nucleic Acids Res.* 36.Database issue (Jan. 2008), pp. D344–50.

[135] Eoin Fahy et al. "Update of the LIPID MAPS comprehensive classification system for lipids". en. In: *J. Lipid Res.* 50 Suppl.Supplement (Apr. 2009), S9–14.

[136] Petar Veličković et al. "Graph Attention Networks". In: *arXiv* [*stat.ML*] (Oct. 2017).

[137] Armen G Beck et al. "Recent developments in machine learning for mass spectrometry". en. In: *ACS Meas. Sci. Au* 4.3 (June 2024), pp. 233–246.

[138] Benyou Wang et al. "On Position Embeddings in BERT". In: (Oct. 2020).

[139] Anqi Mao et al. "Cross-entropy loss functions: Theoretical analysis and applications". In: *arXiv* [*cs.LG*] (Apr. 2023).

[140] Linus Ericsson et al. "Self-supervised representation learning: Introduction, advances and challenges". In: *arXiv* [*cs.LG*] (Oct. 2021).

[141] Markus Freitag and Yaser Al-Onaizan. "Beam search strategies for Neural Machine Translation". In: *arXiv* [*cs.CL*] (Feb. 2017).

[142] Pan Du et al. "Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching". en. In: *Bioinformatics* 22.17 (Sept. 2006), pp. 2059–2065.

[143] Kevin R Coombes et al. "Improved peak detection and quantification of mass spectrometry data acquired from surface-enhanced laser desorption and ionization by denoising spectra with the undecimated discrete wavelet transform". en. In: *Proteomics* 5.16 (Nov. 2005), pp. 4107–4117.

[144] Aditya Divyakant Shrivastava et al. "MassGenie: A transformer-based deep learning method for identifying small molecules from their mass spectra". en. In: *Biomolecules* 11.12 (Nov. 2021), p. 1793.

[145] T Konstantin Rusch et al. "A survey on oversmoothing in graph neural networks". In: *arXiv* [*cs.LG*] (Mar. 2023).

[146] Frank J Massey join(' '. "The kolmogorov-smirnov test for goodness of fit". In: *J. Am. Stat. Assoc.* 46.253 (Mar. 1951), p. 68.

[147] Nadim Nachar. "The Mann-Whitney U: A test for assessing whether two independent samples come from the same distribution". In: *Tutor. Quant. Methods Psychol.* 4.1 (Mar. 2008), pp. 13–20.

[148] Richard A Armstrong. "When to use the Bonferroni correction". en. In: *Ophthalmic Physiol. Opt.* 34.5 (Sept. 2014), pp. 502–508.

[149] Edward E Cureton. "Rank-biserial correlation". en. In: *Psychometrika* 21.3 (Sept. 1956), pp. 287–290.

[150] Hamparsum Bozdogan. "Model selection and Akaike's Information Criterion (AIC): The general theory and its analytical extensions". en. In: *Psychometrika* 52.3 (Sept. 1987), pp. 345–370.

[151] Patrick E McKight and Julius Najab. "Kruskal-Wallis Test". In: *The Corsini Encyclopedia of Psychology*. Hoboken, NJ, USA: John Wiley & Sons, Inc., Jan. 2010.

[152] Patrick Schober et al. "Correlation coefficients: Appropriate use and interpretation". en. In: *Anesth. Analg.* 126.5 (May 2018), pp. 1763–1768.

[153] Bruce Thompson. "Canonical correlation analysis". In: *Reading and understanding MORE multivariate statistics* (*pp*. Ed. by Laurence G Grimm. Vol. 437. American Psychological Association, xiii, 2000, pp. 285–316.

[154] Adriana Romero et al. “FitNets: Hints for thin deep nets”. In: *arXiv* [*cs.LG*] (Dec. 2014).

[155] Simon Kornblith et al. “Similarity of neural network representations revisited”. In: *arXiv* [*cs.LG*] (May 2019).

[156] Arthur Gretton et al. “Measuring statistical dependence with Hilbert-Schmidt norms”. In: *Lecture Notes in Computer Science*. Lecture notes in computer science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 63–77.

[157] YuanLi. *CKA-Centered-Kernel-Alignment: Reproduce CKA: Similarity of Neural Network Representations Revisited*. en.

[158] B Schh et al. “Comparing support vector machines with Gaussian kernels to radial basis function classi”. In: (1997).

[159] Maithra Raghu et al. “Do Vision Transformers see like convolutional neural networks?” In: *arXiv* [*cs.CV*] (Aug. 2021).

[160] Olga Russakovsky et al. “ImageNet large scale visual recognition challenge”. en. In: *Int. J. Comput. Vis.* 115.3 (Dec. 2015), pp. 211–252.

[161] O Roy and M Vetterli. “The effective rank: A measure of effective dimensionality”. In: *Proc. Eur. Signal Process. Conf. EUSIPCO* (Sept. 2007), pp. 606–610.