FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Assignment of master's thesis

Title:	Self-supervised machine learning for the interpretation of molecular mass spectrometry data
Student:	Bc. Roman Bushuiev
Supervisor:	Tomáš Pluskal, Ph.D.
Study program:	Informatics
Branch / specialization:	Knowledge Engineering
Department:	Department of Applied Mathematics
Validity:	until the end of summer semester 2023/2024

Instructions

Despite modern scientific advances, only a tiny fraction of molecules from nature have been discovered to date. Tandem mass spectrometry is a powerful technique enabling the identification of molecules present in biological and environmental samples (Kind et al., 2010). For each molecule, it measures a mass spectrum – a set of fragments of the molecule represented in terms of their masses and abundances. However, existing methods for elucidating the whole unknown molecule from its measured fragments are extremely limited, because they rely on narrow annotated libraries (Dührkop et al., 2019 and Stravs et al., 2022).

The objective of the thesis is to explore the paradigms of deep self-supervised machine learning and subsequent transfer learning for the interpretation of mass spectra. More precisely, the tasks are:

1. Collect a large dataset (millions of samples) of unannotated mass spectra suitable for self-supervised training.
2. Design a deep-learning model and a training objective allowing the model to learn effective representations (embeddings) of mass spectra without annotations.
3. Experimentally validate the capacity of the learned representations on downstream tasks aiming to discover new molecules.

References:

Dührkop, K., Fleischauer, M., Ludwig, M. et al. SIRIUS 4: a rapid tool for turning tandem mass spectra into metabolite structure information. Nat Methods 16, 299–302 (2019).

Electronically approved by Ing. Magda Friedjungová, Ph.D. on 26 January 2023 in Prague.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

https://doi.org/10.1038/s41592-019-0344-8
Kind, T., Fiehn, O. Advances in structure elucidation of small molecules using mass spectrometry. Bioanal Rev 2, 23–60 (2010). https://doi.org/10.1007/s12566-010-0015-9
Stravs, M.A., Dührkop, K., Böcker, S. et al. MSNovelist: de novo structure generation from mass spectra. Nat Methods 19, 865–870 (2022). https://doi.org/10.1038/s41592-022-01486-3

Electronically approved by Ing. Magda Friedjungová, Ph.D. on 26 January 2023 in Prague.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Master's thesis

Self-supervised machine learning for the interpretation of molecular mass spectrometry data

Roman Bushuiev

Department of Applied Mathematics

Supervisor: Tomáš Pluskal, Ph.D.

May 4, 2023

Acknowledgements

First and foremost, I would like to express my profound gratitude to my parents for providing me with the opportunity to pursue a university degree in the Czech Republic. Their support and generosity have laid the foundation for my career, and for that, I am eternally grateful. I would also like to extend my heartfelt appreciation to my supervisor, Dr. Tomáš Pluskal, for granting me the chance to work on the intriguing project presented in this thesis, for consistently delivering exceptional supervision, and for always supporting and encouraging my personal and professional development. Joining Dr. Pluskal's research group ignited my passion for research and has undoubtedly been a pivotal decision in my life.

I am deeply grateful to the members of the research groups I had the privilege of visiting while working on this thesis. My sincerest thanks go to Prof. Sebastian Böcker, Dr. Kai Dührkop, and all other members of the Böcker lab at the University of Jena for sharing their deep expertise in computational mass spectrometry and for the fruitful discussions centered around machine learning applications for small molecules. I also wish to express my genuine appreciation to Prof. Regina Barzilay, Serena Khoo, and Peter G Mikhael from the Massachusetts Institute of Technology for the productive brainstorming sessions and insightful conversations on the application of deep learning to mass spectrometry, as well as deep learning in the more general context of biochemistry.

Finally, I am grateful to Dr. Corinna Brungs and Dr. Robin Schmid for teaching me the intricacies of mass spectrometry.

This work was supported by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90140).

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as a school work under the provisions of Article 60 (1) of the Act.

Czech Technical University in Prague

Faculty of Information Technology

Citation of this thesis

Bushuiev, Roman. *Self-supervised machine learning for the interpretation of molecular mass spectrometry data*. Master's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2023.

Abstrakt

Objevování nových molekul je zásadní pro vědecký pokrok v biologických vědách a pro výzkum nových léčiv. Doposud však bylo popsáno méně než deset procent chemikální přítomných v lidském těle nebo v celé rostlinné říši. Hmotnostní spektrometrie je nejpopulárnější analytická technika pro detekci nových molekulárních struktur. Avšak kvůli složitosti experimentálních dat dokáží současné výpočetní metody interpretovat pouze nepatrnou část z dostupných hmotnostních spekter. V této práci představujeme nový přístup k dekódování hmotnosně-spektrometrických dat. Zatímco stávající nástroje se spoléhají na lidskou expertizu nebo na anotované referenční knihovny, naše metoda umožňuje extrakci molekulárních informací přímo z experimentálních měření na základě samořízeného učení. Konkrétně, vyvinuli jsme neuronovou síť založenou na Transformeru a zkompilovali nové datasety MSV^n obsahující 700 milionů neanotovaných hmotnostních spekter. Ukazujeme, že model trénovaný na MSV^n pomocí syntetických úloh, jako je například predikce maskovaných částí vstupních spekter, se sám naučil různé vlastnosti molekulárních struktur. Tyto neurální reprezentace hmotnostních spekter označujeme jako DREAMS (Deep Representations Empowering the Annotation of Mass Spectra) a ukazujeme, že se samostatně organizuzjí do bohatých molekulárních sítí a přitom odhalují nové druhy znalostí, nedosažitelné předchozími metodami. Získané výsledky potvrzují potenciál samořízeného učení posunout paradigma výpočetní hmotnostní spektrometrie, a pokládají tak solidní základ pro budoucí výzkum v této oblasti.

Klíčová slova hmotnostní spektrometrie, metabolomika, samořízené učení, Transformer

Abstract

Discovery of new molecules is crucial for scientific progress in life sciences and in drug discovery. Yet, currently, less than ten percent of chemicals have been uncovered within the human body as well as in the entire plant kingdom. Mass spectrometry is the most popular analytical technique for detecting novel molecular structures. However, due to the complexity of experimental data, current computational methods can interpret only a tiny fraction of available mass spectra. In this work, we introduce a novel approach for deciphering mass spectral data. While existing tools rely on human expertise or annotated reference libraries, our method enables extraction of molecular information directly from raw experimental measurements using self-supervised deep learning. Specifically, we developed a Transformer-based neural network and compiled new MSV^n datasets comprising 700 million unannotated mass spectra. We demonstrate that the model trained on MSV^n using artificial annotation-free objectives, such as predicting masked portions of input spectra, learns diverse properties of molecular structures. We term these neural representations of mass spectra as DreaMS (Deep Representations Empowering the Annotation of Mass Spectra) and show that they are unconditionally organized in rich molecular networks, revealing new knowledge unattainable by previous methods. The obtained results confirm the potential of self-supervised learning to shift the paradigm of computational mass spectrometry and, therefore, lay a solid groundwork for future research in this direction.

Keywords metabolomics, mass spectrometry, self-supervised learning, Transformer

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

The discovery and identification of small molecules and metabolites has a significant impact on many scientific fields, including drug development [1, 2], environmental analysis [3, 4], and disease diagnosis [5]. However, only a tiny fraction of small molecules has been discovered to date, with estimates ranging from 5% to 10% within the human body and the entire plant kingdom [6, 7]. The vast majority of the chemical space remains unexplored. Yet, the discovery of new molecular structures represents a major technological opportunity and challenge to expand our biochemical knowledge base for both uncharacterized and well-characterized organisms [8, 9].

Mass spectrometry (MS) based metabolomics [10], enabled by the Nobel Prize winning electrospray ionization (ESI) technique [11], has been established as a comprehensive approach for studying molecules and their biological processes in organisms. In essence, mass spectrometry ionizes and separates compounds based on their mass-to-charge ratio (m/z), providing information about the molecular structures present in a sample.

To enhance the depth and specificity of mass spectrometry analysis, tandem mass spectrometry (MS/MS or MS^2) [12] has been developed as a powerful extension of the technique. Tandem mass spectrometry involves the coupling of two or more mass analyzers in a single instrument, allowing for the selection and further fragmentation of specific molecular ions. For each selected ion, this process generates a mass spectrum describing the m/z ratios and abundances of individual fragments providing additional information on a molecular structure. The combinatorial nature of the fragmentation process enables more precise characterization

of unknown molecules and therefore significantly enhances their identification.

However, the complete annotation of chemical structures from mass spectra still remains a crucial bottleneck of metabolomics. In fact, a mere 2% of mass spectrometry data can be annotated with reference standards [13] and around 10% with a contemporary machine-learning toolbox [14]. The state-of-the-art SIRIUS platform [15] (comprising fragmentation trees [16], CSI:FingerID [17], CANOPUS [18], and other methods) has been developed for the last two decades and became a working horse for the interpretation of mass spectra. SIRIUS comprises tools of discrete optimization, combinatorics, and classic machine learning, which are laboriously regularized by human expertise in metabolomics. Despite the revolutionary success of deep learning in other domains of biochemistry [19, 20, 21], neural networks have not yet surpassed SIRIUS in mass spectrometry.

Our work is a first step towards the foundation model for metabolomics – a general pre-trained neural network capable of solving a wide range of tasks currently limiting mass-spectrometry-based scientific discovery. We observe that the current bottleneck of deep learning methods is a limited scope of spectral libraries (Figure 1.1, Figure 4.1, Figure 4.2). As a solution, we propose a self-supervised learning approach that can efficiently extract knowledge from millions of raw, unlabeled data points. This is achieved by training the neural network on artificial tasks, such as predicting hidden parts of input spectra. We experimentally show that such an artificial pre-training leads to the emergence of structural features of small molecules derived purely from millions of unannotated mass spectra. We demonstrate how our method can be employed for molecular networking and how it can be effectively fine-tuned to learn from small annotated datasets.

1.1 Contributions

Our work presents the following key advancements toward the development of a large-scale foundation model for metabolomics:

- We conduct a systematic review of related work, to the best of our knowledge, encompassing all deep learning methods for tandem mass spectrometry.
- We examine the annotated NIST20 and MoNA spectral libraries and determine that their size is insufficient for strictly supervised learning. To max-

imize their utility in a semi-supervised setting, we develop a novel strategy for train/validation splitting called Murcko histograms. Our approach is specifically designed to account for the fragmentation nature of tandem mass spectrometry and surpasses existing approaches in both qualitative and quantitative aspects.

- We extract over 700 million tandem mass spectra from diverse metabolomics studies published in the MassIVE repository and establish nine high-quality subsets, which we term MSV^n datasets. To achieve this, we design a distributed pipeline of algorithms for processing metabolomics data files, assessing the quality of the underlying data, and effectively clustering mass spectra.
- We develop a new Transformer-based neural network for mass spectrometry. Specifically, we improve the inductive bias of the Transformer toward mass spectrometry by utilizing Fourier features, allowing the model to effectively operate on high-accuracy measurements of molecular masses.
- We examine the pre-training of the proposed neural network toward several self-supervised objectives on MSV^n datasets. Our newly developed approach for evaluating the effectiveness of pre-training enables us to select the optimal model among 100 experiments with different hyperparameters and training setups. We term the intermediate representations extracted from the model as DreaMS (Deep Representations Empowering the Annotation of Mass Spectra)[1]. Our results demonstrate that these representations gain knowledge about molecular structures through the course of self-supervision.

1.2 Thesis structure and notation

We begin the thesis with the Background chapter, which defines the studied problem and introduces essential concepts. Next, we provide a systematic overview of Related work, with a particular emphasis on existing deep learning models. Subsequently, we describe our analysis of the annotated spectral libraries and the collection of unannotated MSV^n datasets in chapter Training data collection and analysis. The following chapter Methods and experimental setup describe the ar-

[1] We choose DreaMS as the naming consonant with SMILES (Simplified molecular-input line-entry system) to highlight the correspondence between the deep representations of spectra and molecular structures. Similarly to BERT (Bidirectional Encoder Representations from Transformers), we use DreaMS to refer to both the representations and the neural network architecture.

chitecture of the DREAMS neural network, self-supervised pre-training objectives, and methods for evaluating self-supervision. Following that, we present our results and key findings regarding the pre-training in chapter Results. Finally, we outline the Conclusions & Future work.

Throughout the thesis, we use the standard mathematical notation [22]. To avoid ambiguity, we reserve $\mathbf{m}, \mathbf{i} \in \mathbb{R}^n$ to represent the m/z and intensity values of a mass spectrum containing n peaks. We may also use $\mathbf{i}$ to denote binned mass spectra, where n represents the number of m/z bins, $\mathbf{i}$ denotes the sum of intensities in each bin, and $\mathbf{m}$ denotes the average m/z in each bin. In line with deep learning literature, we use the $\cdot \| \cdot$ operator to denote the concatenation of two tensors along the first dimension. We use $\arg\max_x y$ either to retrieve the indices of $x \in \mathbb{N}$ largest elements from $y \in \mathbb{R}^k$ or to retrieve the argument $x \in X$ minimizing some function $f(x) : X \to \mathbb{R}$.

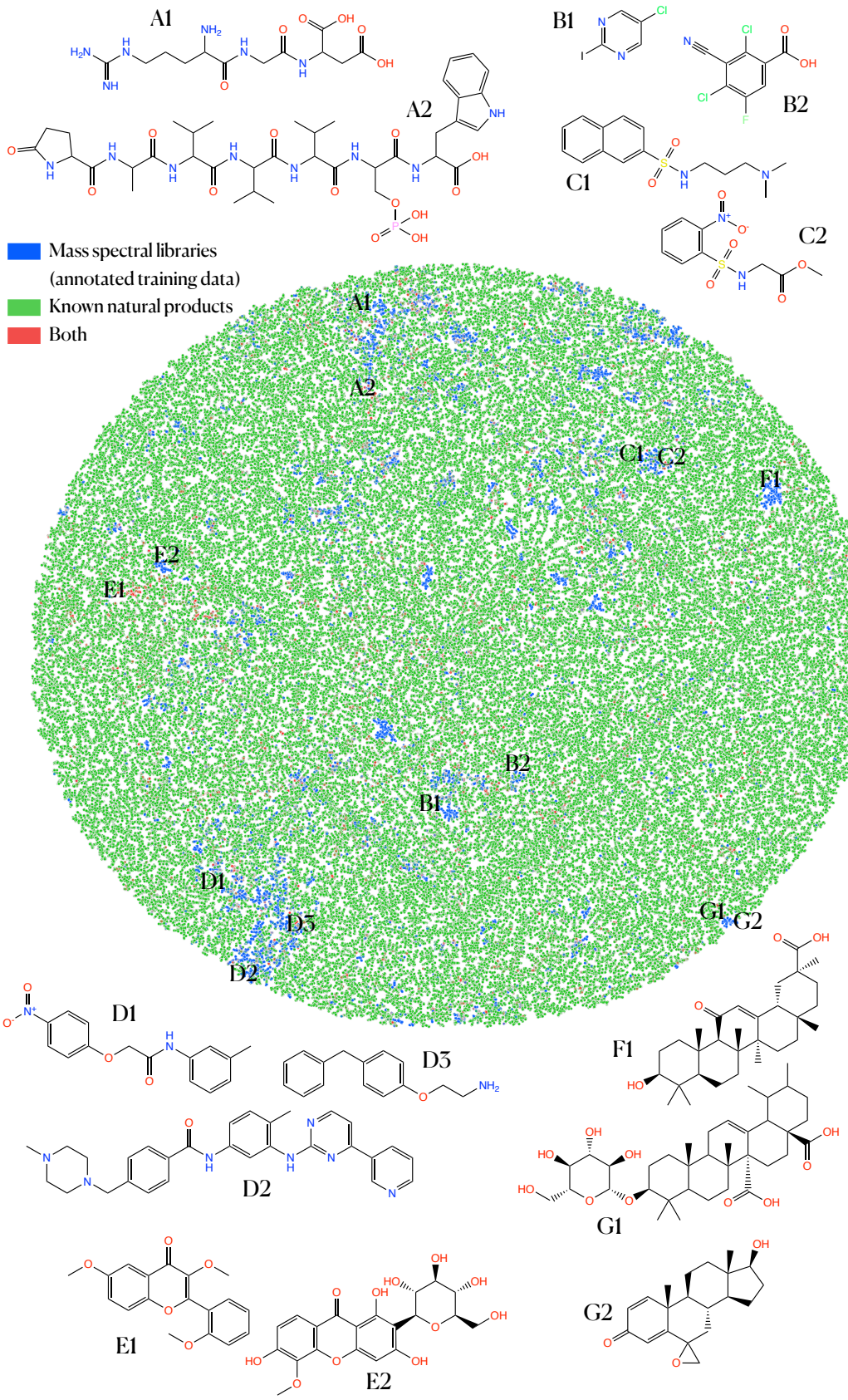


Figure 1.1: The boundaries of supervised methods for mass spectrometry. The figure shows the small molecules of annotated spectral libraries projected onto known natural products. Spectral libraries sparsely populate specific regions with chemical derivatives, not rendering the biological diversity of small molecules and thus severely limiting supervised methods. The projection is a TMAP [23] computed on the molecules of NIST20 [24], MoNA [25] (spectral), COCONUT [26] (natural products) within the mass range of [200, 1000] Da, and standardized with ChEMBL Structure Pipeline [27].

CHAPTER 2

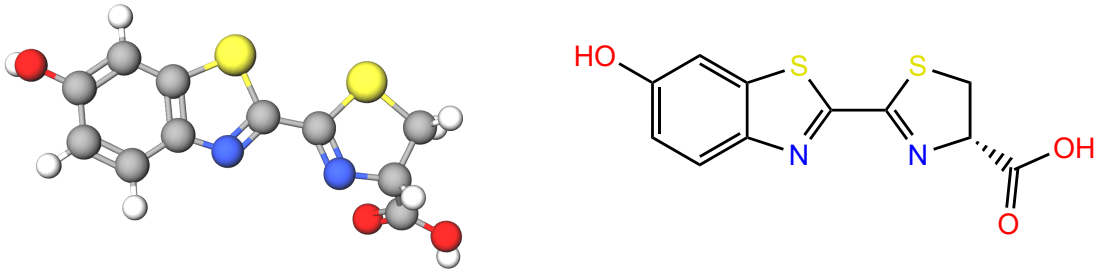
Background

In this chapter, we present an overview of the fundamental concepts that underpin this thesis. Our objective is not to delve into an exhaustive discussion of these fields but rather to offer the essential background required for the subsequent analysis. For a more in-depth exploration of the subjects, readers are encouraged to consult specialized literature [28, 29, 30]. Also, it should be understood that within the domain of biochemistry, there is hardly ever a rule without an exception.

2.1 Biochemistry

2.1.1 Small molecules

An atom is the fundamental building block of matter. Each atom is formed of **protons** and **neutrons** comprising a nucleus, as well as **electrons** orbiting around the nucleus. The number of protons defines a **chemical element** of an atom. For example, H (hydrogen) atom has 1 proton, C (carbon) atom has 6 protons, Br (bromine) atom has 35 protons, and so on. A **molecule** is a compound made up of two or more atoms that are chemically **bonded** together by sharing electrons. The more pairs of electrons they share the stronger the bond between atoms. **Molecular structure** is typically understood as a labeled undirected graph, where nodes represent atoms and are labeled by the corresponding chemical elements along with their spatial coordinates. Edges represent bonds and are labeled by the number of shared electron pairs (Figure 2.1).



3D Ball-and-stick representation

2D Skeletal structure

Figure 2.1: Molecular structure of *firefly luciferin* – a compound responsible for the characteristic yellow light emission from many firefly species.

Molecular representations

Molecules containing carbon-hydrogen bonds are named **organic compounds**. Since they constitute the majority of known chemicals, it is convenient to represent them using simplified planar graph representations - **skeletal structures**. In a skeletal structure, nodes are implicitly associated with carbon atoms, and hydrogens adjacent to carbons are omitted. Noteworthy, this adaptation does not affect the expressivity of the representation because hydrogens can be unambiguously filled based on the bonding capacity of each element given by its composition. A spatial arrangement of atoms is encoded in special **stereochemical** types of bonds visualized as either dashed or solid triangles representing two opposite directions orthogonal to a molecular plane. Molecules differing only in stereochemical bonds are referred to as **stereoisomers**.

In order to simplify computer storage and processing, molecular structures are commonly encoded as strings. The three most widely used variants are **SMILES** (Simplified molecular-input line-entry system), **InChI** (International Chemical Identifier), and **InChIKey**. Although both SMILES and InChI serve the purpose of uniquely identifying a molecule as a sequence of characters, SMILES are more human-readable and simpler but do not undergo a unified standard. In contrast, InChI strings have more complex yet standardized grammar. To give an example, SMILES string of *firefly luciferin*, depicted in Figure 2.1, is `C1[C@@H](N=C(S1)C2=NC3=C(S2)C=C(C=C3)O)C(=O)O`, while it's InChI string is `InChI=1S/C11H8N2O3S2/c14-5-1-2-6-8(3-5)18-10(12-6)9-13-7(4-17-9)11(15)16/h1-3,7,14H,4H2,(H,15,16)/t7-/m1/s1`. InChIKey representations are fixed-size hashes (e.g. `IWJYW`

BVPCGUPLO-BFUDMSGGSA-N) derived from InChI strings, which are convenient, for instance, to perform searches of molecules in large databases. Another compact coarse-grained representation of a molecule is its **chemical formula**, which represents the histogram of chemical elements within a molecule. Accordingly, the chemical formula of *firefly luciferin* is $C_{11}H_8N_2O_3S_2$.

The comparison of molecular structures is commonly conducted by utilizing their **fingerprints**, which are fixed-size binary vectors. A basic example of a molecular fingerprint is the Molecular ACCess System (MACCS) [31], where each of its 166 bits represents the presence or absence of a specific predefined substructure. The most widely-adopted family of fingerprints is the extended-connectivity fingerprints (ECFP) [32], which are fixed-size hashes encoding the local neighborhoods of molecular atoms. To compare two molecules, the most common approach is to generate the corresponding fingerprints and compute their Tanimoto similarity. The Tanimoto similarity is defined as a ratio of the number of shared positive bits to the total number of unique positive bits present in both fingerprints.

Molecular properties

As "mass spectrometry" in the title of the thesis suggests, **molecular mass** is a central notion for this work. It is characterized as a sum of **atomic masses** constituting the molecule and is usually measured in **Da** (Daltons). Single Dalton is defined as $\frac{1}{12}$ of the mass of ^{12}C (carbon atom containing 6 protons and 6 neutrons). Importantly, a molecule is roughly termed as a "small molecule" if its mass is less than 1000 Da. As a consequence of the definition of a Dalton and the significantly smaller mass of electrons compared to protons and neutrons, one nuclear particle has a mass approximately equal to 1 Da. Specifically, a proton has a mass of approximately 1.007 Da, a neutron has a mass of approximately 1.009 Da, and an electron has a mass of approximately 0.0005 Da[2]. While the number of protons in the atom of a chemical element is given by the definition, the notation ^{12}C is used to explicitly express the number of neutrons and protons.

Atoms having the same number of protons but different numbers of neutrons are referred to as **isotopes** of a chemical element. For instance, Cl (chlorine) el-

[2]Although the mass of an atom might be expected to equal the sum of the masses of its particles, it is always slightly less (with the exception of the hydrogen atom). This phenomenon is caused by the nuclear binding energy and is termed the mass defect of the nucleus.

ement has two stable[3] isotopes: ^{35}Cl having a mass of 34.96885269(4) Da and ^{37}Cl having a mass of 36.96590258(6) Da. Naturally, ^{35}Cl occurs in roughly 76% of cases and ^{37}Cl occurs in remaining 24% of cases. Another element S (sulfur) has four stable isotopes: ^{32}S, ^{33}S, ^{34}S, and ^{36}S with natural abundances 94.99%, 0.75%, 4.25%, and 0.01% respectively. In contrast, F has only a single stable isotope ^{19}F. The mass of the most abundant isotope is often termed as a **monoisotopic mass** of an element.

Another molecular property essential for the domain of mass spectrometry is a **molecular charge**. A molecule is defined as **negatively charged**, **neutral**, or **positively charged** if its atoms in total have more, equal, or fewer electrons than protons respectively. A charged molecule is termed **ion** and its charge is often expressed as an integer indicating the difference between the number of protons and electrons. The sign of such an integer can be often found in different notations and depictions of a molecule. For example, a positively-charged ion of a molecule M can be denoted as M^+.

2.1.2 Liquid chromatography tandem mass spectrometry

The identification of molecules present in a sample is a fundamental task in various fields of biology and environmental science. To achieve this, Liquid Chromatography Tandem Mass Spectrometry (LC-MS/MS) is the most widely employed technique. It constitutes an intricate pipeline formed of liquid chromatography and several stages of mass spectrometry (i.e. tandem mass spectrometry) to separate, elucidate, and quantify compounds in complex mixtures. Nevertheless, the interpretation of the output data from LC-MS/MS presents a significant challenge as information about the molecules is only available in terms of their masses or the masses of their fragments. To better understand the challenge, it is necessary to briefly discuss the individual components of LC-MS/MS and define several important terms.

Acquisition of mass spectra

Mass spectrometry (MS) plays a critical role in the workflow of LC-MS/MS, enabling the determination of the molecular mass of a compound. The fundamental principle of MS involves ionizing a sample to create charged molecules or frag-

[3] Isotope is stable if it does not decay into other elements on geologic timescales.

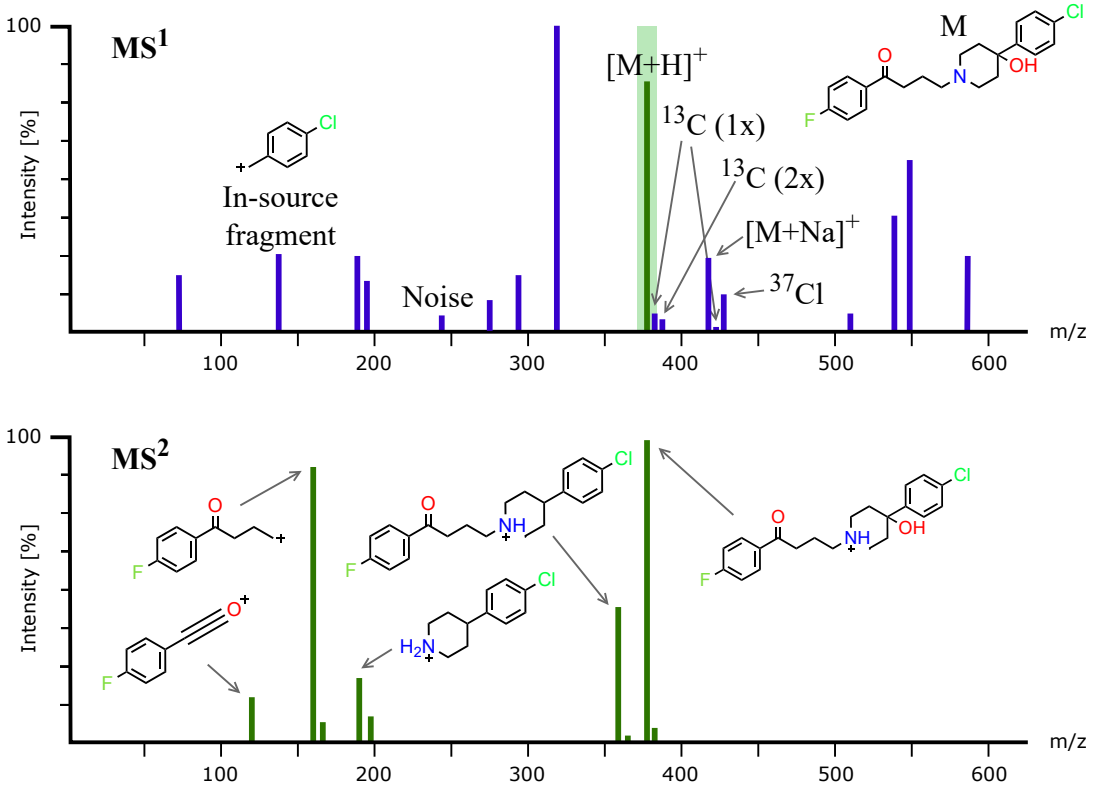


Figure 2.2: Examples of MS^1 and MS^2 spectra. (Top) An example of an MS^1 spectrum, featuring haloperidol (designated as "M"), with a mass of 375.14 Da. Several peaks are labeled, while others may correspond to different compounds. Isotopes are denoted by ^{13}C (1x) and ^{13}C (2x), and ^{37}Cl, indicating ions ($[M+H]^+$ or $[M+Na]^+$) containing corresponding isotopes. Notice, that the spectrum is simplified for visualization purposes, and one can frequently observe more intricate isotopic patterns and adduct species. **(Bottom)** MS^2 spectrum acquired by fragmenting protonated haloperidol from MS^1 belonging to the isolation window highlighted in green.

ments that are subsequently separated based on their mass-to-charge ratio (**m/z**) and detected using a mass analyzer. It is important to note that MS instruments are only capable of measuring m/z values and not the mass or charge[4] of the ions individually. The resulting **mass spectrum** is a collection of two-dimensional points, with the first dimension representing m/z values and the second dimension corresponding to their respective abundances (i.e. the **intensities** of the detected signals).

The ionization process in MS can occur through a variety of methods, including electron impact ionization (EI), electrospray ionization (ESI), and matrix-assisted laser desorption ionization (MALDI). Regardless of the method used, the result is

[4]Although, advanced mass spectrometry instruments report charges.

the formation of ions with different m/z ratios, which are then separated by the mass analyzer. ESI is the most convenient and common method to couple with LC. During the electrospray ionization, the liquid sample is sprayed through a small needle that has a high voltage applied to it. The high voltage causes the liquid to form tiny droplets, and as these droplets move through the air, they pick up an electrical charge. These droplets will either be positively or negatively charged depending on the polarity of the applied voltage. Further, the charged droplets continue to break apart and re-form until they eventually become individual ions. During the ionization process, ions form **adducts**, which are clusters of molecules that stick together due to electrostatic interactions. For example, a molecule in the sample may pick up a positively charged droplet ion such as a proton, Na (sodium), or K (potassium). As a consequence, the resulting ion measured in a mass spectrum has a higher mass-to-charge ratio than the original molecule. Such adduct species of molecule M are then denoted as $[M+H]^+$, $[M+Na]^+$, or $[M+K]^+$ respectively.

It is important to note that the understanding of the sample introduced to MS system should not be limited to a set of mutually exclusive molecules. Instead, it should be regarded as a complex mixture of compounds with millions of duplicates for each molecular structure. For the sake of simplicity, we often refer to a "molecule" as a group of identical compounds in the state prior to the ionization. In fact, within a single mass spectrometry experiment one can frequently observe m/z values that correspond to various adduct species of "the same molecule", as well as differing isotopic compositions of "the same molecule" (Figure 2.2).

After ionization, the mass analyzer separates the ions based on their m/z ratios. There are various types of mass analyzers, such as time-of-flight (TOF), quadrupole, and Orbitrap, each with unique strengths and weaknesses. However, they all operate on the principle of using a combination of electromagnetic fields to isolate ions. Finally, the ion detector, typically integrated into the mass analyzer, measures the m/z value and intensity of the signal. While we won't delve into the technicalities, it's worth noting certain peculiarities of the measurement that are significant for subsequent MS data analysis.

Properties of mass spectra and MS instruments

To begin with, the detector records the entire ion signal as a function of time, resulting in a mass spectrum that exhibits a continuous signal proportional to the intensity of ions relative to their mass-to-charge (m/z) ratio. This type of spectrum is known as a **profile** mass spectrum. In contrast, the instrument often produces **centroid** spectra, which undergo pre-processing via an algorithm that extracts peak information from the profile mass spectrum. The resulting signals are reported at specific m/z values and are termed **peaks**. Also, it is a common practice to pre-process intensities such that they are represented as fractions of the maximum spectrum intensity (corresponding to the **base peak**), and are referred to as **relative intensities**.

The accuracy of the measured m/z ratios is profoundly reliant on the instrument's quality and its constituents. The two most crucial indicators of quality are the **resolution** and **accuracy** of the instrument. Resolution denotes the ability to differentiate ions with nearly identical masses, whereas accuracy indicates how close the measured m/z value is to the actual ground-truth value. Since modern instruments generally possess high separation capabilities, the resolution is often not a problem for downstream data analysis. Nonetheless, accuracy remains a pivotal concept.

Vendors typically provide the accuracy of individual instruments in **ppm** (parts per million), which means that accuracy is inversely proportional to the measured mass. Specifically, a ppm of 5 would indicate that for the ground-truth m/z m, the measured value would fall within the interval $m \pm 5 * 10^{-6}m$.

Another critical property of an instrument is its **sensitivity**. Low-sensitivity measurements may fail to detect all anticipated molecules. Conversely, high-sensitivity measurements may lead to a significant amount of **noise** - peaks that do not correspond to any actual molecules.

Annotation of mass spectra

To annotate a mass spectrum means to annotate its individual peaks with molecular structures. Despite the seemingly straightforward nature of this task, it is fraught with several significant challenges.

First of all, spectral peaks are not necessarily associated with the anticipated molecules. Some may represent **contaminants** present in the MS system, others may be molecular fragments (known as **in-source fragments**). Furthermore, the prevailing majority of low-intensity peaks in a typical spectrum are simply instrument noise.

Even the ability to distinguish noise from real anticipated signals cannot resolve a central challenge. A single peak of a compound does not provide a unique characterization of its structure. Indeed, even assuming a high resolution of an instrument, there may exist millions of combinatorially generated elemental compositions satisfying the same m/z. The information contained within an MS1 spectrum can offer some insights into the possible structural features of the underlying molecule. For instance, assuming a rich isotopic distribution, one could guess a template for the chemical formula. Alternatively, types of adduct species may characterize ionization sites of a molecule suggesting the presence of certain functional groups. However, the information obtained from MS1 alone does not contain any information about atom connectivity and is generally insufficient for a comprehensive characterization of the desired molecular structure.

Tandem mass spectrometry

Tandem mass spectrometry (MS/MS or MS^2) is a powerful technique enabling the enrichment of mass spectrometry data with structural information. It involves using two or more subsequent mass spectrometry experiments. After the first stage of mass spectrometry (**MS^1**) as described above, the instrument selects specific ions of interest termed **precursor ions**. It is realized by defining an **isolation window** of specific width sliding across the m/z range and selecting ions of interest. The selected m/z values can be either arbitrary (data-independent acquisition; DIA) or pre-defined beforehand (data-depended acquisition; DDA).

Once the precursor ion is identified, it is then subjected to a second stage of mass spectrometry (**MS^2**), where the machine brakes it into fragments. The most prominent technique to fragment the ion is collision-induced dissociation (CID). It works by accelerating the molecule towards a gas, causing it to collide with the gas molecules and recursively break into smaller substructures. The more **collision energy** is provided, the more molecular fragments are obtained as a result. The second mass spectrometer is then used to measure the **MS^2** (**MS/MS, frag-**

mentation) spectrum, where peaks represent m/z ratios of individual fragments. Additionally, the process of selecting and fragmenting ions can be recursively repeated up to the MS^n level for any reasonable positive integer n retaining fragments.

Because the instrument can only detect ionized fragments, only a portion of substructures are recorded. For instance, a singly-charged ion broken into two fragments will form one ion and one neutral molecule (**neutral loss**) depending on the current location of the charge within the molecule. However, since "precursor ion" is in fact a group of identical molecules (as described in Acquisition of mass spectra) and the charge site is not deterministic with respect to molecule, $\mathbf{MS^2}$ spectrum often contains peaks for both parts with intensities reflecting their probability distribution. In particular, fragmentation spectrum often contains **precursor peak** corresponding to the whole non-fragmented precursor ion.

It is important to note that the graph-theoretical abstraction of fragmentation as a consequent removal of bonds is often, but not always, correct. For example, during CID fragmentation, a molecule can undergo rearrangement or transfer reactions leading to graph deformations, such as the formation of new rings [33, 34].

Annotation of tandem mass spectra

The distribution of masses provided in an MS^n spectrum offers substantially more structural information about the precursor molecule than a sole MS^1 spectrum. Given an MS^n spectrum the aim is to "arrange" the masses into a complete molecular structure. The extent to which MS^n information is sufficient for deducing the complete structure remains a fundamental open question. However, regardless of the completeness of the structural information, a complex high-accuracy fragmentation spectrum usually uniquely describes the molecule. The opposite statement is true only under the assumption of a similar experimental setup. The same compound can be fragmented in completely different ways depending on circumstances such as adduct species or applied collision energy. Rationally, this observation motivates the repeated measurement of the same compound with different instrument parameters enabling the enrichment of the structural information.

The goal of the annotation of an MS^n spectrum typically only consists of elucidating its underlying precursor molecule. However, in practice, this procedure

requires the annotation of individual peaks. In addition to the aforementioned annotation challenges such as dealing with noise and contamination, MS^n spectra may also encounter problems caused by the precursor selection setup.

Firstly, the width of the isolation window significantly affects the information present in a fragmentation spectrum. A wide window may isolate multiple molecules of similar masses and fragment them, resulting in a single **chimeric spectrum**, which is be misleading for further annotation. On the other hand, a narrow window may miss desired isotopes of the same molecule. Ultimately, the isolation window may be triggered for a wrong m/z range, resulting in a spectrum containing nothing but noise.

Secondly, collision energy affects the number of fragments, their size, and their structure, which in turn affects the number of peaks and their positions. Frequently, as a result of low collision energy, an MS^n spectrum may contain only a single meaningful peak representing an unfragmented molecule. In contrast, high energy may result in too severe fragmentation, limiting structural annotation.

Finally, CID fragmentation has limitations with regard to the absolute interpretation of a molecular structure. For example, it is nearly impossible to distinguish stereoisomers with sole MS^n data. However, orthogonal sources of information, such as liquid chromatography, may separate the isomers before introducing them to the mass spectrometry stage [35, 36].

Liquid chromatography

Liquid chromatography is frequently used as a sample preparation step before mass spectrometry analysis. Its purpose is to separate and purify the compounds of interest. The technique works by carrying the mixture with a liquid **mobile phase** through a **column** packed with a **stationary phase** (immobile material with specific chemical composition). Depending on the physical and chemical properties, each compound interacts with the column in different ways. As a result, the molecules are separated as they travel through the column and emerge at different timestamps, which are termed **retention times** (**RT**). Intensities of subsequent mass spectrometry signals as a function of RT is referred to as **chromatogram**. LC-MS/MS experiment is then represented as a three-dimensional collection of data points – mass spectra ordered by retention time.

2.2 Machine learning

2.2.1 Deep learning

Solving a problem with deep learning begins by formulating the problem as a continuous function $f : \mathbb{R}^n \to \mathbb{R}^m$. For example, to predict the chemical class of a molecule from its mass spectrum, the spectrum must be represented as a continuous vector $\mathbf{x} \in \mathbb{R}^n$ or as a collection of such vectors. Likewise, each chemical class of interest should be encoded as a vector $\mathbf{y} \in \mathbb{R}^m$ or in terms of multiple such vectors. Rather than attempting to manually construct f to map any input $\mathbf{x}$ to the desired output $\mathbf{y}$, it is assumed that f possesses degrees of freedom $\boldsymbol{\theta}$ (i.e. **parameters**). More specifically, f is assumed to belong to a class of parametrized functions $f \in \{f_{\boldsymbol{\theta} \in \Theta}\}$, where Θ represents the space of all possible values for each parameter. By adjusting the parameters $\boldsymbol{\theta}$, f can take various forms. The objective of **learning** is to identify the optimal form of f solving a problem on a given **dataset** of n reference input/output examples $D = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \ldots, (\mathbf{x}_n, \mathbf{y}_n)\}$, where $\mathbf{y}$'s are commonly referred to as **labels** or **annotations**. This goal is accomplished by solving the following continuous optimization problem:

$$\arg\min_{f \in \mathcal{F}} \sum_{(\mathbf{x},\mathbf{y}) \in D} \mathcal{L}(f(\mathbf{x}), \mathbf{y}),$$

where the function $\mathcal{L}$ (**loss function**) produces smaller values when the output of f for input $\mathbf{x}$ closely resembles the desired output $\mathbf{y}$, and larger values otherwise. The optimization problem is typically addressed using variations of the gradient descent algorithm. The iterative process of searching for the optimum is referred to as **training**.

The most prominent variation of gradient descent is stochastic gradient descent. This approach involves performing optimization iterations on random disjoint **batches** of training examples within the dataset rather than on the entire D. This modification typically results in more robust training, leading to better local optima. Furthermore, training is usually conducted over multiple traversals (i.e. **epochs**) of all dataset batches.

The core principle of deep learning is to employ an expressive functional class $\mathcal{F}$ and impose minimal assumptions on the problem, allowing the optimization process to discover the most performant model with respect to the loss function $\mathcal{L}$ over the dataset D. The most canonical example of such class is a **feed-forward**

neural network (FFN; $f : \mathbb{R}^n \to \mathbb{R}^m$), which is defined as follows:

$$\begin{aligned} f &= f_l \circ f_{l-1} \circ \cdots \circ f_1, \\ f_i(\mathbf{x}) &= \sigma(\mathbf{W}_i \mathbf{x}), \\ \sigma(\mathbf{x})_i &= \max\{0, \mathbf{x}_i\}, \end{aligned}$$

Here, matrices $\mathbf{W}_i$ represent the parameters that are estimated during the training. In other words, an FFN consists of **layers** f_i, alternating linear functions $\mathbf{W}_i$ with non-linear maps σ (i.e., **activation functions**). Here, non-linearity σ is defined as the most-popular ReLU activation. A fundamental theoretical result of deep learning is the Universal Approximation Theorem, which roughly states that such a composition of layers "is expressive enough to approximate any continuous function" [37]. However, to solve a practical problem one typically needs to strengthen the assumptions on the functional class (i.e. impose **inductive biases**) and to adjust the described approach for the definition of a loss function and a dataset.

Typically, the process of modeling an input data modality (e.g., molecular graph or image) as a continuous vector is not entirely unambiguous. Consequently, a variety of neural networks exist, such as graph neural networks (GNNs), convolutional neural networks (CNNs), and numerous others, each adept at capturing the specific characteristics of the modality. The prominent class of neural networks is Transformers. Although they were originally proposed for long sequences, such as text [38], they have gained considerable attention across all major data modality domains. The success of Transformers can be attributed to their effective scalability to large datasets and the "generic" inductive bias that simply focuses on identifying relationships between all pairs of input objects.

Also, it may be advantageous not to exclusively regard the training examples $(\mathbf{x}_i, \mathbf{y}_i)$ as ultimate inputs and outputs. For instance, generative diffusion models, employed for generating novel objects, express both $\mathbf{x}_i$ and $\mathbf{y}_i$ in terms of $\mathbf{y}_i$. Training batches are formed by sampling two random frames from the trajectories of the Markov process noising the input data points. Subsequently, a neural network is trained to denoise the data frames. As a result, when provided with random noise as input, the network is capable of generating a novel object that resembles those present in the dataset.

2.2.2 Self-supervised learning

In many cases, there may not be enough labeled data to train a neural network to directly output the desired **y** from the given **x**. This situation is particularly evident in mass spectrometry, where billions of mass spectra exist, but only a tiny fraction has corresponding annotations. The paradigm of **self-supervised learning** (SSL) helps to overcome the limitations of such strictly **supervised** settings.

The primary concept of SSL is to train the neural network on a dataset artificially constructed solely from unannotated **x**'s. Label-free tasks are typically created by corrupting inputs and training the model to correct them. The underlying idea is that the neural network internally learns the "semantics" of the data by solving such "syntax-based" tasks. A canonical example of successful self-supervision was demonstrated in natural language processing (NLP) by Devlin et al. [39]. They **pre-trained** a Transformer-based neural network, BERT, to predict the masked parts of input sentences and showed that further supervised **fine-tuning** of the model with an additional layer achieves state-of-the-art performance across a wide range of NLP tasks.

Subsequent analysis of the pre-trained BERT and its **embeddings** (i.e. intermediate outputs of some layer f_i on a given input **x**) reveals that the neural network gains a deep understanding of natural language exclusively through self-supervision [40, 41].

Inspired by the success of self-supervised learning in NLP, its applicability in other domains is continually being investigated. However, the effectiveness of SSL varies across different domains. For example, it has been successfully applied to protein sequences [42] and computer vision [43, 44], but has not yet achieved remarkable results for small molecules [45]. Our work explores whether self-supervised learning can be effective for mass spectrometry.

CHAPTER 3

Related work

Although the annotation of tandem mass spectra is crucial for accurately identifying small molecules, it continues to be a central challenge in computational metabolomics. Research efforts directed towards addressing the problem can be classified into three primary categories: forward annotations, inverse annotations, and spectral similarity measures.

Forward annotations aim to predict the fragmentation spectrum of a given molecule, whereas Inverse annotations focus on inferring the molecular structure or its properties based on the observed spectrum. Spectral similarity enables the comparison of experimental spectra with reference standards or forward predictions, thereby facilitating the clustering of similar spectra to propagate the inverse annotations (i.e. Molecular networking). Our method aims to advance each of the research directions by deriving general representations of mass spectra. Therefore, we separately discuss it in the context of considerably different yet related works (section Unsupervised representation learning).

The chapter has two goals: (i) to provide an exhaustive overview and systematization of state-of-the-art methods operating on tandem mass spectra, with a particular emphasis on deep learning, and (ii) to frame our work in the context of existing methods. Note that, we do not aim to cover the literature on machine learning applied to other MS data modalities, such as predictions from imaging mass spectrometry data [46, 47], tabular MS datasets [48], or retention time prediction [49, 50]. Similarly, since machine learning approaches for narrower subdomains of metabolomics (e.g. proteomics [51] and lipidomics [52]) can rely on molecule-specific priors, we only mention several relevant methods

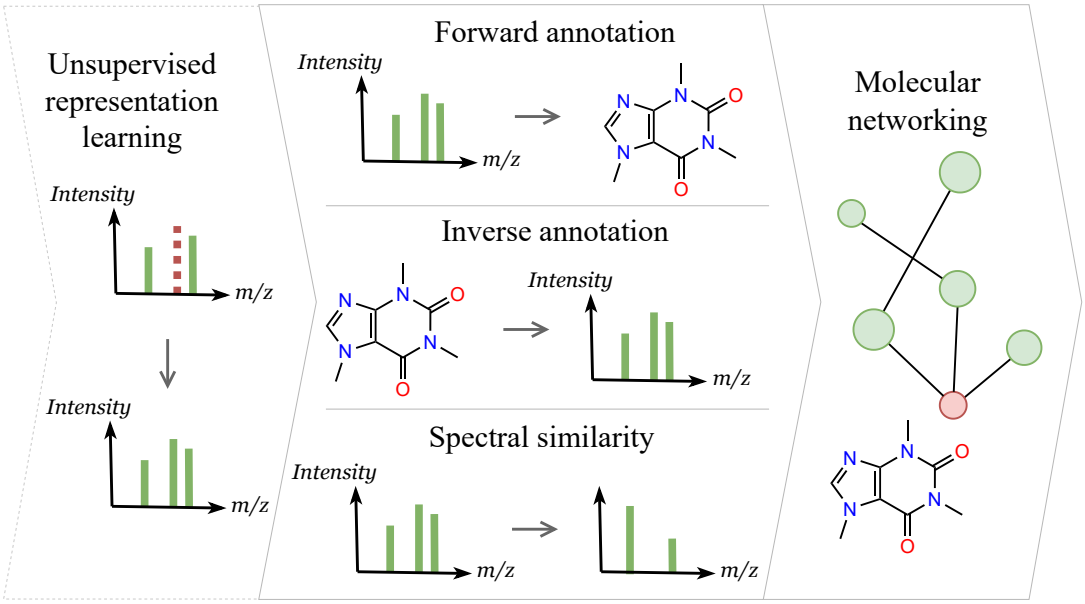


Figure 3.1: Conceptual systematization of related work framing our method. Forward annotations, inverse annotations, and spectral similarities are employed to create molecular networks. Our method of unsupervised representation learning can serve as a starting point for each of the approaches.

without discussing the entire fields.

3.1 Spectral similarity

Spectral similarity is at the core of computational mass spectrometry methods. With an experimentally measured spectrum, one can determine the underlying molecule if an almost identical spectrum exists in annotated databases, commonly referred to as spectral libraries. In general, it is challenging to define a metric that perfectly reflects the similarity between the underlying compounds. Consequently, there exist at least 43 traditional (non-machine learning-based) algorithms [53], which can be prioritized depending on the specific application. The most widely adopted ones are variations of a dot product between binned spectra [54].

For example, in weighted cosine similarity [55], the i-th element of a binned vector is defined as $\mathbf{m}_i^a \mathbf{i}_i^b$ instead of the standard $\mathbf{i}_i$. Parameters a and b (e.g., $a = 3, b = 0.6$) are typically empirically estimated from data. Modified cosine similarity [56] generalizes the dot product by considering not only the products of peaks within the same m/z bins but also the products of peaks shifted by the difference in precursor m/z values. Such a modification makes the cosine similarity

invariant to distinct adducts.

Recently, contrastive learning methods have been proposed aiming to learn spectral similarity with improved correlation with respect to underlying molecular structures. In essence, this approach involves training a neural network f using a loss function of the form $\mathcal{L}(\mathbf{s}_1, \mathbf{s}_2) = d(f(\mathbf{s}_1), f(\mathbf{s}_2))$, where d is a distance measure that is minimized if spectra $\mathbf{s}_1$ and $\mathbf{s}_2$ correspond to similar molecules and maximized otherwise. Once the neural network is trained, d can be used as a spectral similarity measure. For instance, MS2DeepScore [57] employs a feed-forward network f operating on binned spectra, with cosine similarity as the distance measure d. The minimization/maximization of $\mathcal{L}$ is achieved by minimizing the Euclidean distance between d and the Tanimoto similarity of the corresponding compounds.

Voronov et al. [58] propose a similar approach but with a different neural network architecture, defining f as a standard Transformer [38] encoder that operates on spectral peaks encoded with sinusoidal embeddings [59, 38]. Bittremieux et al. [60] introduce the contrastive learning method GLEAMS to cluster proteomics tandem mass spectra, utilizing a convolutional neural network as f, Euclidean distance as d, and minimization/maximization performed for spectra of identical/different peptides.

Another two methods aiming to advance the spectral similarity are MS2LDA [61] and word2vec [62]. Since, technically, they are not necessarily limited to the sole comparison of mass spectra and are unsupervised, we discuss them in section Unsupervised representation learning.

3.2 Forward annotations

The main drawback of spectral similarity searches is the limited size of spectral libraries. Nevertheless, in principle, possessing a flawless forward predictor, one could generate a spectrum for any molecule of interest. Datasets generated in this way are referred to as *in silico* spectral libraries. Traditional forward methods, such as CFM-ID [63], MAGMA [64], and MetFrag [65], recursively fragment a molecule in a combinatorial way and assign a plausibility score for each fragment candidate based on hand-crafted rules or simple machine-learning techniques. The resulting tandem mass spectrum is determined by the cascade of

fragmentations with the highest plausibility.

Alternatively, deep-learning-based methods employ feed-forward networks to directly predict mass spectra from molecular fingerprints [66] or from molecular graphs using classic graph neural networks such as GCN [67, 68], GAT [69, 68, 70], or GRAPHORMER [71, 72]. Some methods also utilize an ensemble of both approaches [73].

More sophisticated methods reformulate the forward annotation as a prediction of chemical formulas determining m/z values. Goldman et al. propose the SCARF model, which comprises two neural networks: one for predicting the set of chemical formulas constrained by the precursor formula and another for predicting the corresponding intensities [74]. Murphy et al. introduce the GRAFF-MS method, which predicts probability distributions over the space of plausible chemical formulas pruned by the mass decomposition algorithm [75, 76]. Alternatively, Goldman et al. propose the ICEBERG algorithm, which simulates fragmentation similarly to the classic methods but utilizes neural network predictions instead of a hand-crafted scoring [74].

3.3 Inverse annotations

Inverse methods aiming to predict molecules from spectra can be classified into two categories. The first one consists of methods predicting molecular fingerprints or other approximate representations of compounds. The second category consists of *de novo* generative models, which aim to construct entire molecular structures.

3.3.1 Approximate inverse annotations

The SIRIUS software [15], belonging to the first category, can be considered the state-of-the-art method for elucidating mass spectra. It comprises a collection of tools, among which fragmentation trees [16], CSI:FINGERID [17], and CANOPUS [18] are the most prominent ones. Fragmentation trees aim to represent a spectrum as a tree graph, where nodes correspond to chemical formulas of fragments and edges represent associated losses. As a result, the root node of a tree with the highest likelihood represents the chemical formula of the whole molecule. The notion of likelihood is determined by hand-crafted scoring functions through maximum a posteriori estimation.

CSI:FINGERID utilizes fragmentation trees to predict the chemical fingerprint of a molecule. Each bit of the fingerprint is calculated by a separately trained Support Vector Machine (SVM) [77] involving kernels specifically designed for mass spectra and fragmentation trees. The latest version of SIRIUS also incorporates deep kernel learning [78]. Since the molecular fingerprint can be easily computed for any molecule, CSI:FINGERID predictions are often directly used to retrieve the compounds with the most similar fingerprints from chemical databases such as PubChem [79]. Finally, given a CSI:FINGERID fingerprint and the corresponding molecular formula, CANOPUS predicts the chemical class of the compound using a feed-forward neural network.

Although the prediction of chemical formulas is considered to be solved for the most common chemical elements (C, H, N, O, P, S), the accurate retrieval of molecular structures remains challenging. As a result, several deep learning methods have recently been proposed to replace SVMs for predicting CSI:FINGERID fingerprints. Given that the community is still exploring mass spectrometry-specific inductive biases, researchers have investigated both simple feed-forward networks. Since the community is still searching for mass-spectrometry-specific inductive biases, there were investigated either simple feed-forward networks [80, 81] or more advanced architectures utilizing the outputs from traditional methods. Goldman et al. propose MIST [82] - a Transformer neural network modified to operate on chemical formulas assigned by SIRIUS and substructures assigned by MAGMA. Notably, the authors also investigate the contrastive fine-tuning of the model pretrained to predict molecular fingerprints.

Instead of predicting fingerprints, another kind of related works focuses on the prediction of task-specific molecular properties. For instance, MS2PROP [83] predicts a quantitative estimate of drug-likeness (i.e. how likely it is that a molecule is a drug) [84] and synthetic accessibility (i.e. how easy it is to synthesize a molecule) [85] directly from spectra employing a standard Transformer encoder architecture. Likewise, Gebhard et al. [86] experiment with several machine-learning algorithms such as linear regression or XGBoost [87] for the prediction of molecular complexity (MC) [88]. They motivate the prediction of such a property by highlighting the efficiency of mass-spectrometry-based search of biosignatures (i.e. life signals) beyond Earth.

3.3.2 *De novo* inverse annotations

Despite the current practicability of the approximate approaches, they are not as conceptually appealing as *de novo* methods directly generating molecular structures. Indeed, knowing a molecular structure, one could easily compute any chemical fingerprint or the above-mentioned molecular property.

There were proposed several works formulating this problem as a Transformer-based translation between mass spectra and SMILES strings. For example, Zhang et al. propose MASSGENIE [89] which is a standard Transformer architecture augmented with VAE-SIM [90] - SMILES-based variational autoencoder [91] helping to search for candidate molecules. Shrivastava et al. experiment with MS^2-TRANSFORMER [92] - a slightly modified Transformer architecture utilizing fragmentation trees. Each Transformer encoder layer sum up the output of Attention block (i.e. Global Aggregation) operating on a set of embedded peak with the output of MPNN (message passing neural network [93]; i.e. Local Aggregation) operating on embedded peaks structured in the corresponding fragmentation tree graph. Another approach employing SIRIUS outputs is MSNOVELIST [94]. Given a CSI:FINGERID and a molecular formula, it autoregressively decodes SMILES in a series of LSTM (long short-term memory) [95] blocks. Another related sequence-to-sequence approach CASANOVO was proposed for proteomics. It utilizes the Transformer architecture, where the encoder operates on peaks and the decoder generates peptide sequences - strings over the vocabulary of over 20 amino acids.

3.4 Molecular networking

Unfortunately, current computational tools have not yet reached the capacity to provide accurate automated solutions for comprehensively interpreting mass spectrometry data. The most widely employed method for the annotation of mass spectra involves combining spectral similarity, inverse, and forward predictions for further manual inspection. Typically, a collection of mass spectra (e.g., spectra from several LC-MS/MS runs or spectra of interest combined with spectral libraries) is examined as a molecular network (MN). In the network, edges represent spectral similarity, while nodes are annotated using inverse methods. Forward methods may further expand the network.

The original molecular networking approach [56] employs modified cosine

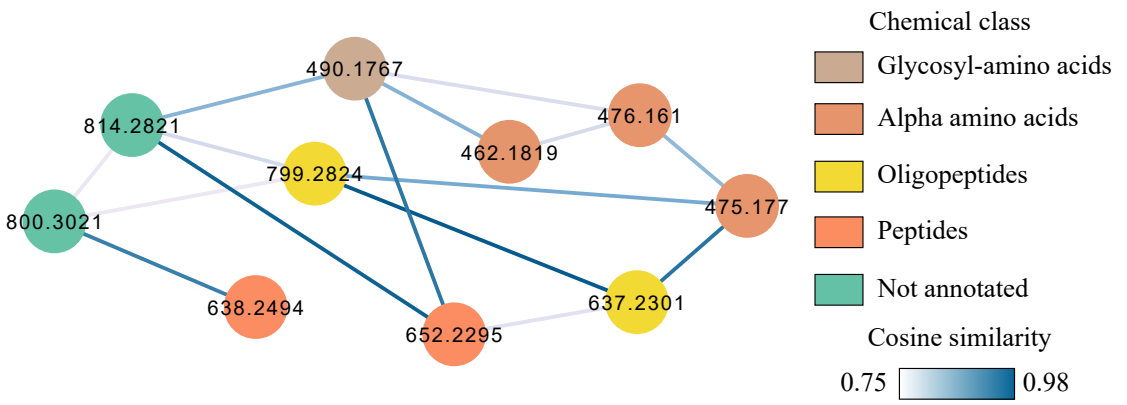


Figure 3.2: The fragment of a molecular network. Each node is associated with a tandem mass spectrum and edge colors represent their cosine similarities. The node values correspond to precursor m/z ratios, and their colors denote the CANOPUS predictions of chemical classes. The molecular networking approach enables propagating the predictions to the spectra, which were not confidently annotated by CANOPUS, suggesting the peptidic compounds as the annotation for the green nodes. The underlying mass spectrometry dataset belongs to an unpublished study on plant metabolome conducted by Tito Damiani et al.

similarity to generate edges. However, there are various orthogonal modifications or extensions to the original node similarity concept. For instance, MetGem [96] is based on t-SNE dimensionality reduction [97] of pairwise cosine distances, and feature-based molecular networking [98] extends MN by considering MS^1 features such as retention time or ion mobility separation [79]. Ion identity molecular networking additionally takes into account the correlation within different adduct species of the same molecule [99]. Overall, molecular networking is a flexible and powerful tool [100] that has successfully demonstrated its applicability in numerous biological studies [56, 101, 102, 103, 104].

3.5 Unsupervised representation learning

One way or another, all aforementioned methods rely on annotated spectral libraries. To the best of our knowledge, there exist only three works at least partially aiming to learn representations of mass spectra directly from unlabeled data. Namely, MS2LDA [61] and SPEC2VEC [62] are especially relevant to our method. These works are inspired by natural language processing algorithms LDA [105] and WORD2VEC [106], which are precursors of BERT [39] inspiring our work.

LDA (2003) ⟶ word2vec (2013) ⟶ BERT (2018)

MS2LDA (2016) ⟶ spec2vec (2021) ⟶ **Our work** (2023)

In essence, MS2LDA treats each spectrum ("document") as a set of Mass2Motifs ("words") - discretized masses of peaks and neutral losses. Then, it statistically searches for the reoccurring motifs determining "topics" of mass spectra. Such derived "topics" are able to encode both generic and specific biologically-relevant structural features and therefore enhance the annotation of mass spectra. Another algorithm spec2vec represents each spectrum in a similar way but derives representations of spectra by implementing the word2vec continuous bag of words approach. The concept is to train a shallow neural network predicting an m/z value of a peak given m/z values of (e.g. 500) neighboring peaks. As a result of such unsupervised[5] training, cosine distance on representations of spectra extracted from the neural network outperforms traditional spectral similarities in terms of structural similarity between underlying molecules.

A different machine-learning technique utilizing unannotated data is proposed by Kutuzova et al. [107]. They investigate a bi-modal variational autoencoder [108] jointly operating on mass spectra and the associated SMILES strings. Prior to bi-modal training, the authors experiment with uni-modal pre-training on both unlabeled data modalities. They show that using compounds unpaired with mass spectra improves the annotation of molecules from spectra. However, uni-modal pre-training on unannotated spectra worsens model performance in the examined experiments. The authors hypothesize that such a drawback could be caused by the heterogeneity of experimental MS data.

Similar to how word2vec is unable to capture long-range dependencies between words and LDA relies on assumptions about topic distributions, spec2vec and MS2LDA generate limited representations of mass spectra and cannot be trained on large datasets. Consequently, they have only been investigated in the context of spectral similarity. In contrast, BERT is a Transformer-based self-supervised learning algorithm that is easily scalable to large text corpora. Although it is also trained to predict masked parts of input sentences, it produces significantly richer representations and offers more flexibility for use in various tasks. BERT has truly revolutionized the domain of natural language processing, thereby motivating our work in mass spectrometry.

[5] Technically, spec2vec is trained on annotated data and the information about molecular structures is used to define datasets and estimate model hyperparameters. Nevertheless, similarly, as word2vec is conceptually designed as an unsupervised algorithm, spec2vec does not use any annotations for training.

CHAPTER 4

Training data collection and analysis

4.1 Analysis and processing of annotated NIST20 and MoNA datasets

As mentioned in the previous chapter, nearly all machine learning models for mass spectrometry are supervised and therefore depend on annotated spectral libraries. The two most prominent and easily accessible annotated datasets are the Mass Bank of North America (MoNA) [25] and NIST20 [24]. In this section, we briefly analyze both libraries and demonstrate that they are not rich enough for exclusively supervised approaches. Nevertheless, they represent the primary source of molecular information available for mass spectra. Consequently, we also use them to validate the effectiveness of pre-training and supervised fine-tuning of the pre-trained model. To make the best use of these libraries in a supervised setup, we develop a novel approach for defining train/validation splits. Our splitting method surpasses existing approaches in both qualitative and quantitative aspects.

4.1.1 Basic statistics of spectral libraries

NIST20 is a dataset created by a team of experienced mass spectrometrists at the National Institute of Standards and Technology (NIST). Over the course of three decades, the dataset has been continuously expanded and thoroughly evaluated, ensuring that every spectrum is accurately measured and examined for correctness. In contrast, MoNA is a community-driven repository of spectra that is curated in an automatic way. MoNA includes more than 20 spectral libraries, among which MassBank and GNPS are the largest and most notable. These libraries form

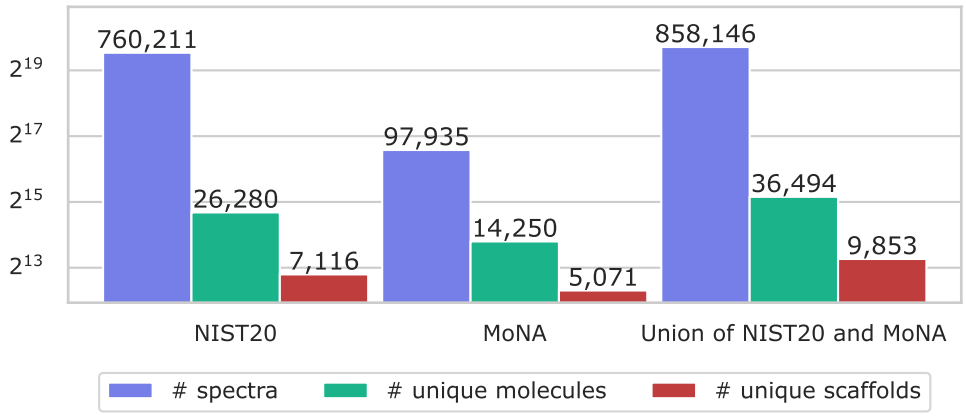


Figure 4.1: Counts of unique mass spectra, molecules, and molecular scaffolds in NIST20 and MoNA datasets. Notice the exponential scale highlighting the overabundance of similar molecular structures.

the core training data for machine learning algorithms, with additional datasets being rarely used.

There is a certain property of spectral libraries which, we believe, makes them quite characteristic in the context of machine learning datasets. For most molecules, one can find at least several but not more than tens of spectra. This observation underscores both the advantages and limitations of utilizing such datasets for training.

In terms of the number of mass spectra, both datasets are relatively large comprising roughly 850 thousand spectra merged together[6]. However, in terms of distinct molecules, the datasets are rather small. The union of the two datasets contains approximately 36 thousand unique compounds, which is a tiny fraction of the known 111 million substances in PubChem [79]. Furthermore, a significant proportion of the included molecules are mutual derivatives (Figure 1.1), yielding less than 10,000 unique molecular scaffolds. Important to mention, that the distribution of precursor masses is skewed towards the range [1, 500] Da leading to mere 6,342 larger unique compounds and 2,936 unique scaffolds in the range of [500, 1000] Da (Figure 4.2). Such a limited inclusion of molecular structures makes it challenging for machine learning models to extrapolate to the level of *de novo* prediction of arbitrary molecules.

Such redundancy arises from the richness of available mass spectrometry se-

[6]For both libraries we consider only spectra acquired in positive ionization mode. We use the LC-MS/MS Positive Mode library of MoNA downloaded on 8 Mar. 2023.

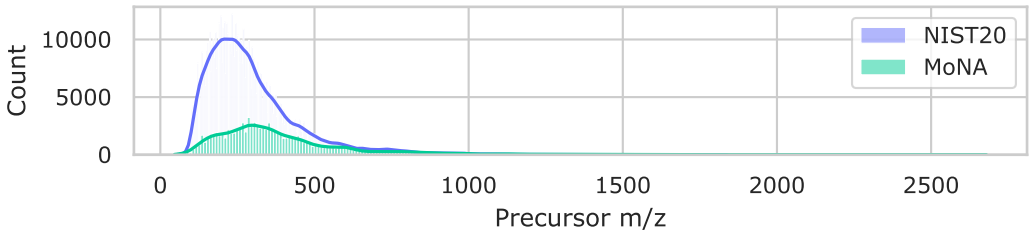


Figure 4.2: Distribution of precursor m/z ratios in NIST20 and MoNA datasets. Spectral libraries poorly cover the molecules within the mass range of [500, 1000] Da.

tups for the same compound. In particular, MoNA and NIST20 provide spectra acquired with different instruments and CID fragmentation energies, as well as spectra with the same precursor molecules but various adducts and charge states (Figure 4.3). In principle, it may help a machine learning model to learn "the manifold" of plausible spectra. However, the common practice prevailing in the literature is to reduce the datasets to consistent data points, such as only spectra of precursors with [M+H]+ adduct.

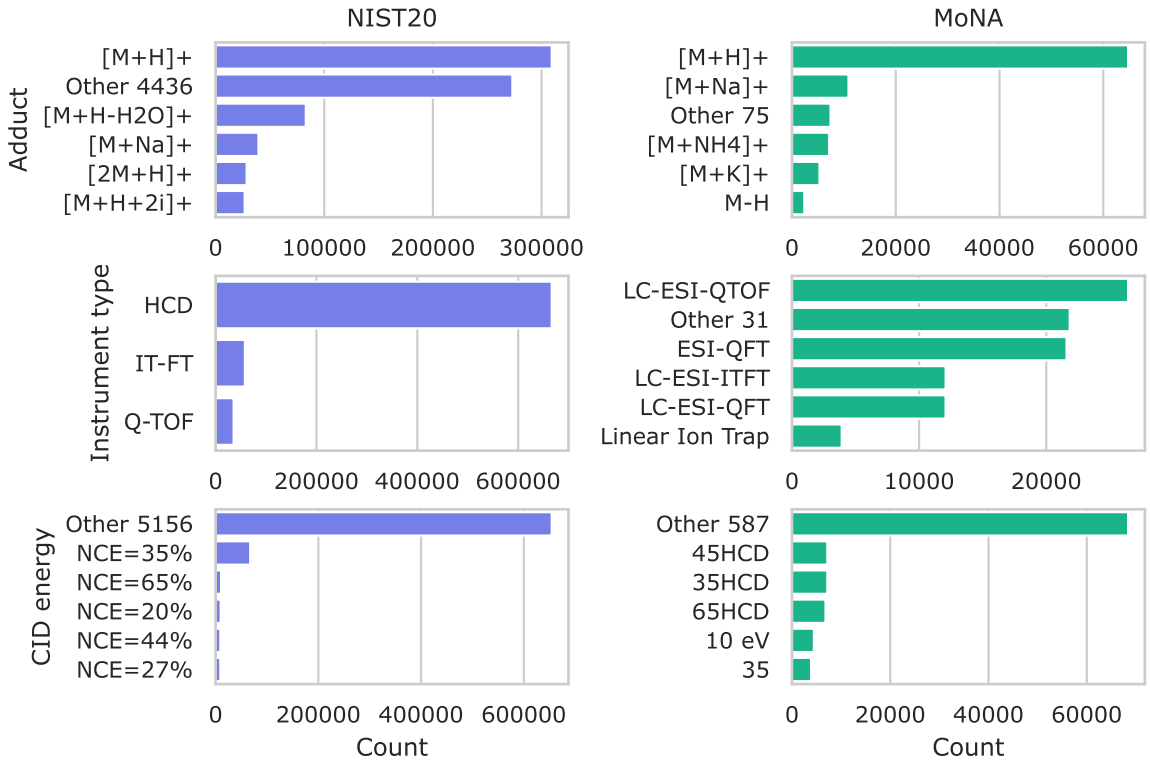


Figure 4.3: Histograms of spectral metadata entities in NIST20 and MoNA. The variety of mass spectrometry settings is the reason and the cause of overabundance of similar molecular structures.

4.1.2 Data splitting with Murcko histograms

Suppose we aim to develop a new machine learning method for the *de novo* inverse annotation of mass spectra (see *De novo* inverse annotations). Regardless of the methodology we implement, the goal of training is to obtain a model that can extrapolate beyond the training data and generate valid molecules for previously unobserved spectra. How can we validate such a model? To achieve this, the computational mass spectrometry community organizes regular CASMI (Critical Assessment of Small Molecule Identification) contests [109], in which models are evaluated on several benchmarks and novel mass spectrometry data. However, these competitions are held at most once a year and are designed primarily as the ultimate test of model performance rather than for their gradual development.

With the current limited capacity of spectral libraries, one has to employ a standard machine-learning technique of training data splitting, where a fraction of the available data is used as a held-out validation benchmark and excluded from training. Frequently, this fraction is determined by random subsampling. However, this often results in biased datasets, leading to biased models that fail to generalize to unobserved data [110]. In the mass spectrometry domain, non-random train and validation folds are currently obtained using either structure-disjoint or scaffold-disjoint splitting. The structure-disjoint split separates molecules with non-identical connectivity of atoms (i.e. the first 14 characters of InChI keys), while the scaffold-disjoint split separates molecules with non-identical Murcko scaffolds [111]. A Murcko scaffold is a core molecular structure obtained by removing side chains and retaining only the essential ring systems and linkers that define the overall architecture of a molecule.

Let us consider the molecules from NIST20 shown in Figure 4.4 as. If we define the *de novo* inverse annotation as a graph generation problem (for instance, by defining a loss function as the mean binary cross-entropy over the adjacency matrix of the molecular graph), we can see that molecules (A) - (D) are quite similar to each other. Therefore, the *de novo* predictor generating molecule (B) from the spectrum of (C) can be considered reasonably effective. However, to the best of our knowledge, none of the existing train/validation splitting approaches is capable of grouping all four molecules in the same fold. Consequently, given the entire NIST20 dataset, a model is always trained on some of the four molecules and validated on the remaining ones. This implies that if a model can find some trivial

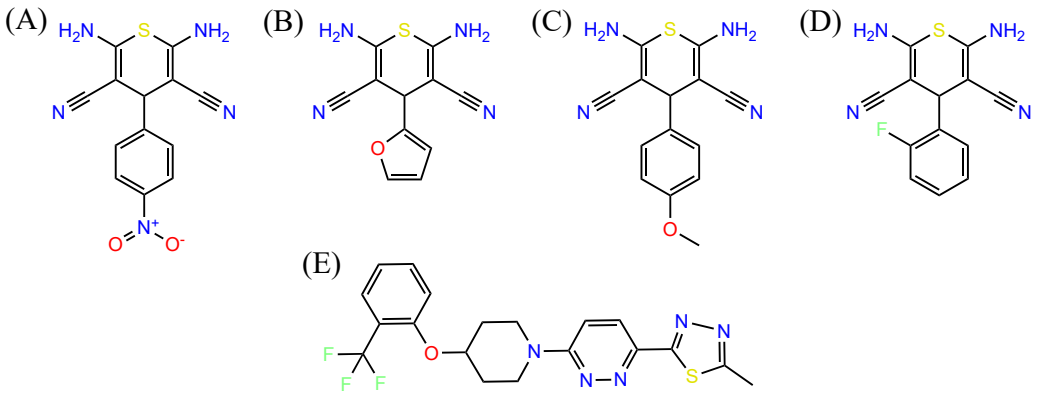


Figure 4.4: Four molecular structures from NIST20 illustrating the shortcomings of currently applied train/validation splitting techniques. (A-D) Four molecules from NIST20, which can be characterized by two spectral peaks at 177.0223 and 101.0163 m/z values. The combination of the two peaks does not occur in any other spectrum within NIST20 and MoNA. **(E)** Molecule from the unpublished in-house MCE library prepared by Corinna Brungs et al. yielding the aforementioned combination of m/z values[7].

pattern that uniquely characterizes the four molecules in the scope of training data (i.e. overfit), it can achieve relatively high validation performance. In fact, an example of such a pattern is the presence of two peaks at 177.0223 and 101.0163 m/z values. However, given the vast combinatorial space of plausible molecular structures, an overfitted model obviously cannot extrapolate beyond NIST20. For instance, a molecule from our in-house library yields the same combination of m/z values but lacks the distinctive nitrogenated Thiane ring. It's worth noting that this analysis is not far-fetched, but rather a more in-depth investigation of a single cluster depicted in Figure 1.1.

The observations described above motivate us to develop a new train/validation scheme. First, we note that existing splits are agnostic to the fragmentation nature of tandem mass spectrometry. For instance, two molecules differing only in the length of the carbon chain connecting two subfragments possess distinct structures and scaffolds, yet such chains can be easily fragmented by CID, resulting in nearly identical spectra. Moreover, contemporary approaches often allocate entire molecules and their abundant fragments (which appear as separate data samples) to different folds (see bottom examples in Figure 1.1), which consequently makes machine learning models susceptible to overfitting.

Secondly, we believe that the splitting methods currently employed are concep-

[7]Here m/z values are considered to be distinct up to the absolute difference of $5 * 10^{-4}$.

tually flawed. While, by definition, the design of train/validation splits should start with the question of "How to distribute similar molecules across different folds?", existing methods instead seek to answer "How to cluster different molecules within the same folds?". More exactly, the distinction between the two approaches appears in the number of compounds per fold, where the fold is purely the output of the splitting algorithm. The latter approach often results in the abundance of clusters containing almost identical molecules, which must be further combined to form a meaningful amount of data folds. However, since the underlying methodology does not differentiate but unify molecules, they are often combined randomly, making one unaware of possible data leakages.

To address the issue of insensitivity to fragmentation, our method operates on molecular fragments as the primary design principle. To address the second issue, we define a heavily relaxed notion of molecular similarity, such that the distinction between folds is well-defined and easily understood. We term our algorithm Murcko histograms since it operates on the histograms associated with Murcko scaffolds.

Given the Murcko scaffold of a molecule, algorithm Data splitting with Murcko histograms operates on two separate groups of its atoms. The first group consists of sets of atoms, with each set determining a ring, whereas the second group includes all atoms connecting these rings. For each ring, the algorithm calculates a pair of natural numbers: the number of neighboring rings and the number of adjacent linkers. These pairs define the domain of the resulting histogram, where the values represent the counts of such pairs within a molecule. Figure 4.5 provides examples of molecules and their corresponding Murcko histograms.

In a straightforward manner, two molecules can be considered similar if they have identical Murcko histograms and be considered dissimilar otherwise. This process resembles a single iteration of the Weisfeiler-Lehman (WL) graph isomorphism test [112] on a coarse-grained molecular graph, where rings are collapsed into single nodes and are colored differently from linkers. However, a notable difference between Murcko histograms and the histograms arising in the WL test is that Murcko histograms are calculated solely for rings. This approach makes the algorithm invariant to the lengths of linkers, which is a desired property for assessing molecular similarity with respect to fragmentation mass spectra.

Algorithm 1: Definition of Murcko histogram

Input: Molecular graph
$G = (V, E), V = \{1, \ldots, n\}, E \subseteq \{\{u, v\} \mid u, v \in V \wedge u \neq v\}$.
Output: Murcko histogram h.

$G \leftarrow$ MurckoScaffold(G)
$V_R \leftarrow \{V_r \subset V \mid V_r$ contains nodes of all cycles sharing at least two edges$\}$
$V_L \leftarrow \{v \in V \mid \deg(v) > 1 \wedge v$ is in not in any cycle$\}$
$h \leftarrow$ a map $\mathbb{N}^2 \to \mathbb{N}$ initialized as $(\forall i, j \in \mathbb{N}^2)(h(i, j) = 0)$
for $V_r \in V_R$ **do**
    $r \leftarrow \sum\{|V_r \cap V_r^{'}/2| \mid V_r^{'} \in V_R \setminus V_r\}$
    $l \leftarrow |V_r \cap V_l|$
    $h(r, l) \leftarrow h(r, l) + 1$
return h

To further refine the similarity measure with respect to mass spectrometry, we establish a more lenient relation on Murcko histograms than strict identity. Specifically, we consider two molecules as related if the Murcko histogram of one molecule is a subhistogram of the other solely in rings. Additionally, two molecules are considered similar if they are transitively related, regardless of the symmetry of chained relations. To prevent the collapse of all molecules into a single fold, this relation is replaced with an identity if at least one of the two molecules being examined in a pair has fewer than k rings. We set $k = 3$ by default. This relaxed approach enables the similarity measure to group subfragments of the same compound. Figure 4.7 illustrates an example of three such molecules.

While the strict identity on Murcko histograms clusters the entire NIST20 dataset

Algorithm 2: Definition of relation on Murcko histograms

Input: Two Murcko histograms h_1, h_2, minimum number of rings k to compute the non-identity relation.
Output: True if one of h_1, h_2 is a subhistogram of another in Murcko rings, False otherwise.

if $\min\{\sum_{i,j\in\mathbb{N}} h_1(i, j), \sum_{i,j\in\mathbb{N}} h_2(i, j)\} \leq k$ **then**
    return $h_1 = h_2$
if $(\forall i \in \mathbb{N})(\sum_{j\in\mathbb{N}} h_1(i, j) \leq \sum_{j\in\mathbb{N}} h_2(i, j))$ **then**
    return True
if $(\forall i \in \mathbb{N})(\sum_{j\in\mathbb{N}} h_2(i, j) \leq \sum_{j\in\mathbb{N}} h_1(i, j))$ **then**
    return True
return False

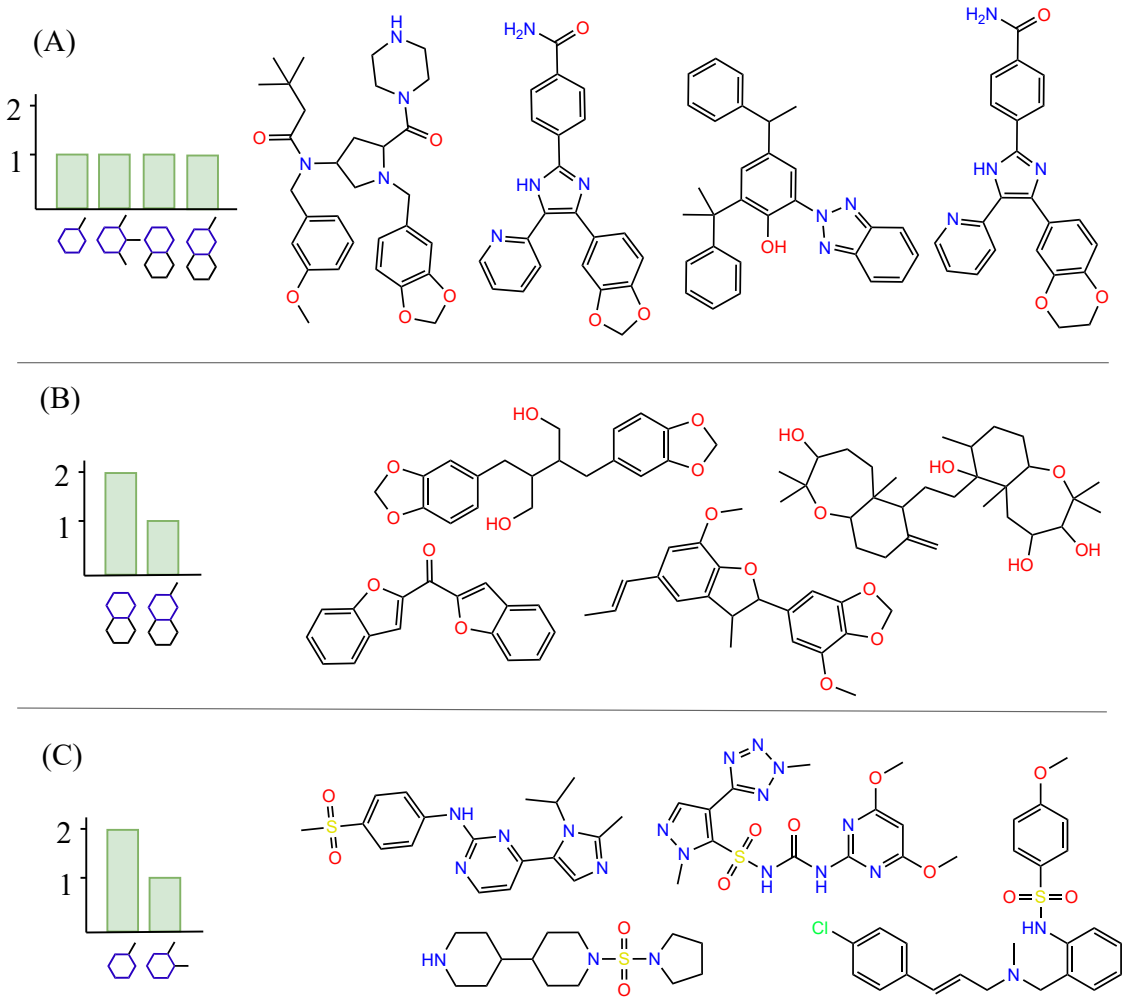


Figure 4.5: Groups of NIST20 molecules sharing identical Murcko histograms. Schematic molecules under the histograms denote the number of adjacent rings and linkers. Notice, that in the context of structure- and scaffold-disjoint splitting, all the molecules are regarded as distinct.

into 314 equivalence classes, the relaxed similarity relation results in 39 unique components. Of these, four clusters account for 99.5% of the molecules (39% + 29% + 17% + 15%), with the remaining ones featuring complex fused rings that make them outliers in terms of Murcko histograms. By isolating the group representing 15% of compounds, we achieve a 73%/27% train/validation split, significantly alleviating data leakage issues compared to existing approaches (Figure 4.6). Importantly, we can interpret the possible leakage given by our method as necessarily the pair of compounds, among which one of the compounds is a fragment of another and contains less than k rings.

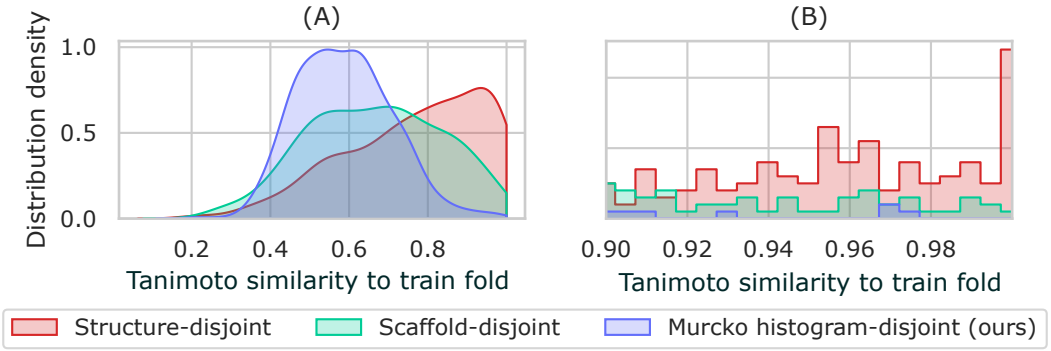


Figure 4.6: Murcko histograms surpass currently applied data splitting techniques in terms of Tanimoto similarity. **(A)** Distribution density of maximum Tanimoto distances when searching for the most similar training-fold molecules for 500 random validation-fold samples within NIST20. **(B)** Scaled [0.9, 1] domain from Figure (A). The commonly-applied structure-disjoint split is inadequate as a validation of machine learning algorithms, as it results in many data leakages (i.e., 1.0 similarities between training and validation examples). While the scaffold-disjoint splitting mitigates the leakages, its distribution is skewed towards the [0.8, 1.0] range. In contrast, Murcko histograms result in a distribution centered around 0.55 and rarely produce similarities above 0.9.

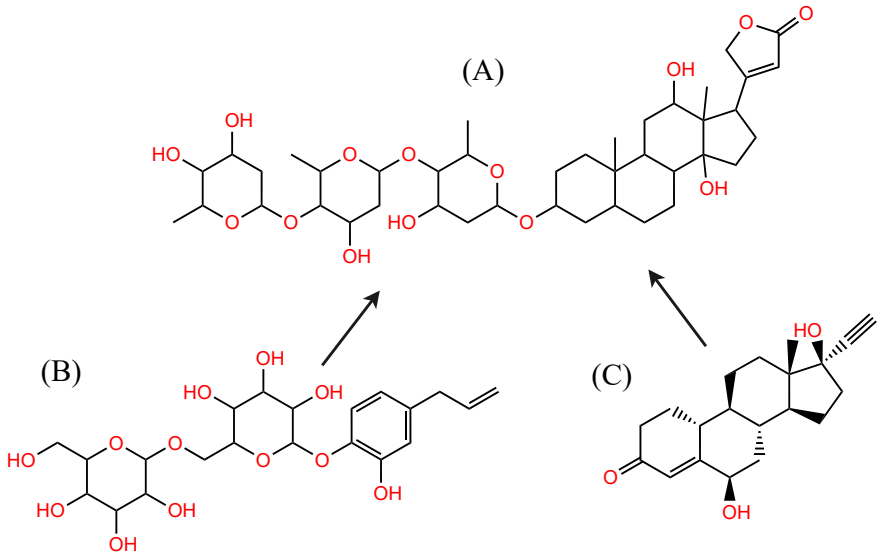


Figure 4.7: Murcko histograms relation links subfragments within identical folds. The figure shows molecules from the NIST20 dataset with different Murcko histograms that belong to the same connected component in the graph defined by the subhistogram relation. Although molecules **(B)** and **(C)** do not share mutually inclusive Murcko histograms, the histograms of both **(B)** and **(C)** are subhistograms of **(A)**, resulting in the shortest undirected path between **(B)** and **(C)**.

4.2 Mining millions of unannotated MS^n spectra from MassIVE repository

To overcome the limitation of spectral libraries, we collect an orders-of-magnitude-larger dataset of high-quality unannotated mass spectra from the MassIVE repository. MassIVE[8] is a community resource developed by the NIH-funded Center for Computational Mass Spectrometry to promote the global, free exchange of mass spectrometry data. The repository contains raw and processed data and is designed to support a wide range of mass spectrometry-based studies, including proteomics, metabolomics, and lipidomics. MassIVE contains data from various mass spectrometry instruments and data formats, making it a versatile and comprehensive database. Currently, it comprises around 500TB of data containing 6.4 billion mass spectra. Even though historically MassIVE was developed as a resource for proteomics, it has recently started to rapidly expand as a metabolomics database (Figure 4.8).

The MassIVE repository is comprised of datasets, each corresponding to a specific applied mass spectrometry study and possessing a unique MSV identifier. For example, MSV000091421 dataset studies metabolomics of blood samples for Asthma cohort, MSV000090317 comprises a mass spectrometry analysis of the roots of Vietnamese *Fibraurea recisa* plant, whereas MSV000081119 contains the study of beer samples from different locations. A dataset is essentially a directory of files and does not undergo strict specification. Any registered user can upload a dataset, regardless of the purpose of the conducted experiments or the quality of MS data. Such broad inclusivity is crucial for achieving our goal of collecting a diverse and extensive dataset. However, the raw nature of the data, along with its immense scale, presents a series of challenges:

- How can we extract a subset of high-quality MS^n spectra? Given the variability of experimental setups, how can we define a "high quality MS^n spectrum"?
- How can we minimize redundancy in the extracted dataset and make it balanced with respect to unknown fragmented molecules?
- Given MassIVE's bias towards proteomics, how can we identify mass spectra corresponding to metabolites?

[8] https://massive.ucsd.edu

In the following sections, we describe our approach to overcoming these challenges and introduce a pipeline of algorithms that enable us to extract large refined collections of unannotated MS^n spectra from MassIVE, which we refer to as MSV^n datasets.

4.2.1 Collecting metabolomics data files

Contradictorily, despite not limiting our work to a specific class of molecules, we currently have to exclude the majority of MassIVE's volume, which predominantly corresponds to proteomics research studies. This implies that most tandem mass spectra in MassIVE are associated with peptides. Although such spectra are of great interest in their own right, incorporating them into our work would limit the generalization capabilities of the deep learning model and introduce a bias towards peptides. Regrettably, given the metadata available in MassIVE, distinguishing between metabolomics and proteomics studies is challenging. Moreover, metabolomics samples may contain a variety of peptides in addition to the anticipated metabolites.

Fortunately, the metabolomics community typically follows the convention of naming metabolomics datasets with the "GNPS" prefix [100]. Thus, we further concentrate only on GNPS datasets within MassIVE. However, we emphasize that

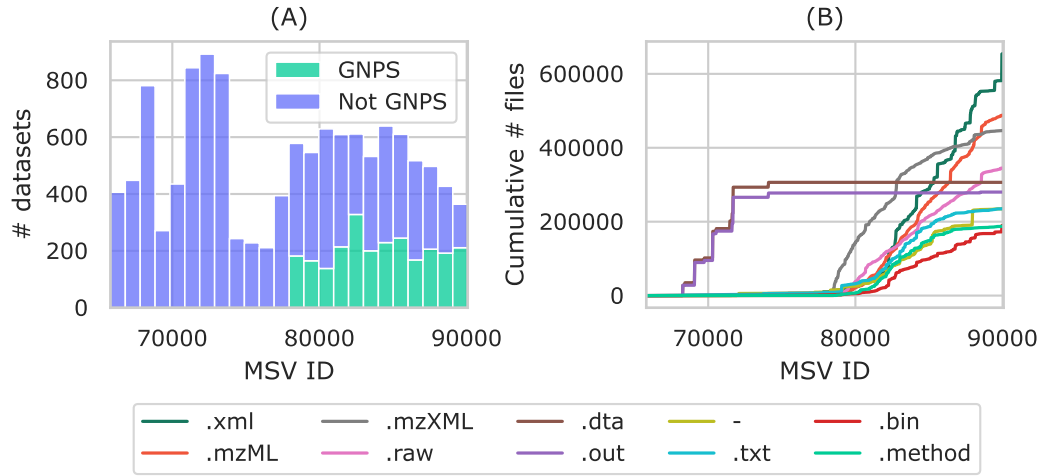


Figure 4.8: Growth of MassIVE as a standardized repository for metabolomics data. **(A)** Starting with approximately the MSV ID of 80,000 MassIVE began consistently increasing in the number of GNPS datasets containing `.mzML` and `.mzXML` files with metabolomics data. **(B)** Along with the emergence of GNPS datasets MassIVE started to converge to two aforementioned file standards (`.xml` files describe metadata and `.raw` files are automatically converted to `.mzML`).

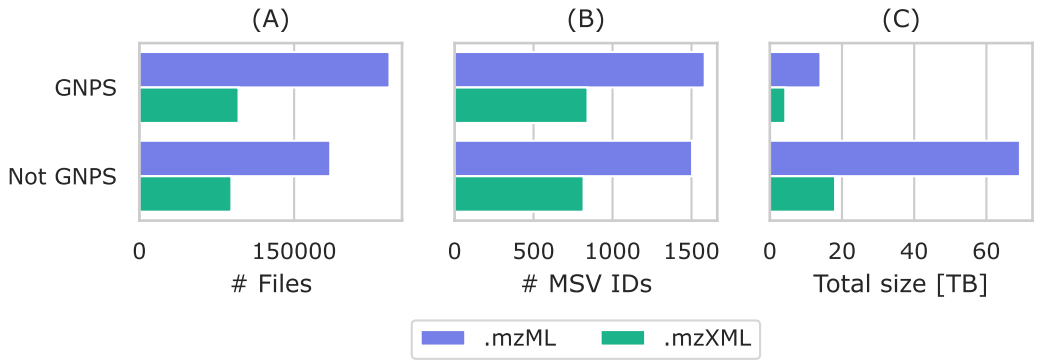


Figure 4.9: Data file-level statistics of the GNPS part of MassIVE. Even though the GNPS part contains approximately the same number of files as its complementary part, GNPS occupies considerably less storage space. Such observation indicates the diversity of metabolomics data files.

estimating the number of metabolomics datasets present in the complementary non-GNPS portion of MassIVE is difficult, as this information is only available in the form of text descriptions.

Each vendor of mass spectrometry instruments provides its own data format, resulting in MassIVE datasets being composed of various file formats. Despite the variability in file extensions, two major data formats have become dominant (Figure 4.8) across MassIVE: `.mzML` [113] and its predecessor, `.mzXML` [114]. These formats were developed with the full participation of vendors and researchers in order to create a single open specification for mass spectrometry data. The dominance of these formats is also a consequence of a feature of MassIVE, which automatically converts raw data formats (such as `.RAW` from Waters or `.TDF` from Bruker Daltonics) to `.mzML`. As a result, we extract spectra only from files that are in the `.mzML` and `.mzXML` formats. Essentially, these files contain three-dimensional mass spectrometry datasets, as introduced in section Liquid chromatography tandem mass spectrometry, where two-dimensional mass spectra are ordered by chromatographic retention time.

After selecting only GNPS datasets and `.mzML`, `.mzXML` files we obtain 649,494 files occupying 32.51 TB. To eliminate identical or similar files, we select only the most recent file among files with the same name and MSV ID but located in different subdirectories. It yields the final 338,649 files occupying 18.35 TB.

4.2.2 Processing and storage of MS data files

Once we have identified the MassIVE files of interest, we download and process each of them. We perform on-the-fly processing, retaining only a specific portion of the information. In this phase, our goal is to preserve nearly all MS^n spectra while discarding MS^1 spectra not triggered for MS^2 acquisition. We also carry out basic filtering to eliminate empty or corrupted spectra. During this workflow, we (i) gather metadata that can be utilized by machine learning algorithms, and (ii) compute data quality indicators that enable us to further construct high-quality subdatasets. Finally, we store the spectra in the `.hdf5` format, which is suitable for deep learning applications.

Collecting spectra and their metadata

If we were to keep all MS^n spectra from the collected data files, we would have roughly 814 million spectra. However, we discard certain spectra that we consider invalid, despite following the principle of collecting as many tandem spectra as possible regardless of their quality (Figure 4.11). Specifically, we remove approximately 40 million spectra that do not possess any single peak, as well as the other spectra that do not have only unique m/z values, have non-positive values, or do not have m/z values sorted. We also exclude spectra with zero intensities, as they are likely acquired in a profile mode or have undergone an undesired preprocessing step, such as binning of the m/z range. Furthermore, we subject precursor MS^1 spectra to the same filtering conditions, resulting in a reduced dataset of approximately 714 million tandem mass spectra.

Precursor MS^1 spectrum is a primary piece of metadata for each MS^n spectrum. It contains important information about isotopic and adduct distributions of the

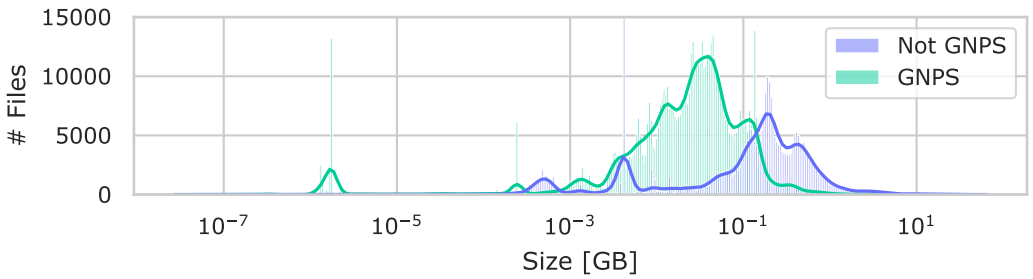


Figure 4.10: GNPS part of MassIVE is balanced regarding the size of data files. The figure shows that the distribution of data file sizes is log-normal-like for the GNPS part of MassIVE and is less balanced for the complementary fraction.

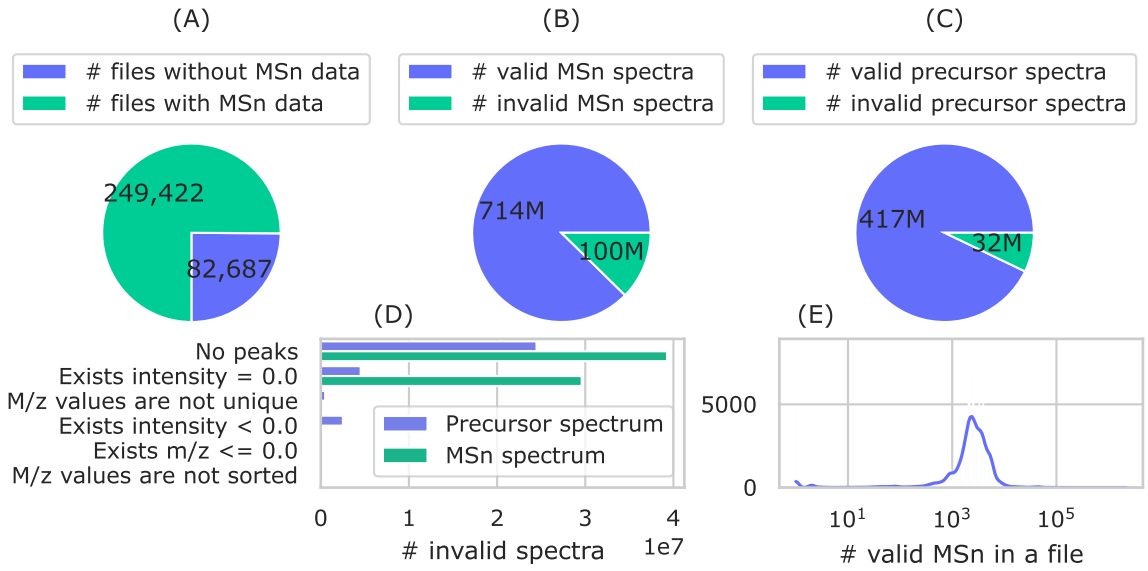


Figure 4.11: Outcomes of filtering invalid or inappropriate data from the GNPS subset of MassIVE. Figure **(A)** reveals that roughly 25% of files are not MS/MS experiments. Figure **(B)** indicates that out of 814 million spectra, 100 million are invalid, while Figure **(C)** presents similar statistics for precursor spectra. Figure **(D)** characterizes the invalid spectra. Figure **(E)** demonstrates that the distribution of valid MS^n spectra across files resembles a log-normal pattern.

precursor molecule and allows, for example, the detection of chimeric spectra by analyzing the density of peaks within the isolation window around precursor m/z. Therefore we collect precursor spectra for all extracted MS^n spectra. Further, we equip tandem mass spectra with "data-file-level" and "spectrum-level" metadata. More precisely, by "data-file-level" we mean the underlying instrument information and the overall structure of the MS dataset (Figure 4.12), which directly reflect the quality of MS data and discussed in the following section Estimating MS data quality.

"Spectrum-level" metadata refers to the information characterizing each individual spectrum. Among such data, the most valuable entries include the m/z of the precursor molecule triggered for fragmentation and the MS level (i.e., n in MS^n) of a spectrum, which determines the number of consecutive recursive acquisitions starting with the same MS^1 spectrum. For any method annotating MS^n spectrum, the precursor m/z is essential as it reveals the mass of the entire fragmented compound. MS levels enable the aggregation of multiple spectra corresponding to the same precursor. This aggregated "gestalt" contains significantly more structural information compared to individual spectra. However, unfortunately, multi-stage MS^n fragmentation data is rarely available (Figure 4.13).

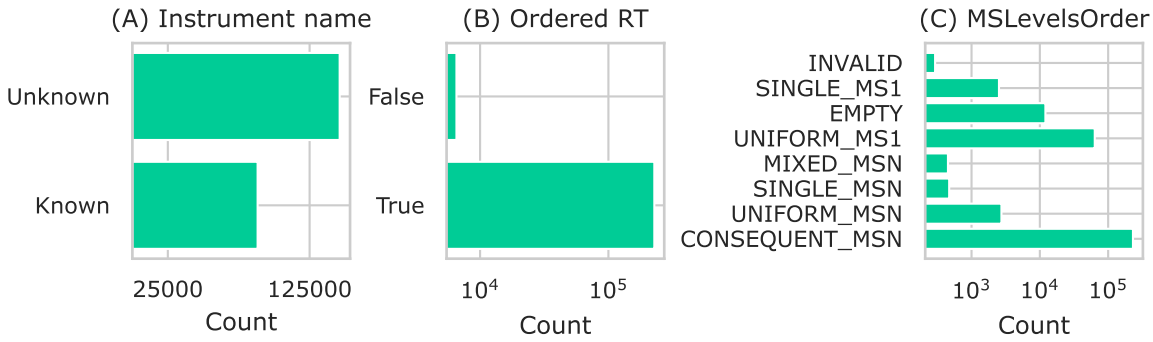


Figure 4.12: Histograms of file-level quality-related metadata entities computed on the downloaded GNPS subset of MassIVE. Figure **(A)** demonstrates that information regarding the MS instrument is often missing. Figure **(B)** indicates that a non-empty fraction of data files do not have their spectra ordered with respect to retention time. Figure **(C)** displays the distribution of spectra ordering by MS level, including non-downloaded types such as `INVALID` or `UNIFORM_MS1`. Notably, a significant portion of files with MS^n contain only a single spectrum (`SINGLE_MSN`) or lack MS^1 data (`UNIFORM_MSN`).

While the precursor m/z and MS level information are always specified in MS datasets, other valuable metadata entities are often unavailable due to their recording being limited to certain MS instrument models. For instance, the precursor charge can provide an annotation method with insights into the molecular structure's size. The number of charge sites, for example, can suggest the diameter and curvature of the molecular graph. However, the charge is unknown in approx-

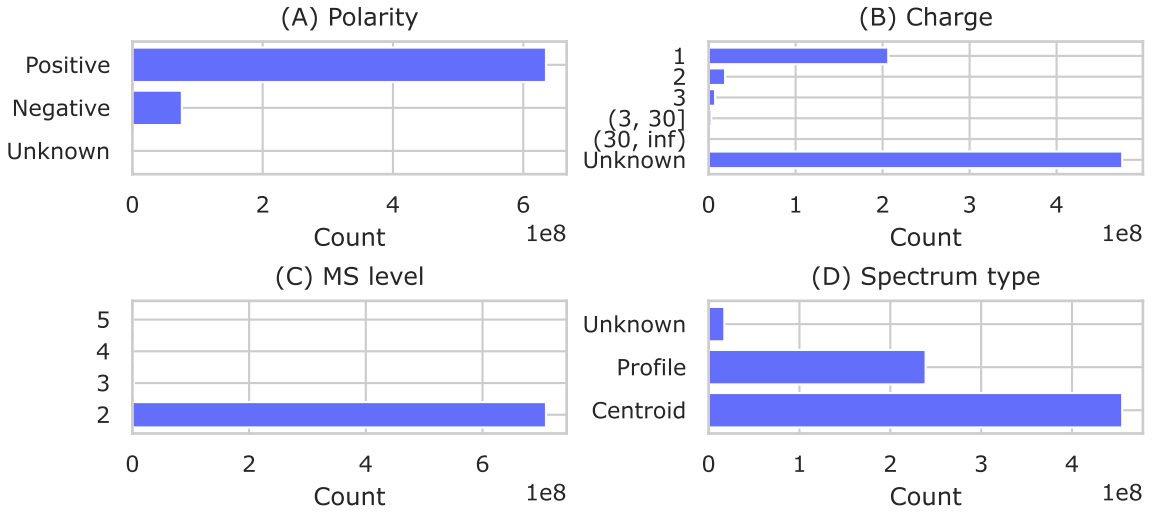


Figure 4.13: Histograms of spectrum-level metadata entities computed on the downloaded GNPS subset of MassIVE. Figure **(A)** demonstrates that the vast majority of spectra are acquired in positive ionization mode. However, Figure **(B)** shows that the precursor charge is frequently unknown. Figure **(C)** indicates that multi-stage MS^n data for $n > 2$ is almost non-existent within the filtered files. Lastly, Figure **(D)** illustrates that a substantial portion of spectra is stored in unprocessed profile mode. Note that all counts represent non-zero values.

imately 66% of cases (Figure 4.13). Similarly, extracting useful information from the CID energy specified in data files is challenging, as it lacks standardization.

Estimating MS data quality

The quality of mass spectrometry (MS) data plays a critical role in computational metabolomics workflows. However, the concept of quality is difficult to define rigorously. In this work, we utilize the term "quality" to refer to the richness of MS data with respect to the information about molecules present in a given sample. This notion of "richness" constitutes a somewhat latent concept and its completeness cannot be evaluated by any contemporary method. Nevertheless, the quality of individual MS datasets can be compared based on certain characteristics.

Some of these characteristics are directly implied by the model of the mass spectrometry instrument. For example, instrument vendors often provide parts per million (ppm) accuracy and resolution specifications, which can be leveraged by MS data processing methods. In particular, the higher the accuracy, the more precise the MS^1 isotopic distributions, which in turn results in a greater richness of MS information. However, instrument characteristics cannot significantly facilitate our large-scale data processing of MassIVE, as the instrument model is unknown in roughly one-third of cases (Figure 4.12). Moreover, other qualitative parameters, such as the number of non-noise peaks in MS^n spectra, are not directly associated with the instrument but rather with its setup by the user. For instance, insufficient collision-induced dissociation (CID) energy can result in only a few peaks (molecular fragments) and an information-poor spectrum. Therefore, we implement several heuristic data-centric algorithms to estimate the quality of MS datasets directly from spectra and their structure.

Given an MS dataset, we first assess its global structure. First of all, it means that we validate the retention time ordering of spectra. If spectra are not ascendingly ordered by RT, we sort them, yet indicating this step in the output data format as may suggest it's low quality. Secondly, we classify the retention time ordering of spectra with respect to MS levels into the following categories:

$$\texttt{MSLevelOrder}(\mathbf{l}) = \begin{cases} \texttt{SINGLE\_MS1}, & \text{if } |\mathbf{l}| = 1 \wedge \mathbf{l}_1 = 1, \\ \texttt{SINGLE\_MSN}, & \text{else if } |\mathbf{l}| = 1 \wedge \mathbf{l}_1 > 1, \\ \texttt{UNIFORM\_MS1}, & \text{else if } \mathbf{l}_i = \mathbf{l}_{i-1} = 1, \\ \texttt{UNIFORM\_MSN}, & \text{else if } \mathbf{l}_i = \mathbf{l}_{i-1} > 1, \\ \texttt{CONSEQUENT\_MSN}, & \text{else if } \mathbf{l}_i - \mathbf{l}_{i-1} \in \{0, 1\} \vee \mathbf{l}_{i-1} = 1, \\ \texttt{MIXED\_MSN}, & \text{else if } \mathbf{l}_i - \mathbf{l}_{i-1} \in \{0, 1\} \cup \mathbb{Z}_-, \\ \texttt{INVALID}, & \text{otherwise}, \end{cases}$$

where $\mathbf{l} \in \mathbb{N}^r$ are MS levels of spectra ordered by r retention times and all conditions involving $\mathbf{l}_i$ hold $\forall i \in \{2, \ldots, r\}$. The rationale behind such an approach is to discard MS^n spectra missing precursor information and to detect corrupted mass spectrometry data files.

For example, we are not interested in `SINGLE_MS1` or `UNIFORM_MSN` categories, which contain only MS^1 data. Further, we classify tandem data into three major types: `UNIFORM_MSN`, `CONSEQUENT_MSN`, and `MIXED_MSN`. While `CONSEQUENT_MSN` type guarantees the clean ordering of spectra, it does not account for multiple MS^3 acquisitions starting with a single MS^1. Therefore, we define `MIXED_MSN` category as MS^n spectra followed by MS^1 without a strictly defined format. Different approaches to internal data handling by the instrument may result in a different ordering of multi-stage fragmentation spectra. Finally, `UNIFORM_MSN` is associated with files containing only tandem data of the same MS level. Such a format may suggest the manual pre-processing of the data and indicate its low quality.

Each spectrum in MassIVE may be in a different format, depending on the instrumental setup. Some spectra may be centroided, while others may remain in a raw profile form. To ensure consistency in our dataset, we aim to predict the type of each spectrum and keep only the centroided ones. To achieve this, we employ the heuristic MZmine 3 algorithm Estimating MS data quality [115]. In essence, the algorithm assesses whether the base peak of a spectrum is represented as a sequence of densely packed high peaks or not. If it is, the spectrum is classified as profile (or thresholded, meaning that it does not contain any zero intensity), and if not, it is classified as centroided.

Although `MSLevelOrder` and spectrum type can filter undesired spectra, these

Algorithm 3: Estimate the type of a spectrum

Input: Spectrum m/z values $\mathbf{m} \in \mathbb{R}^n$ and intensities $\mathbf{i} \in \mathbb{R}^n$.
Output: Estimated type of the spectrum.

if $n < 5$ **then**
  return CENTROID
$b \leftarrow \arg\max \mathbf{i}$
$S \leftarrow \{s \in \{1, \ldots, n\} \mid (\forall s' \in \{0, \ldots, s - b\})(\mathbf{i}_{b+s'} > \frac{\mathbf{i}_b}{2})\}$
if $\max S - \min S < 3$ **or** $\mathbf{m}_{\max S} - \mathbf{m}_{\min S} > \frac{\max \mathbf{m} - \min \mathbf{m}}{1000}$ **then**
  return CENTROID
else
  if $(\exists i \in \mathbf{i})(i = 0)$ **then**
    return PROFILE
  else
    return THRESHOLDED

categorical metrics do not provide any information-richness score. As MS datasets describe samples in terms of molecular masses, we emphasize that the most valuable characteristics of data quality should be expressed in terms of masses. Thus, it is crucial to estimate instrument accuracy for measuring m/z values directly from the data. To achieve this, we implement the Estimating MS data quality algorithm, which takes an MS dataset as input and produces a real-valued score estimating the absolute accuracy of the underlying instrument.

The proposed algorithm for estimating instrument accuracy is based on calculating the variance of the same m/z value across different retention time (RT) steps. To achieve this, the algorithm constructs an extracted ion chromatogram (XIC), which is a slice of a 3D mass spectrometry dataset that is orthogonal to a certain m/z ratio. The width of the slice is defined by a parameter called the m/z tolerance, which determines the notion of being ”the same”. Initially, the algorithm collects the masses of the highest peaks within the dataset and constructs an XIC for each peak, using a high m/z tolerance. This step allows the algorithm to determine putative sequences of m/z values that correspond to the same molecule.

Next, the algorithm constructs a new collection of narrower XICs, starting with the median m/z of each individual XIC from the first round. The median standard deviation of the resulting long sequences is considered the estimation of the absolute instrument accuracy. This process helps to identify the m/z values that deviate from their actual values due to various factors, including instrument imprecision

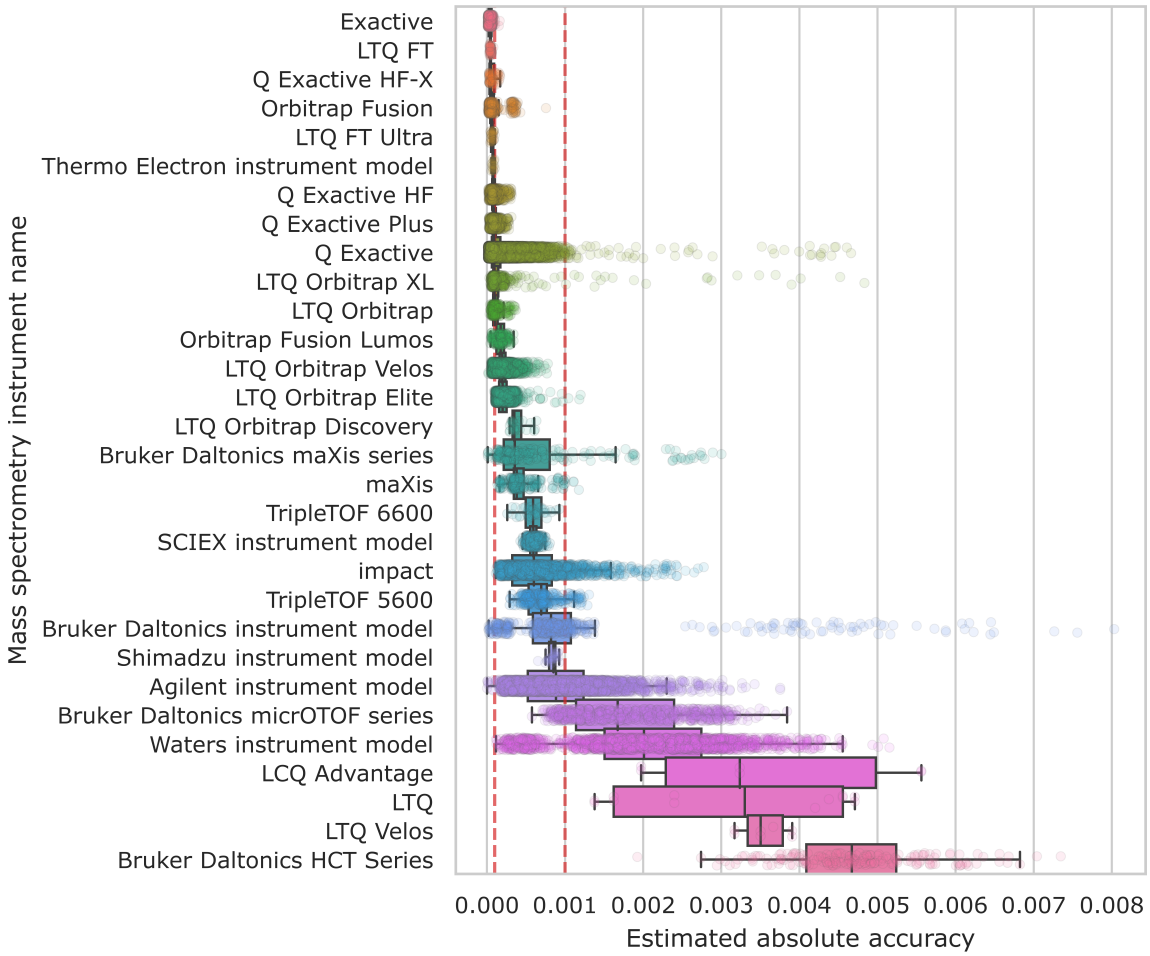


Figure 4.14: The red vertical lines depict two thresholds, which we fuzzily use to distinguish ”high” accuracy (below 10^{-4}) from ”medium” (below 10^{-3}) and ”low” accuracies (above 10^{-4}). Note that the figure does not aim to validate the instruments, but rather to assess the adequacy of the accuracy estimation. In fact, the accuracy of instruments such as TripleTOF depends on user-defined parameters.

or systematic bias. Figure 4.14 demonstrates the estimations of the m/z accuracy for different MS instruments.

In addition to the global dataset-level qualitative features, we calculate local spectrum-level properties to estimate the quality of individual spectra. One such property is the number of high-intensity[9] peaks present within a given spectrum. Although, it is worth noting that the number of peaks does not necessarily correlate positively with the level of information richness regarding the molecular structure. On one hand, carefully prepared datasets such as the CASMI16 contest can contain spectra with only a few peaks, which are sufficient to annotate the molec-

[9]For the specific choices of spectrum-level thresholds see section Formation of high-quality MSV^n datasets.

Algorithm 4: Estimate the absolute accuracy of a mass spectrometry instrument

Input: Mass spectrometry dataset given as a collection of m/z values $\{\mathbf{m}^{(t)}\}_{t=1}^{r}$ and intensities $\{\mathbf{i}^{(t)}\}_{t=1}^{r}$ corresponding to individual MS^1 spectra indexed by retention order $t \in \{1, \ldots, r\}$.

Output: Estimated absolute accuracy of the MS instrument.

Function BUILDXIC(t, i, ϵ, ξ)**:**
- $X \leftarrow \{\}$
- $i^* \leftarrow \mathbf{i}_i^{(t)}$
- **for** $t' \leftarrow i+1$ to r, $t' \leftarrow i-1$ to 1 **do**
  - **if** $(\exists i' \in \{1, \ldots, |\mathbf{m}^{(t')}|)(|\mathbf{m}_{i'}^{(t')} - \mathbf{m}_i^{(t)}| < \epsilon \wedge \mathbf{i}_{i'}^{(t')} < \xi i^*)$ **then**
    - $X \leftarrow X \cup (t', i')$
    - $i^* \leftarrow i'$
  - **else**
    - **break**
- **return** X

$I_1 \leftarrow \{\}$
for $t \leftarrow 1$ to r **do**
- **for** $i \leftarrow 1$ to $|\mathbf{m}^{(t)}|$ **do**
  - **if** $\mathbf{i}_i^{(t)} \in \arg\max_3 \mathbf{i}^{(t)}$ **and** $(\forall j \in I_1)(|\mathbf{m}_j^{(t)} - m| \geq 1.5)$ **then**
    - $I_1 \leftarrow I_1 \cup \{(t, i)\}$

$X_1 \leftarrow \{\}$
for (t, i) in I_1 **do**
- $X \leftarrow$ BUILDXIC$(t, i, 0.5, 0.1)$
- **if** $|X| \geq 5$ **then**
  - $X_1 \leftarrow X_1 \cup X$

$I_2 \leftarrow \{\}$
for X in X_1 **do**
- $m^* \leftarrow$Median$\{m \mid (m, i) \in X\}$
- $I \leftarrow \{(t, i) \in \{1, \ldots, r\} \times \{1, \ldots, |\mathbf{m}^{(t)}|\} \mid |\mathbf{m}_i^t - m^*| < 0.5\}$
- $I_2 \leftarrow I_2 \cup \{(t, i) \mid i = \max\{i' \mid (\exists t' \in \{1, \ldots, r\})((t', i') \in I)\}\}$

$X_2 \leftarrow \{\}$
for (t, i) in I_2 **do**
- $X \leftarrow$ BUILDXIC$(t, i, 0.01, 0.1)$
- **if** $|X| \geq 5$ **then**
  - $X_2 \leftarrow X_2 \cup X$

$S \leftarrow \{\text{STDEV}(\{\mathbf{m}_i^t \mid (t, i) \in X\}) \mid X \in X_2\}$
return MEDIAN(S)

ular structure (e.g., spectrum SM882102[10]). On the other hand, it is not uncommon for a typical mass spectrometry experiment in MassIVE to produce spectra with hundreds of low-accuracy peaks, among which only a few are not instrument noise. Consequently, it may become impossible or nearly impossible to assemble the complete molecular structures from such spectra. Nevertheless, given the experimental nature of MassIVE datasets, filtering millions of spectra based on the minimum number of high-relative-intensity peaks can effectively eliminate many information-poor spectra.

Determining whether a given peak is noise or not is a pivotal aspect of MS data processing. The most common approach utilized for this purpose is to classify a peak as noise if its intensity falls below a certain threshold and classify it as a true peak otherwise. This classification scheme is often used to eliminate noisy peaks, allowing the data processing workflow to operate solely on the remaining filtered spectrum. It is indeed often the case that such a procedure eliminates the majority of noisy peaks, however, it typically leads to the elimination of important low-intensity peaks. In other words, establishing a threshold that yields both high precision and high recall in the identification of noisy peaks is challenging, especially given the dispersion of intensity behaviors varying from instrument to instrument. As our objective is to collect a dataset of raw spectra, we do not eliminate any peaks. Instead, we eliminate entire spectra if they do not possess a sufficiently large intensity amplitude. We define the intensity amplitude as the ratio of the maximum intensity to the minimum intensity within a spectrum.

Storing resulting data files

Upon downloading each MassIVE file, we collect its data and metadata and compute its qualitative metrics. We then extract all this information into the compressed `.hdf5` format, which enables us to (i) effectively store the immense MassIVE in a compact, compressed format and (ii) organize the data into tensor-shaped structures suitable for deep learning. This process allows us to store 338,649 MassIVE files, which occupy a total of 18.35 TB, into a mere 2.9 TB (including a `.log` file associated with each `.hdf5` that contains statistics on the not collected complements of the data files). Table 4.1 shows the full specification of the `.hdf5` format.

[10] https://mona.fiehnlab.ucdavis.edu/spectra/display/SM883903

MS^n data	Data type
M/z values	float64
Intensities	float32
MS level	int8
RT	float32
Charge	int8
Polarity	int8
Precursor m/z	float32
Window lbound	float32
Window ubound	float32
CID energy	float32
Spectrum type	int8
Ion injection time	float32
Definition string	utf-8 str
Precursor id	int32

Metadata	Data type
File name	utf-8 str
Instrument name	utf-8 str
`MSLevelOrder`	utf-8 str
$\lvert X_1 \rvert$	int64
$\lvert X_2 \rvert$	int64
MEAN(S)	float64
MEDIAN(S)	float64

Precursor data	Data type
M/z values	float64
Intensities	float32
RT	float32
Ion injection time	float32
Scan id	int32

Table 4.1: Specification of our `.hdf5` format for MS^n spectra. "MS^n data" and "Precursor data" are `.hdf5` groups, whereas "Metadata" entities are `.hdf5` attributes. All tensors are 1-dimensional of the length m equal to the number of collected spectra. The only exception is "M/z values" and "Intensities" which are of the shape (m, n), where n is the maximum number of peaks across all spectra. Spectra having less than k peaks are padded with zeros. $|X_1|$, $|X_2|$, MEAN(S), and MEDIAN(S) refer to the Estimating MS data quality algorithm.

4.2.3 Formation of high-quality MSV^n datasets

Having more than 700 hundred million of MS^n spectra with rich metadata provides us with the flexibility to establish several subsets with varying degrees of MS data quality. Considering only the entire collected dataset or solely a refined high-quality subset would limit our investigation of self-supervised pre-training. In fact, it is not completely understood if large deep learning models benefit or suffer from noisy training examples [116]. Clearly, the lower the quality of data we tolerate, the substantially larger dataset the model has the freedom to operate on.

Yet, there are certain properties of mass spectra that we strictly disallow in the context of this work. Specifically, we discard spectra that exhibit features suggesting a peptide- or lipid-like nature of the underlying compounds, which are presumably dominant in MassIVE and therefore expected to be present in the GNPS part as well. These features include spectra with a charge greater than one (if known), precursor m/z greater than 1500 Da, or maximum m/z of a peak in a

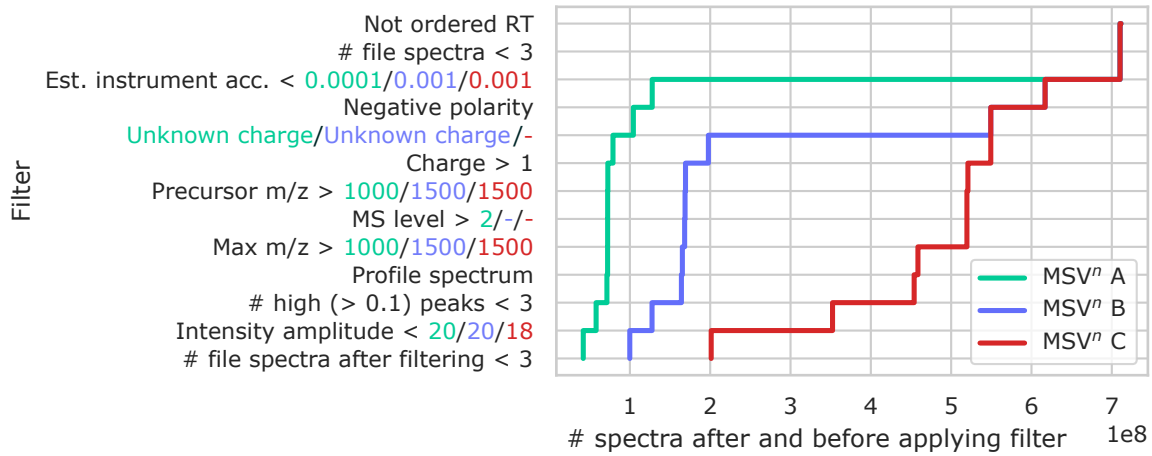


Figure 4.15: Construction of MSV^n dataset from 700 million MS^n spectra downloaded from the GNPS part of MassIVE. The rows are ordered consistently with our implemented filtering pipeline, illustrating the cumulative reduction of data volume. The final filtered datasets contain 41,951,922, 99,876,649, and 201,223,336 mass spectra, respectively. The figure reveals that MSV^n A is primarily filtered by selecting only high-accuracy mass spectra, while MSV^n B is reduced mainly by filtering out unknown charges. Lastly, MSV^n C is predominantly filtered based on the intensity properties of mass spectral peaks.

spectrum greater than 1500 Da. These features are direct indicators of the large size of molecules, which is beyond the scope of "small-molecule" metabolomics.

Additionally, we do not consider any spectra acquired in negative ionization mode. While the molecules ionizable in negative and positive ESI modes are roughly disjoint and provide orthogonal sources of information [117], we exclude negative ionization mode spectra in this work. This decision is based on the fact that negative mode is highly underrepresented in the collected data (Figure 4.11), and MS^n spectra often exhibit distinct fragmentation behaviors [118, 119].

Finally, we remove data files with untrustworthy content. Specifically, we discard data files that contain fewer than three spectra before or after filtering conditions are applied. Additionally, any files having spectra shuffled with respect to retention time or MS levels are eliminated. Spectra estimated to be in a profile format are also discarded. It is important to note that we have retained only those spectra that possess at least three high-intensity peaks. In this context, "high-intensity" refers to a peak with an intensity of at least 10% of the intensity of a base peak. As discussed in section Self-supervised pre-training on raw MS^n spectra, this criterion is crucial for defining self-supervision objectives over the final filtered datasets.

To establish variations with varying degrees of MS quality, we employ four major filtering conditions: estimated accuracy of the MS instrument, knowledge of precursor charge, intensity amplitude, and m/z thresholds. Specifically, we establish three datasets: MSV^n A, MSV^n B, and MSV^n C. We name the datasets such that the alphabetical order of A, B, C correlates positively with data size but negatively with data quality.

Figure 4.15 shows the individual filters applied to refine the dataset into three variants. The A variant is primarily filtered by the rigorous instrument accuracy threshold of 0.0001. Such filtering implicitly discards nearly all spectra containing m/z values beyond 1000 Da and with a charge greater than one. This confirms the rationality of the accuracy estimation since instrument accuracy is naturally an increasing function of a molecular mass. For subset B, we relax the filtering mainly by tolerating accuracy estimates below 0.001 and m/z values below 1500 Da. In this setup, the unknown charge of the precursor is the most severe filtering condition. We further relax this step to form the largest subset C. As it transpires, the strictest filtering requirements for MSV^n C are the two structural properties of spectral peaks: number of intense peaks and intensity amplitude.

Clustering spectra with locality-sensitive hashing

Each of the extracted MSV^n datasets is relatively clean with respect to our definition of spectral quality. However, the applied quality operates on a spectrum- or file-wise basis and does not provide information about the quality of the final collections of spectra combined from different sources. In particular, without further analysis, we cannot determine whether datasets contain duplicate spectra or the degree of such possible redundancy. Nevertheless, considering the dataset as high-quality from a machine learning standpoint necessitates a minimal degree of redundancy.

Given the dataset of size n, a naive approach attempting to at least eliminate identical or nearly identical spectra would require all $\Theta\left(\frac{n(n-1)}{2}\right)$ pairwise comparisons. Even for the smallest MSV^n A dataset, it leads to the intractable 10^{15} comparisons. To address the limitation, we employ the approximate method allowing us to efficiently cluster spectra in $\Theta(n)$ operations[11] (i.e. in linear time).

[11] Here, we use the Big Theta notation to ignore the initialization of the algorithm and multiplicative factors arising from several operations performed for each spectrum.

Moreover, the introduced approach approximates the cosine similarity of spectra which has been conventionally used to compare spectra for decades (see section Spectral similarity).

Our approach employs the algorithm of random projections belonging to the family of locality-sensitive hashes (LSH) [120]. Given a binned spectrum $\mathbf{s} \in \mathbb{R}^n$, its corresponding hash $h(\mathbf{s})$ is computed with a map $h : \mathbb{R}^n \to \{0,1\}^m$ defined as

$$h(\mathbf{s}) = [\mathbf{W}\mathbf{s} \geq 0], \text{ where } \mathbf{W} \in \mathbb{R}^{m,n}, \mathbf{W}_{ij} \sim \mathcal{N}(0,1),$$

where $[\cdot]$ denotes an element-wise Iverson bracket. Intuitively, for each $i \in \{1,\dots,m\}$, $\mathbf{W}_{i,:}\mathbf{s}$ is a dot product of $\mathbf{s}$ with a random n-dimensional hyperplane. Each of such m hyperplanes splits the n-dimensional space into two complementary subspaces, which means that the sign of each of the dot products $[\mathbf{W}_{i,:}\mathbf{s} \geq 0]$ determines to which of the two subspaces $\mathbf{s}$ belongs to. Since the hyperplanes pass the origin of the space, random projections are identical in all dot-products only if spectra are "similar" in a sense of cosine similarity. The larger m is, the more such "similarity" reflects the cosine similarity. More precisely, it can be shown [120] that for the fixed m it holds that

$$\mathbb{P}(h(\mathbf{s}_i) = h(\mathbf{s}_j)) = 1 - \arccos \underbrace{\left(\frac{\mathbf{s}_i^\top \mathbf{s}_j}{\|\mathbf{s}_i\| \|\mathbf{s}_j\|} \right)}_{\text{Cosine similarity}} \frac{1}{\pi},$$

where $\mathbb{P}$ is the probability over $\mathbf{s}_i, \mathbf{s}_j \in \mathbb{R}^n$. In other words, $\mathbb{P}(h(\mathbf{s}_i) = h(\mathbf{s}_j))$ is monotonically increasing as a function of cosine similarity. Indeed, arccos is monotonically decreasing with the codomain of $[0,\pi]$. Further normalizing it with π and subtracting from one gives a monotonically increasing function with the codomain of $[0,1]$.

As a consequence of such an approach, spectra can be clustered based on the equality of the corresponding hashes, which requires only a single traversal over the given collection of spectra. Therefore, we apply LSH for each of the MSV^n datasets. There are two parameters of the described method: the number of hyperplanes (i.e. m) and the width of a bin when converting spectra to the binned representations (determining n). While we observe almost no difference when considering different bin sizes in the range of $[0.05, 0.5]$, the number of hyperplanes significantly affects the resulting notion of redundancy within the spectra. Thus, we produce the condensed datasets with the computationally cheap yet relatively precise bin size of 0.5. With regard to the number of hyperplanes, we form

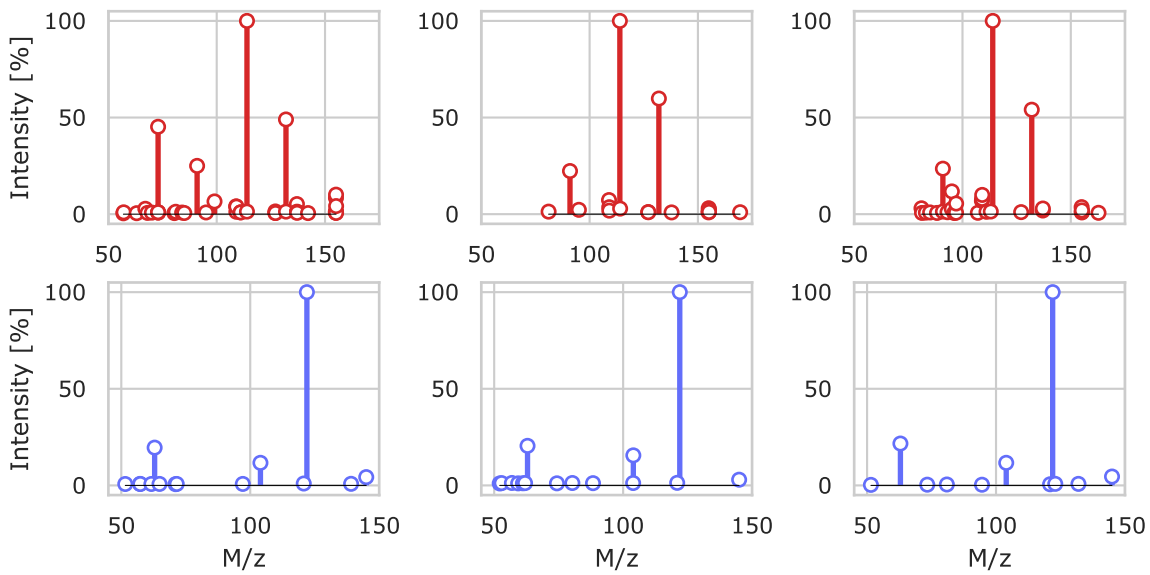


Figure 4.16: The impact of the number of hyperplanes on locality-sensitive hashing of mass spectra. The figure presents examples of MSV^n A spectra sharing identical locality-sensitive hashes (LSH) obtained using different numbers of random hyperplanes. **(Top)** LSHs obtained with 25 random hyperplanes. **(Bottom)** LSHs obtained with 1,000 random hyperplanes. Both configurations are robust to noise but are better suited for different purposes. While 1,000 hyperplanes cluster only "identical" spectra, 25 hyperplanes cluster all "similar" spectra. For instance, the top-left-most spectrum contains an intense peak at approximately 70 Da, which is not observed in the rest of the spectra in the top panel.

two extra subsets for each MSV^n. The first variant is designed to strictly eliminates all similar spectra by setting low $m = 25$. Since the number of distinct LSH increases exponentially with n, we set significantly larger $m = 1000$ for the second variant. As Figure 4.16 demonstrates in the example, such high m leads to a significantly moderate reduction of the spectra. Finally, by choosing only a single random spectrum from all spectra yielding identical LSHs we form 6 additional deduplicated subsets, which sizes are summarized in Table 4.2.

# LSH hyperplanes	MSV^n A	MSV^n B	MSV^n C
–	41,951,922	99,876,649	201,223,336
1000	13,705,892	33,541,098	79,342,367
25	2,611,997	5,685,087	11,777,221

Table 4.2: The sizes of the MSV^n variants after applying the filtering based on locality-sensitive hashes. Notably, each of the full MSV^n datasets contain approximately 95% of "similar" spectra (25 hyperplanes) and 70% of nearly "identical" spectra (1000 hyperplanes). Nevertheless, from the principle of LSH, "similar" and "identical" here are robust to low-intensity noise.

CHAPTER 5

Methods and experimental setup

In the previous chapters, we have shown that annotated tandem mass spectrometry data is sparse and is not particularly appropriate for pure data-driven approaches. We believe it explains the longstanding superiority of classic machine learning algorithms equipped with human priors (i.e. SIRIUS software) over the deep learning methods. Although neural networks are trivially limited by the capacity of spectral libraries, encoded human expertise mitigates the sparsity of available annotations. In this work, we, however, pursue an alternative approach. We aim to minimize the inclusion of human expertise but maximize the number of experimental training spectra regardless of their annotations. Therefore, we collect MSV^n datasets comprising hundreds of millions of diverse unannotated tandem mass spectra. In this chapter, we describe our deep learning method, which is able to comprehend MSV^n datasets and extract the knowledge from raw experimental data. We start the chapter by introducing the DREAMS (Deep Representations Empowering the Annotation of Mass Spectra) neural network architecture with a focus on its expressivity. Afterward, we describe the training objectives allowing the model to learn from the unlabeled MSV^n spectra in a self-supervised manner. Finally, we define validation metrics allowing us to assess the effectiveness of the introduced self-supervised training.

5.1 DreaMS architecture

As a deep neural network, our model consists of multiple layers stacked upon each other to form a high-order composition of parametrized functions. Given a mass spectrum, the model firstly encodes each spectral peak with a PEAKENCODER

layer and then processes the whole set of encoded peaks with TENCODER – a series of Transformer encoder blocks. The output of TENCODER is d-dimensional continuous representations of individual peaks. The embedding of a precursor peak is exactly the "representation" in the DREAMS acronym. During the training, the final shallow PEAKDECODER layer is used to adjust the representations for a loss function. While in the next section, we discuss the training objectives, in this section, we focus exclusively on the model architecture and its expressivity.

5.1.1 Input representation of a mass spectrum

Mass spectrum as a matrix

Each mass spectrum can be naturally represented as a matrix $\mathbf{S} \in \mathbb{R}^{2,n}$ constructed as

$$\mathbf{S} = \begin{bmatrix} \mathbf{m}_1 & \mathbf{m}_2 & \dots & \mathbf{m}_n \\ \mathbf{i}_1 & \mathbf{i}_2 & \dots & \mathbf{i}_n \end{bmatrix},$$

where each column corresponds to one of the n spectral peaks and is represented as a continuous vector $[\mathbf{m}_i, \mathbf{i}_i]^\top \in \mathbb{R}^2$. Although this representation fully describes the spectrum, it requires some enhancements. Firstly, it is beneficial to include additional information describing the precursor m/z. Often, MS^n spectra do not contain peaks representing the entire unfragmented molecules. However, the precursor mass is always available and significantly valuable for annotating a mass spectrum. Therefore, we prepend an additional column with the precursor mass m_0 and an artificial intensity of 1.1:

$$\mathbf{S} = \begin{bmatrix} \mathbf{m}_0 & \mathbf{m}_1 & \mathbf{m}_2 & \dots & \mathbf{m}_n \\ 1.1 & \mathbf{i}_1 & \mathbf{i}_2 & \dots & \mathbf{i}_n \end{bmatrix}.$$

The addition of a precursor peak also has a beneficial property from the standpoint of the neural network architecture. The Transformer encoder, TENCODER (discussed later), can be seen as a graph neural network, where the nodes of a graph are spectral peaks, and the "appropriate" edges are discovered by the model during training. Since one often wants to aggregate the learned information on nodes into a single representation of a graph, it is a common practice to introduce an additional "master node" that serves as an embedding of the entire graph [121, 122]. In our case, the peak $[\mathbf{m}_0, 1.1]^\top$ serves as such a master node.

Another important remark is that all spectra within the training dataset are standardized to have an identical length (i.e., the number of peaks n). In principle,

one could train a Transformer with variable lengths of spectra, as its architecture is agnostic to n. In other words, there is no Transformer parameter expressed in terms of n. However, it is a common practice in deep learning to pack input samples into batches and perform training iterations by averaging the loss over them. Since a batch is formally a tensor with the first dimension corresponding to the number of packed samples and the other dimensions representing individual samples, it would be impossible to accomplish with a varying number of peaks. Therefore, we employ a standardization procedure to retain exactly n peaks in each spectrum. Denoting the number of spectral peaks as k, we choose the n most-intense peaks if $k > n$ and pad the missing peaks with zero values $\mathbf{s}_i = [0, 0]^\top$ for all $i \in k, k+1, \ldots, n$ if $k < n$.

$$\mathbf{S} = \begin{bmatrix} \mathbf{m}_0 & \mathbf{m}_1 & \mathbf{m}_2 & \ldots & \mathbf{m}_k & \mathbf{0} & \ldots & \mathbf{0} \\ 1.1 & \mathbf{i}_1 & \mathbf{i}_2 & \ldots & \mathbf{i}_k & \mathbf{0} & \ldots & \mathbf{0} \end{bmatrix}.$$

PeakEncoder

As discussed in the subsequent section, each layer of TEncoder preserves the input shape of every peak. In other words, given the aforementioned spectrum representation, TEncoder is constrained to two-dimensional representations of peaks. The most direct method to elevate a peak's dimensionality to an arbitrary d is by employing a peak-wise feed-forward neural network, $\text{FFN}_{\text{peak}} : \mathbb{R}^2 \to \mathbb{R}^d$. However, we find that such a modification lacks sufficient sensitivity to high-resolution m/z values (Figure 5.1), which is an issue considering that the first decimals of masses are crucial for elucidating mass spectra [123]. To tackle the problem, we utilize the approach of Fourier Features from computer vision [124], which allows the model to better operate on high frequencies.

Specifically, we map each m/z ratio $\mathbf{m}_i$ to a frequency domain with a function $\text{FourierFeatures} : \mathbb{R} \to [0, 1]^{2t}$ of pre-defined sine and cosine functions:

$$\text{FourierFeatures}(m)_t = \sin(2\pi b_t m),$$
$$\text{FourierFeatures}(m)_{t+1} = \cos(2\pi b_t m).$$

Each b_t here defines a frequency. While the authors of Fourier features randomly set the frequencies without observing significant differences for computer vision tasks, we notice improvements when incorporating a modest chemical prior specific to the mass spectrometry domain. From the construction of each MSV^n

dataset, we know the lower bound m_{min} and the upper bound m_{max} for m/z values present in the datasets. While m_{max} is determined by the maximum mass threshold, we set m_{min} equal to the instrument accuracy estimate. To obtain the frequencies, we uniquely associate each b_t with a single element from the following set:

$$\left\{ \frac{1}{km_{\text{min}}} \;\middle|\; k \in \mathbb{N} \wedge m_{\text{min}} \leq km_{\text{min}} \leq 1 \right\} \cup \left\{ \frac{1}{k} \;\middle|\; k \in \mathbb{N} \wedge 1 \leq k \leq m_{\text{max}} \right\}.$$

Intuitively, the first term ensures the encoding of high-resolution decimals, while the second term guarantees the encoding of an integer part of a mass. Decimals are encoded with frequencies given as every second multiplier of m_{min} up to 1, and integers are encoded with natural frequencies up to m_{max}. For instance, in the MSV^n A dataset, where $m_{\text{min}} = 10^{-4}$ and $m_{\text{max}} = 1000$, such a schema yields 5000 high frequencies and 1000 low ones. Consequently, we apply a feed-forward network $\text{FFN}_F : \mathbb{R}^{2t} \to \mathbb{R}^{d_m}$ with l_F layers, which maps the frequencies to d_m-dimensional representations, internally learning their linear combinations.

Our frequency definition is inspired by the fact that the space of possible m/z values is not continuous, but rather determined by the intricate space of feasible elemental compositions of molecules [125]. As such, treating each m/z as an individual real value limits a neural network's performance. We posit that introducing a feed-forward layer operating in the frequency domain may enable the entire model to learn the structure of the space of masses.

It is worth noting that the Fourier Features method has previously been explored in the context of mass spectrometry under the name of sinusoidal embeddings [58, 59]. However, the authors employed a distinct strategy for defining frequencies and did not incorporate additional feed-forward layers. Examining the strategies in the auxiliary experiment evaluating their sensitivity to precise m/z ratios of Cl isotopes, we observe the superiority of our approach (Figure 5.1).

For the input intensities, we apply a separate feed-forward network $\text{FFN}_P : \mathbb{R}^2 \to \mathbb{R}^{d_p}$ of l_P layers operating of both m/z and intensity values. We combine the outputs of FFN_P and FFN_F by concatenation yielding the final composition of $\text{PEAKENCODER} : \mathbb{R}^2 \to \mathbb{R}^{d_m+d_p}$:

$$\text{PEAKENCODER}(m, i) = \text{FFN}_F(\text{FOURIERFEATURES}(m)) \parallel \text{FFN}_P(m, i),$$

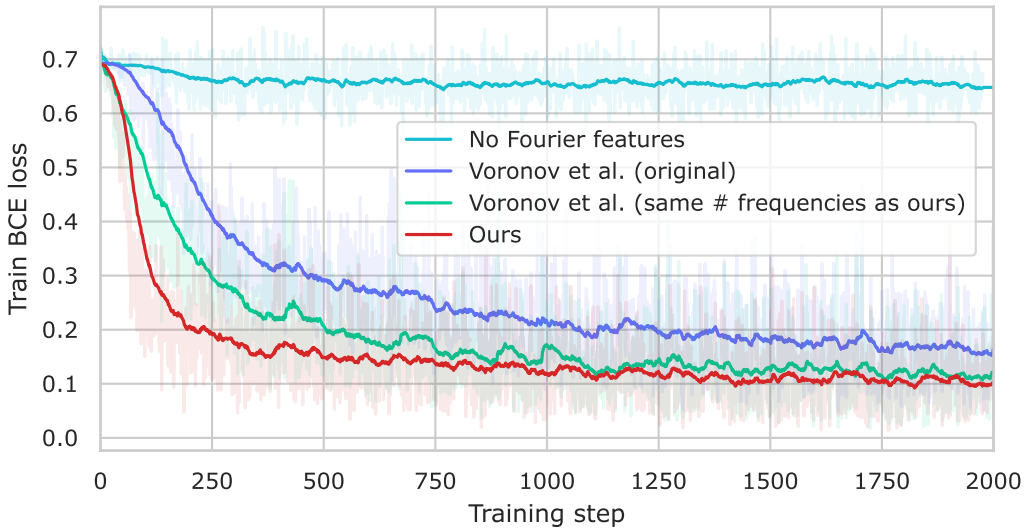


Figure 5.1: Supporting experiment motivating the use of Fourier features. The curves depict the training loss over training steps for binary classification in identifying the presence of Cl within a subset of NIST20. For the positive class, we select all NIST20 samples where precursors contain chlorine and their spectra exhibit mass differences of 1.997 Da, characteristic of the isotopic difference between ^{37}Cl and ^{35}Cl. The negative class consists of randomly sampled spectra that neither reflect chlorinated precursors nor contain the mass difference of 1.9970 Da. In essence, the binary classification is perfectly solved if the model can accurately determine the presence of the 1.9970 Da mass difference up to the fourth decimal place. We train the shallow DEEPSETS architecture without Fourier features, with the Fourier features strategy proposed by Vornov et al. [58], and with our strategy by setting the minimum wavelength to 10^{-4}. The DEEPSETS model does not converge without the features, as it is insensitive to high frequencies. In contrast, the model converges with Fourier features and performs optimally using our strategy. Notably, none of the setups yield a perfect predictor, which we designate as a high-priority area for future research.

making the final representation of a spectrum $\mathbf{S} \in \mathbb{R}^{d_m+d_p,n}$ look as following:

$$\mathbf{S} = \begin{bmatrix} | & | & & | & | & | \\ \textsc{FourierF}(\mathbf{s}_0) & \textsc{FourierF}(\mathbf{s}_1) & \dots & \textsc{FourierF}(\mathbf{s}_k) & \mathbf{0} & \dots & \mathbf{0} \\ | & | & & | & | & | \end{bmatrix}$$

5.1.2 Transformer encoder backbone

The entire Transformer encoder [38, 126] TENCODER takes a matrix $\mathbf{S}$ and subsequently updates its elements by applying l stacked TENCODERLAYER blocks. Thus, Transformer encoder is a map $\textsc{Encoder} : \mathbb{R}^{d,n} \to \mathbb{R}^{d,n}$ constructed as the composition $\textsc{Encoder}(\mathbf{S}) = \textsc{EncoderLayer}_l \circ \cdots \circ \textsc{EncoderLayer}_1(\mathbf{S})$. Importantly, each TENCODERLAYER has an identical domain and range, which allows for the composition of arbitrary depth. We will further denote the intermediate

output of ith encoder layer and input for the $i+1$th layer as $\mathbf{S}_i$, while still using $\mathbf{S}$ for describing an input to arbitrary encoder layer.

Each Transformer encoder layer consists of three major components: MUTLIHEADATTENTION, FFN, LAYERNORM. Among them MUTLIHEADATTENTION is the most crucial one. While the rest of the Transformer components operate peak-wise, MUTLIHEADATTENTION is the only layer interchanging the information between peaks[12].

MUTLIHEADATTENTION

It starts with projecting $\mathbf{S}$ with three parametrized linear maps $\mathbf{W}_Q,\ \mathbf{W}_K,\ \mathbf{W}_V \in \mathbb{R}^{d,d}$ as

$$\mathbf{Q} = \mathbf{W}_Q\mathbf{S},\ \mathbf{K} = \mathbf{W}_K\mathbf{S},\ \mathbf{V} = \mathbf{W}_V\mathbf{S},$$

which are referred to as queries, keys, and values respectively. Then single-head attention is computed as

$$\text{ATTENTION}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathbf{V}\,\text{softmax}\left(\frac{\mathbf{Q}^\top\mathbf{K}}{\sqrt{d}}\right),$$

where softmax : $\mathbb{R}^{n,n} \to (0,1)^{n,n}$ is defined as $\text{softmax}(\mathbf{X})_{i,j} = e^{\mathbf{S}_{i,j}}/\sum_{k=1}^{n} e^{\mathbf{X}_{k,j}}$. Let us firstly decompose the definition of the attention mechanism supposing that there were no $\mathbf{W}_Q,\ \mathbf{W}_K,\ \mathbf{W}_V$ applied and $\mathbf{Q} = \mathbf{K} = \mathbf{V} = \mathbf{S}$, which allows interpreting each of the matrices as the original spectrum. Matrix multiplication $\mathbf{Q}\mathbf{K}^\top$ computes dot products between all pairs of spectral peaks. Division by $\sqrt{d}$ is a technical detail that scales down the result of multiplication improving the numerical stability of the training procedure. softmax function exponentially normalizes each row of the $\mathbf{Q}^\top\mathbf{K}$ matrix such that it sums up to one. As a result, each column can be interpreted as a distinct probability distribution over the peaks (rows) assigned to each of the peaks (columns). Semantics underlying the distributions are encoded in $\mathbf{Q}^\top\mathbf{K}$ dot products and are learned by the model during the training. It means that the notion of similarity between each pair of peaks is determined by all the preceding parametrized transformations. Intuitively, such an attention block is capable of capturing, for example, certain characteristic m/z differences or encoding the molecular subfragment relation manifested in fragmentation trees. We further refer to the matrix of distributions as an "attention matrix" or $\mathbf{A} = \mathbf{Q}^\top\mathbf{K} \in \mathbb{R}^{n,n}$.

[12] Pedantically, here the attention mechanism should be referred to as self-attention, since peaks only "attend" to the other peaks within the same spectrum.

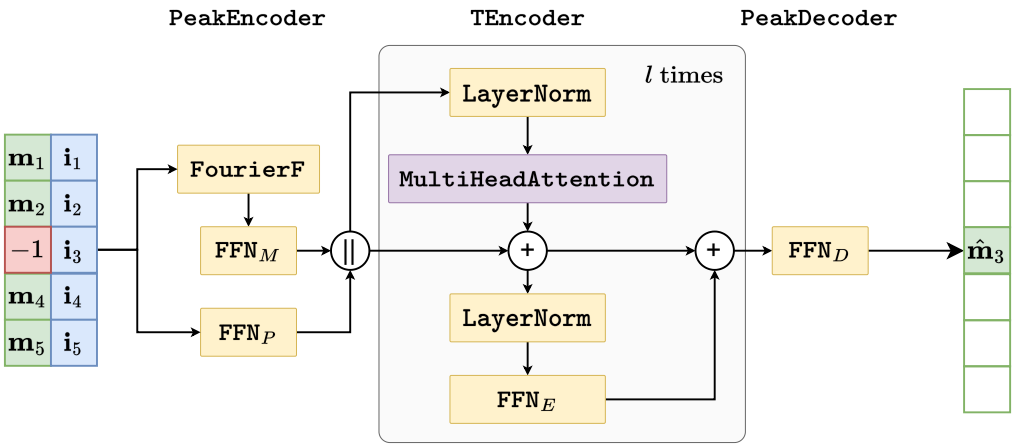


Figure 5.2: DreaMS neural network architecture. Yellow layers operate peak-wise, whereas a purple block operates spectrum-wise. The input and output illustrate the self-supervision training objective, where an input m/z value is masked with -1 and the network learns to predict its one-hot representation. The training objectives are introduced in the following section.

Finally, the representation of each peak is updated as the weighted average of original values $\mathbf{V}$ with respect to the corresponding attention distribution, which is exactly the $\mathbf{VA}$ multiplication. In such a way the representation of each peak aggregates the information from all other peaks. Important to mention, that the attention mechanism is prevented from attending to/from padded zero-valued peaks. This is achieved by substituting all the elements of $\mathbf{Q}^\top\mathbf{K}$ at positions i, j, such that $i > k$ or $j > k$, with a large negative value such as -10^9. Since, $\lim_{x\to-\infty} e(x) = 0$, softmax transforms the elements of $\mathbf{A}$ at the corresponding positions to close-to-zero values.

Linear projections $\mathbf{W}_Q$, $\mathbf{W}_K$, $\mathbf{W}_V$ increase the capacity of the attention layer by allowing it to operate on the same peaks from different spaces. In particular, since $\mathbf{W}_Q \neq \mathbf{W}_K$, the attention mechanism is non-commutative, whereas sole dot products would be. As will become clear through the following paragraphs, despite peaks essentially "attend" to each other with a simple dot-product, such a mechanism has a rich expressive power due to the heavy parametrization and stacked nature of the preceding layers.

In order not to limit the Transformer architecture to a single attention mechanism per TEncoderLayer and, therefore, to allow the model to discover the higher diversity of relations between input elements, Attention block is typically

generalized to MULTIHEADATTENTION.

$$\text{MULTIHEADATTENTION}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathbf{W}_O \Big\|_{i=1}^{h} \text{ATTENTION}_i(\mathbf{Q}, \mathbf{K}, \mathbf{V}),$$

where $\mathbf{W}_O \in \mathbb{R}^{hd,n}$ is an additional parameter. Essentially, MULTIHEADATTENTION concatenates the outputs of h distinct attention layers yielding a hd-dimensional representation of each spectral peak. Afterwards, $\mathbf{W}_O$ projects the hd-dimensional peaks back to a d-dimensional form.

Now suppose that the TENCODERLAYER consists only of MULTIHEADATTENTION blocks, and, therefore, TENCODER is a composition of lone MULTIHEADATTENTION layers. In such a scenario, the entire model would be parametrized solely by linear projections $\mathbf{W}_Q$, $\mathbf{W}_K$, $\mathbf{W}_V$, $\mathbf{W}_O$ of each ATTENTION block. Intuitively, the model would not possess enough degrees of freedom to prepare the output of one MULTIHEADATTENTION as an input to another. For this purpose, MULTIHEADATTENTION are altered (w.r.t. the sequence of layers) with additional shallow feed-forward neural networks operating peak-wise.

$\mathbf{FFN}_E$

Each such feed-forward network FFN_E is defined as

$$\text{FFN}_E(\mathbf{s}) = \mathbf{W}_2 \text{GELU}(\mathbf{W}_1 \mathbf{s} + \mathbf{b}_1) + \mathbf{b}_2,$$

where $\mathbf{W}_1 \in \mathbb{R}^{4d,d}, \mathbf{W}_2 \in \mathbb{R}^{d,4d}, \mathbf{b}_1 \in \mathbb{R}^{4d}, \mathbf{b}_2 \in \mathbb{R}^{d}$ are parameters, and GELU [127] is an activation function defined as $\text{GELU}(x) = x\mathbb{P}_{X \sim \mathcal{N}(0,1)}(X < x)$, which is applied element-wise. Since the probability term is exactly the standard Gaussian cumulative distribution function, GELU can be seen as the smooth version of ReLU given that $\text{ReLU}(x) = \max\{0, x\} = x\mathbb{1}_{x>0}$ with $\mathbb{1}$ being an indicator function. Intuitively, such a two-layer feed-forward network adjusts d-dimensional peak representations in $4d$-dimensional space for the subsequent attention layer. Thus, MULTIHEADATTENTION and FFN_E constitute two core TENCODERLAYER components complementing each other. MULTIHEADATTENTION mixes the propagated information within peaks, whereas FFN_E operates purely on the information of standalone peaks.

Composition of blocks into TENCODERLAYER

To better propagate the information along stacked TENCODERLAYER's, TENCODER architecture employs skip connections [128] for both MULTIHEADATTENTION and

FFN_E blocks. Denoting either of layers as f and the corresponding input as $\mathbf{X}$, the skip connection adjusts the output of the block to be $\mathbf{X} + f(\mathbf{X})$ instead of the regular $f(\mathbf{X})$.

Furthermore, after each skip connection, the original Transformer architecture employs LayerNorm blocks [129]. They explicitly control the mean and the variance of individual activations enabling more stable and robust training. LayerNorm transforms the representation of each peak $\mathbf{s}$ according to

$$\text{LayerNorm}(\mathbf{s}) = \frac{\mathbf{s} - \mu(\mathbf{s})}{\sqrt{\sigma(\mathbf{s}) + \epsilon}} * \boldsymbol{\gamma} + \boldsymbol{\beta},$$

where μ and σ are mean and variance, $\boldsymbol{\gamma}, \boldsymbol{\beta} \in \mathbb{R}^d$ are parameters, $*$ denotes an element-wise product, $\mu(\mathbf{s})$ is substracted from each dimension of $\mathbf{s}$, and ϵ is a close-to-zero positive constant. Since the combination of skip connection and the subsequent layer normalization may destabilize the training [130], we follow Wang et al. [131] and apply LayerNorm blocks prior to the skip connections.

In summary, each TEncoderLayer comprises the following composition:

$$\text{TEncoderLayer}(\mathbf{S}) = \mathbf{S} + \text{FFN}_E(\text{MultiHeadAttention}(\text{LayerNorm}(\mathbf{S}))).$$

5.1.3 Decoding output representations

The primary focus of our study is the intermediate embeddings learned by the model during training. However, for the training procedure, it is necessary to prepare the output representations in an appropriate shape. To achieve this, we employ a PeakDecoder, which consists of a single shallow feed-forward network with one hidden layer of size d: $\text{FFN}_D : \mathbb{R}^d \to \mathbb{R}^z$. The value of z is a task-specific output shape. For instance, if our goal is to predict the number of precursor carbon atoms from the spectrum, we set $z = 1$. Alternatively, if we aim to predict a 2048-bit molecular fingerprint, we set $z = 2048$.

5.2 Self-supervised pre-training on raw MS^n spectra

Suppose the following problem. Given a sole theoretical MS^n spectrum (i.e. an arbitrary set of m/z ratios and the corresponding signal intensities), can one conclude if the spectrum can be observed experimentally? Alternatively, given an experimentally measured mass spectrum with some peaks being removed, is it

possible to fill them based on the context of the other peaks? Imagine a deep learning predictor that can perfectly solve at least one of such problems. Is it possible for the predictor to achieve such high performance without internally associating individual peaks with molecular fragments and without operating on the level of chemical structures?

Our assertion is that the answer is negative. In other words, we claim that asymptotically the knowledge of chemical principles and molecular structures in a deep learning model solving synthetic problems such as the identification of corrupted spectra or prediction of masked signals is equivalent to the knowledge embedded in a deep learning model directly solving the inverse annotation. However, while the straightforward inverse annotation is designated to operate solely on labeled spectral libraries, the model trained on synthetic tasks can operate on billions of artificially constructed training examples. Indeed, one can generate hundreds of "labels" for each of the hundreds of millions of raw mass spectra by simply modifying the peak values.

It should be understood that the subject of interest is not the problem itself but the knowledge encoded in the model while searching for the corresponding solution. Therefore, the synthetic problem has to be challenging for the neural network but still solvable. For example, determining the number of spectral peaks is obviously a too simple problem and does not require an understanding of the desired structural properties of the underlying molecules. On the other hand, masking 90% of spectral peaks and training the model to restore them is an unreasonably hard task, as the model lacks valuable input.

5.2.1 Definition of training objectives

The formulation of annotation-free objectives and the corresponding process of learning can be formalized as self-supervised training over the constructed MSV^n datasets. For the training, we employ the DREAMS neural network, where the desired chemical knowledge can be formalized as the outputs $\mathbf{S}_l$ and derived by attention layers. In particular, we focus on three kinds of objectives in the scope of this work.

Prediction of masked peaks as regression

First of all, we consider the simplest approach, where peaks in the mass spectrum are masked, and the model is trained to restore them. More precisely, given an original matrix $\mathbf{S}$, we construct its masked representation $\mathbf{S}^*$ by modifying some of the columns to be $\mathbf{S}^*_{:i} = [-1, -1]^\top$, where $i \in I \subset \{1, \ldots, n\}$. We mask only a fraction of random high-intensity peaks, which means that the indices I are sampled from $\{j \in \{1, \ldots, n\} \mid \mathbf{i}_j \geq 0.1\}$. Furthermore, we make the sampling deterministic with respect to the spectrum. It implies that in each training epoch, we mask identical peaks.

In the context of this work, we limit the size of I to at most two peaks. Such a limitation is caused by our approach to the formation of MSV^n datasets. Since we filter out all spectra having less than three intense signals, it is not guaranteed that for each MSV^n spectrum, there exists I with more than two elements. Similarly, we never mask the precursor peak $\mathbf{s}_0$, since it is always available and therefore should always serve the purpose of the master node, as discussed previously. As an implication of the moderate masking of peaks, we do not experiment with more advanced masking strategies such as replacing the peak with minus ones, original values, and random values with 0.8, 0.1, and 0.1 respective probabilities [39]. Neither do we experiment with the non-deterministic sampling of masking positions.

Having matrices $\mathbf{S}^*$ determined, we accordingly set the last layer of the PEAKDECODER to produce 2-dimensional outputs. Then, the prediction of the masked peak can be formulated as the following regression problem:

$$\mathcal{L}(\hat{\mathbf{S}}, \mathbf{S}) = \sum_{i \in I} \|\hat{\mathbf{S}}_{:i} - \mathbf{S}_{:i}\|_2^2,$$

where $\hat{\mathbf{S}} = \text{DREAMS}(\mathbf{S}^*)$.

Prediction of masked peaks as classification

Even though such formulation is the most straightforward, it has a certain severe limitation. In particular, it does not allow the model to capture the ambiguity of the prediction [44]. For instance, a spectrum acquired in a low CID energy regime can naturally lack a peak that corresponds to a rare fragment. Yet, it may happen that the model learned the presence of such a peak from the other training examples. From the mass spectrometry standpoint prediction of both peaks

can be considered as correct. However, the regressive nature of the loss function would force the model to predict the mean of both peaks instead of the distribution across possible values. To address this issue we treat each output m/z ratio as a discretized value. More precisely, we split the m/z range $[0, m_{\text{max}}]$, given by the MSV^n thresholds, into bins of 0.5 widths forming $c = \frac{m_{\text{max}}}{0.5}$ bins in total. Then we set the shapes of the PEAKDECODER parameters to yield c-dimensioanl values and extend it with softmax such that it transforms output values to a categorical distribution. Then the masking modeling can be expressed as a classification problem with the cross-entropy loss function

$$\mathcal{L}(\hat{\mathbf{S}}, \mathbf{S}) = -\sum_{i \in I} \sum_{j \in \{0,\dots,c-1\}} \mathbb{1}_j(\mathbf{S}_{1i}) \log \hat{\mathbf{S}}_{ji}$$

where $\mathbb{1}_j(\mathbf{S}_{1i})$ equals to 1 only if $0.5j < \mathbf{S}_{1i} \leq 0.5(j{+}1)$, and equals to 0 otherwise. To extend the problem to the prediction of intensities but not only m/z values, we similarly bin intensity range of [0, 1] to ten bins of 0.1 widths and define an identical loss function but with a separate PEAKDECODER. Then the prediction of a whole peak can be defined by summing two loss functions. In our experiments, we weight the intensity loss with a 0.5 multiplicative factor.

Prediction of shuffled intensities

The general critique of self-supervised training by masking is that such methods are forced to operate on corrupted data points [132, 133]. Indeed, regardless of the training setup, one wants to employ the model on real mass spectrometry data but not on the partially masked peaks. However, both loss functions introduced forehand are computed from the representations of masked peaks. As a result of backpropagation, the "chemical knowledge" may be encapsulated exclusively in the masked tokens which are absent during the inference on complete spectra. To overcome the issue, we experiment with an alternative self-supervision objective. Given a mass spectrum, we randomly shuffle the intensities of peaks I with the other random peaks, while retaining the original m/z values unchanged. Formally, we modify the row of intensities $\mathbf{i} = \mathbf{S}_{2:}$ as

$$\mathbf{i}^* = \mathbf{P}\mathbf{i},$$

where with 50% probability $\mathbf{P} \in \{0, 1\}^{n,n}$ is an identity map and with 50% probability $\mathbf{P}$ is a random permutation matrix satisfying

$$(\forall i \neq j \in \{1, \dots, n\})(\mathcal{E}^n \ni \mathbf{P}_{i:} \neq \mathbf{P}_{j:} \in \mathcal{E}^n) \wedge (\forall i \in I)(\mathbf{P}_{i:} \neq \mathbf{e}_i).$$

$\mathcal{E}^n = \{\mathbf{e}_i, \ldots, \mathbf{e}_n\}$ denotes the standard base of $\mathbb{R}^n$. Then the loss function is defined as a binary cross-entropy

$$\mathcal{L}(\hat{\mathbf{S}}, \mathbf{S}) = -\sum_{j \in \{0,1\}} \mathbb{1}_j \log \hat{\mathbf{S}}_{11}$$

with the indicator $\mathbb{1}_j = j$ only if the intensities were not shuffled (i.e. $(\forall i \in \{1, \ldots, n\})(\mathbf{P}_i = \mathbf{e}_i)$), and 0 otherwise. The shapes of the parameters constituting PEAKDECODER are accordingly adjusted to output 1-dimensional values. Notably, the prediction is inferred solely from the precursor peak (i.e. $\hat{\mathbf{S}}_{11}$).

5.2.2 Pre-training validation

Consider, for instance, a set of one hundred distinct models trained through a self-supervised technique. Which one has gained greater knowledge about molecular structures underlying mass spectra? Although the standard deep learning approach entails monitoring the validation loss on a held-out portion of the training data, this method is inapplicable in a self-supervised context. Indeed, as previously mentioned, the self-supervision process must be sufficiently challenging for the model, meaning that low training and validation losses are not necessarily demanded.

Conversely, one could directly fine-tune each pre-trained model by employing labeled spectral libraries, concentrating on problems that necessitate an understanding of chemical structures. However, this method is computationally demanding and lacks flexibility. In particular, it limits the understanding of the dynamics of self-supervision across training iterations.

Validation metrics

To continually and efficiently monitor the performance of pre-training, we employ validation metrics based on extracted embeddings. Specifically, after each training epoch, we calculate the representations $\mathbf{S}_l$ for a subset of NIST20 annotated spectra. We then compute certain metrics derived from pairwise distances between these representations. This approach enables us to evaluate the mutual dependencies between embedded spectra and assess the structure of the learned embedding space.

Our first evaluation metric is the Pearson correlation between the pairwise distances of spectral embeddings and the Tanimoto distances between the associated

Algorithm 5: Contrastive validation

Data: k classes, each given by n d-dimensional continuous vectors: $\{\mathbf{x}_j^{(c)}\}_{j=1}^{n} \subset \mathbb{R}^d$ for all $c \in \{1, \ldots, k\}$. Element-wise distance measure on the classes: $d : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$.

Result: Average intra-inter distance between classes.

$p \leftarrow \frac{n(n-1)}{2}$
$m \leftarrow 0$
for $k_1 \in \{1, \ldots, k\}$ **do**
  $m_1 \leftarrow \sum_{i=1}^{n} \sum_{j=1}^{i-1} d(\mathbf{x}_i^{(k_1)}, \mathbf{x}_j^{(k_1)})$
  $m_2 \leftarrow 0$
  for $k_2 \in \{1, \ldots, k\} \setminus \{k_1\}$ **do**
    $I \leftarrow$ sample $\lfloor \frac{p}{k-1} \rfloor$ pairs from $\{1, \ldots, n\}^2$
    $m_2 \leftarrow m_2 + \sum_{i,j \in I} d(\mathbf{x}_i^{(k_1)}, \mathbf{x}_j^{(k_2)})$
  $m \leftarrow m + \frac{m_1 - m_2}{p}$
return $\frac{m}{k}$

molecules. As a distance on the embeddings, we experiment with cosine similarity and the standard Euclidean distance. For the Tanimoto distance, we employ standard 2048-bit circular fingerprints. The underlying principle is that, throughout the training iterations, effective pre-training should progressively maximize the correlation. We perform the validation on random 10,000 pairs of samples from NIST20 maximizing the entropy of the pairwise Tanimoto similarity distribution.

We also utilize a second group of metrics, which we term "contrastive validation". For this purpose, we create subsets of NIST20, with each one containing equally-sized groups sharing specific discrete-valued properties. For example, one subset is organized into six groups containing spectra of molecules with 5, 10, 15, 20, 25, and 30 carbon atoms. Another subset is divided into five groups based on commonly observed adducts. In total, we create 11 datasets: seven are separated by chemical element counts, while the remaining four are grouped by MS instrument models, molecular structures, CID energies, and adducts. Contrastive validation involves calculating the average embedding distances within groups (intra-distances) and between groups (inter-distances). Subtracting the mean intra-distance from the mean inter-distance (intra-inter-distance) yields a value that reflects the capacity of the learned embeddings. In other words, a lower intra-inter distance indicates a better model differentiation of spectra with distinct contrastive labels.

Ranking induced by validation metrics

The objective of validation metrics is to identify the highest-performing model. Consequently, it is necessary to consolidate the 12 aforementioned metrics into a single value that can effectively represent the pre-training performance measure. This process can be divided into two phases: selecting the most effective epoch for each individual model and choosing the top-performing model from a pool of pre-trained models. For both phases, we consistently apply the same rank product approach [134].

With a pool of models and their corresponding metric values, we first assign a rank $r_i(m) \in \mathbb{N}$ to each mth model for each ith metric. The greater the rank value, the better the performance of the model metric-wise. Denoting the correlation validation rank as r_c and the contrastive validation scores as r_d for each contrastive metric $d \in D$, we obtain the unified rank for the model as

$$r(m) = \sqrt[|D|+4]{r_c(m)^4 \prod_{d \in D} r_d(m)}.$$

In addition to the standard rank product procedure, we apply a heuristic weighting factor of 4 to the correlation validation, as we consider it more significant than any individual contrastive metric. We employ exponential weighting in a product, as it is equivalent to the multiplicative weighting of a sum in a sense of arithmetic hyperoperations.

5.3 Configuration of training and hyperparameters

Self-supervised pre-training

In our experiments, we investigate different combinations of model architectures, self-supervision objectives, and training datasets. In particular, we experiment with six MSV^n datasets which comprise all LSH variants of MSV^n A and MSV^n B. In the scope of this work, we omit the MSV^n C subset due to its large size. We investigate all the introduced pre-training objectives.

For the model architecture, we consider four configurations of hyperparameters varying depending on three basic criteria: the number of TENCODER layers l, the number of attention heads in each layer h, and the hidden dimension $d = d_M + d_P$. Three of the configurations are given by the extreme values of hy-

perparameters bounded by the computational resources (Table 5.1). The fourth configuration is roughly an average of the three extreme setups. We train the models in a single-node multi-GPU setup on the Karolina supercomputer, where each node is equipped with 8 NVIDIA A100 GPUs. Since MSV^n datasets are trimmed to low $n = 60$ peaks, we are able to achieve reasonably complete training experiments on the MSV^n A and MSV^n B datasets by limiting the training time of each model to 24 hours.

To solve the training optimization problem we use the Adam optimizer [135] with standard parameters. We experiment with the training in "low" (64) and "large" (512) batch size modes, as well as, with three learning rates: $5 \cdot 10^{-5}$, $3 \cdot 10^{-4}$, and $6 \cdot 10^{-4}$. For the latter two, we use the inverse square root schedule following Vaswani et al. [38]. We regularize the model by applying a 10% or 50% dropout [136] on the outputs of MULTIHEADATTENTION, FFN_E, as well as on the intermediate outputs of PEAKENCODER and PEAKDECODER. For the 50% setup, we additionally experiment with 10^{-5} weight decay [137]. We estimate the optimal combination of the hyperparameters according to Table 5.1.

Hyperparameter	**Values**
Dataset	{MSV^n A, MSV^n B}
# peaks n	{60}
# masking peaks $\lvert I \rvert$	{1, 2}
DREAMS $l/h/d_M/d_P$	{12/4/320/40, 8/30/320/40, 5/8/730/38, 8/8/480/32}
Depths l_M/l_P	{5/3}
Learning rate/# warmup steps	$\{6 \cdot 10^{-4}/30000, 3 \cdot 10^{-4}/30000, 5 \cdot 10^{-5}/0\}$
Batch size	{64, 512}
Dropout/weight decay	$\{10\%/0, 50\%/0, 50\%/10^{-5}\}$

Table 5.1: Investigated configurations of hyperparameters. Parameters are grouped with slashes (/) to highlight that we do not experiment with any other combinations except for the ones given as elements of sets. Searching for the optimal model architecture we traverse roughly the entire Cartesian product given that the "DREAMS $l/h/d_M/d_P$" is fixed to the first option. For the rest of the combinations, we traverse the space more sparsely driven by educated guesses. Notice that the rest of the potential hyperparameters are fixed throughout the text and therefore not present in the table.

Supervised fine-tuning

In order to additionally evaluate the capabilities of the pre-trained model, we conduct an end-to-end fine-tuning (i.e. additional training) on a series of downstream

tasks using two annotated datasets. Specifically, we examine four tasks with labels derived from precursor structures. We fine-tune the model to predict (i) 2048-bit ECFP fingerprints, (ii) the number of oxygen atoms, (iii) the presence of nitrogen, and (iv) the quantitative estimate of drug-likeness (QED). Each task is designed to assess the model with a slightly different loss function: (i) average binary cross-entropy across individual bits, (ii) mean squared error, (iii) binary cross-entropy, and (iv) mean squared error constrained by a preceding sigmoid function. We split the NIST20 dataset using Murcko histograms, as previously described, and employ a 10-fold cross-validation of the GNPS dataset developed by Dührkop et al. as evaluation data for SIRIUS 4[13]. Crucially, we only select spectra that satisfy the MSV^n A filtering criteria (Figure 4.15) to mitigate the distribution shift between pre-training and fine-tuning phases. However, in the scope of this work, we do not experiment with the entire datasets. The resulting NIST20 subset comprises 259,677 training and 95,542 validation examples, whereas the GNPS subset contains 10 folds with an average of 386 samples each. This discrepancy in dataset sizes enables us to validate the pre-training under two distinct conditions.

For fine-tuning the pre-trained model, we make minor adjustments to the training setup. We substitute the PeakDecoder with a single linear projection to focus on the extraction of task-specific information directly from the TEncoder. We experiment only with end-to-end fine-tuning without freezing any pre-trained layers. To prevent the fine-tuning process from overwriting features acquired during self-supervision, we reduce the learning rate by a factor of six compared to the pre-training stage. Additionally, we disable dropout within the DreaMS layers and utilize a batch size of 64.

To compare the performance of DreaMS with that of another deep-learning baseline, we employ the DeepSets architecture:

$$\text{DeepSets}(\mathbf{S}) = \text{FFN}_\rho\left(\sum_{i=1}^{n} \text{FFN}_\phi(\mathbf{S}_{:i})\right),$$

where $\text{FFN}\phi : \mathbb{R}^2 \to \mathbb{R}^{768}$, $\text{FFN}\rho : \mathbb{R}^{768} \to \mathbb{R}^z$, and z represents the task-specific output dimensionality. Each of the networks contains three hidden layers of 768 neurons. Given that FFN_ρ operates on a peak-wise basis and the sum is associative, DeepSets consists of a series of permutation-equivariant layers followed

[13] https://bio.informatik.uni-jena.de/data/

by permutation-invariant aggregation. We stress that this property is the minimum requirement for a neural network architecture applied to mass spectra[14]. A standalone feed-forward network is not invariant to peak permutations and, consequently, cannot, for example, identify the same mass difference occurring between different peak positions.

[14]Note that this requirement is also satisfied by the DREAMS architecture.

CHAPTER 6

Results

In this chapter, we present our results on the self-supervised pre-training of the DREAMS neural network using MSV^n datasets. We start by discussing our key methodological findings related to the pre-training process and demonstrate the emergence of molecular features through self-supervision. Subsequently, we explore the rich space of DREAMS embeddings and examine the model's fine-tuning for downstream tasks.

6.1 Validation of self-supervised pre-training

6.1.1 Investigation of hyperparameters

In this section, we detail the insights gained from the exploration of the hyperparameter space in relation to our ranking approach. Based on our observations from approximately 100 training experiments, we can draw the following conclusions.

Excessiveness of high-accuracy spectra is an optimal setup for self-supervision

- Redundancy of similar spectra is advantageous, but only for high-accuracy spectra. For MSV^n A, we observe that pre-training performance increases proportionally with the number of LSH hyperplanes, achieving the best results in their absence. Conversely, we notice an opposite trend for MSV^n B datasets. While performance remains approximately equal for data filtered with 25 and 1000 hyperplanes, it deteriorates by up to 50% for the complete

MSV^n B dataset, as indicated by validation metrics.

- Fewer high m/z accuracy spectra outperform a larger number of low-accuracy spectra. Specifically, we consistently find that pre-training DREAMS on MSV^n A datasets yields superior results compared to MSV^n B datasets. However, given that we have not conducted experiments with MSV^n C or the entire 700-million MSV^n within the scope of this study, we cannot definitively conclude if our observation is scalable.

Masking m/z ratios as classification spawns the richest DREAMS representations

- Shuffling or masking intensities are overly simplistic SSL objectives. When pre-training DREAMS for identifying shuffled or masked intensities, we observe an increase in validation loss immediately after the first epoch. This pattern persists even for datasets containing redundancy of similar spectra, such as full MSV^n B. In cases where other objectives lead to a continuous decrease of validation loss on random splits, objectives not involving perturbations of m/z ratios are so simplistic that the model overfits within the first epoch.
- Masking m/z is preferable to masking both m/z and intensity. In line with the previous point, masking intensity in addition to masking m/z does not significantly impact the training process. In fact, it marginally worsens performance on validation metrics and, consequently, the derivation of molecular features.
- Masking as classification greatly outperforms masking as regression. We observe a notable difference when formulating m/z masking as a straightforward regression compared to classification over binned masses. Specifically, while regression on average attains a 0.1 correlation in validation, classification achieves a 0.3 correlation. This fact underlines the significance of capturing the distribution of potential peaks rather than assuming a single possible solution.
- Masking either one or two peaks yields similar results. However, masking two peaks leads to a more stable extraction of chemical knowledge through self-supervised learning. We find that when masking two peaks, contrastive validation metrics – such as the separation of spectra by adducts or by the number of sulfur elements – exhibit substantially lower deviation throughout

the training process.

Large Transformer dimensionality, large training batch size, and large scheduled learning rate are the most effective configurations

- The hidden dimension is the most crucial hyperparameter for the Transformer backbone. Upon examining various combinations of the model's primary hyperparameters, we find that the number of attention heads and the number of Transformer layers have no significant impact. However, we observe that increasing the hidden dimension of the Transformer generally leads to improved validation performance, although the effect is not consistently positive.
- Larger batch sizes of 512 samples outperform smaller batches of 64 examples on average. Utilizing larger batch sizes results in better validation metric values and makes the training approximately two times faster.
- A lower learning rate of $5 \cdot 10^{-5}$ typically leads to higher-than-average validation metrics, but larger learning rates of $3 \cdot 10^{-4}$ and $6 \cdot 10^{-4}$ combined with linear warmup yield the best overall experimental runs. Remarkably, the top-performing models are those that closely approach the peaks of the warmup schedule, rather than those that decay after the peak or maintain a consistently low learning rate.
- While DREAMS often benefits from regularization, the effects of varying dropout and weight decay values are not consistently positive and depend on the specific combination of other hyperparameters.

6.1.2 Self-supervision on mass spectra gradually derives molecular properties

In our exploration of training DREAMS for masked m/z classification with various hyperparameters, we find that most training experiments exhibit a similar pattern. Specifically, the training loss decreases substantially after the initial epochs, followed by a gradual decline. At least two out of the four groups of validation metrics (i.e. contrastive and correlation validations with cosine or Euclidean distances) improve concurrently with the loss. Furthermore, the contrastive metrics assessing the separation of spectra based on molecular properties consistently improve in tandem. These two persistent observations underscore the effectiveness

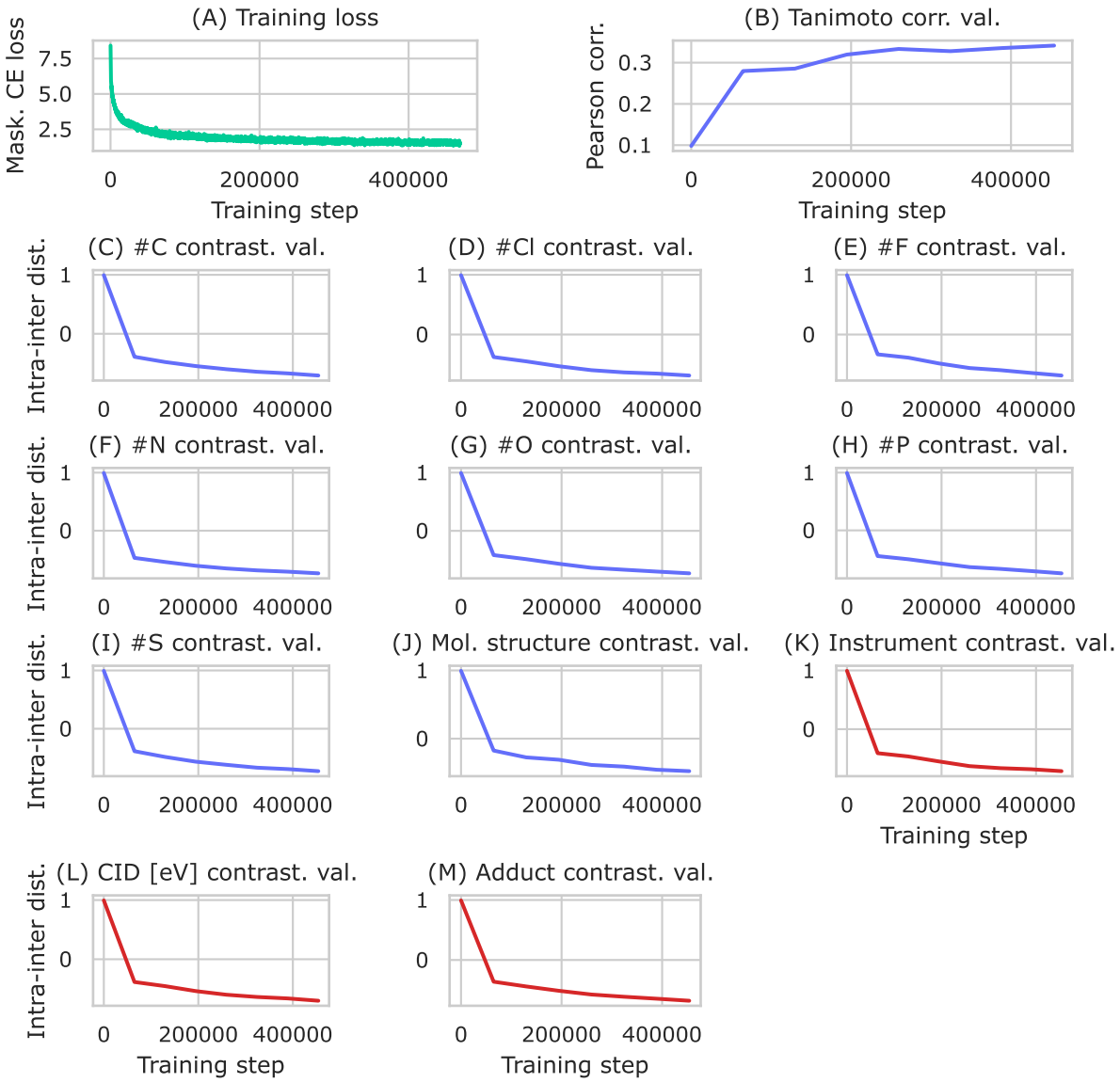


Figure 6.1: Emergence of molecular properties through self-supervised learning on mass spectra. The figure presents the training loss and cosine validation metrics for the DREAMS model, trained to predict masked m/z values in a classification setting. Plot **(B)** illustrates that, during self-supervision, DREAMS progressively learns to approximate the Tanimoto distances of precursor molecules solely from their spectra. The remaining blue curves **(C) - (J)** show similar behavior on contrastive validation metrics that evaluate the separation of embeddings based on discrete structural properties of molecules. Figures **(K) - (M)** reveal that the model concurrently learns to differentiate spectra according to mass spectrometry settings. Notably, all validation metrics improve in proportion to the training loss and consistently across all measures. This strongly suggests that the m/z masking training objective forces the model to operate in the space of derived molecular structures.

of self-supervised pre-training as a method for extracting molecular information exclusively from mass spectra. Figure 6.1 depicts a typical training experiment.

Intriguingly, the pairs of jointly improving metric groups may change within a single experiment. It is not uncommon for a model to exhibit consistent improvement by cosine distances and deterioration in Euclidean distances up to a certain training step, and start operating in the opposite "mode" afterward. However, such runs typically do not yield top performance according to our ranking procedure.

Remarkably, the best model, outperforming all others in the ranking across all experiments and training epochs, has been trained for only a single epoch over the full MSV^n A dataset. The model was configured with the largest Transformer dimensionality of 768, 8 attention heads, and 5 encoder layers, resulting in the most resource-intensive setup with approximately 43 million parameters. Notably, the model employs a mere 10% dropout and was trained with the largest scheduled $6 \cdot 10^{-4}$ learning rate and the largest batch size of 512. Reaching the peak learning rate, the model accurately discovers an optimum that no other configuration can surpass. Moreover, even though the ranking method evaluates cosine and Euclidean validation metrics independently and favors the model with optimal performance in either metric, the top model selected based on cosine distances also attains near-optimal Euclidean correlation in the context of all experiments.

6.2 Analysis of DreaMS representations

In this section, we analyze the structure of DreaMS embedding space. First, we outline our findings regarding the encoded structural information within individual embeddings and the connections between their mutual distances. Following that, we highlight the emergence of molecular networks in the collections of DreaMS representations.

6.2.1 The space of DreaMS is organized by the structural properties of molecules

The standalone contrastive validation metrics demonstrate that DreaMS effectively encode structural features of small molecules. This is particularly evident when considering the distribution shift between experimental MassIVE spectra and the precise spectra of NIST20 used for the validation. To further assess the presence of structural features in experimental spectra that more closely resemble the

practical setup, we analyze the embeddings of a random sample of MoNA [15].

We randomly sample 10,000 spectra from MoNA and compute DREAMS embeddings for each of them. Subsequently, we project the 768-dimensional embedding space onto a 2-dimensional plane using the UMAP algorithm [138], labeling elements with various precursor properties (Figure 6.2). Diametrally, the space is organized according to molecular masses, as illustrated by the number of precursor carbon atoms (B). This observation is expected since each spectrum is associated with a precursor m/z. Nevertheless, the space exhibits a more complex structure concerning the counts of other elements, such as nitrogen (A). Furthermore, we evaluate the spatial organization of embeddings with respect to the compositions of structural features. We observe a distinct region in the space corresponding to large carbohydrates (i.e., molecules composed of only H, C, and O elements). Smaller carbohydrates are organized in a spiral path-like subspace between non-carbohydrates (C). Ultimately, the space captures a complex concept of quantitative drug-likeness estimation (QED), incorporating various molecular properties such as logP, topological polar surface area, the number of hydrogen bond donors and acceptors, the number of aromatic rings, and more (D).

Remarkably, the embedding space reveals a sharp local organization of spectra regarding mass spectrometry setups. For instance, UMAP (E) displays a tendency towards a negative correlation between CID energy and molecular size while also capturing small clusters of similar CID settings. We observe a similar organization of different types of MS instruments, with the QQ type forming notably dense regions (F).

6.2.2 Distance on DreaMS reflects the distance on molecules

The correlation validation metrics were introduced to more comprehensively evaluate the capacity of learned molecular features, as opposed to relying solely on the contrastive validation technique. In a similar vein, we conduct a more detailed analysis of the correlation between the cosine similarity of DreaMS and the Tanimoto similarity of the underlying molecules. Specifically, we sample 5,083 spectra from the in-house MCE spectral library[16], maximizing the entropy of the

[15] It is important to note that MoNA spectra may be included in MassIVE.

[16] Here, we utilize the MCE library to further perform a fair comparison with an existing method trained on spectral libraries.

distribution determined by the associated pairwise Tanimoto similarities. We then assess the Pearson correlation between all 25,836,889 spectral and compound distances.

We discover that such zero-shot predictions of molecular similarity from spectra yield a 0.44 correlation with Tanimoto similarity. We consider it a solid result, given that DreaMS was trained solely for the prediction of masked peaks and was not provided with any molecular information. For comparison, the classic modified cosine score achieves a correlation of 0.13, despite being considerably slower and having an explicitly given invariance to adducts. Ultimately, we compare DreaMS similarity with MS2DeepScore, a contrastive deep-learning method explicitly trained to maximize the correlation with Tanimoto distance in a supervised setting using molecular annotations from spectral libraries. MS2DeepScore achieves a higher correlation of 0.56 but the difference in correlation distributions is not extremely substantial when compared to the self-supervised DreaMS, as shown in Figure 6.3.

6.2.3 DreaMS as a source of novel information on mass spectra

Although the concept of molecular similarity is fundamental to the field of chemoinformatics, it cannot be rigorously defined or objectively evaluated [139]. While Tanimoto similarity based on ECFP fingerprints is often used to compare large collections of "arbitrary" compounds, there is no universal fingerprint suitable for a wide range of biological applications [140, 141]. This implies that contrastive learning methods applied to mass spectra, which aim to mimic predefined molecular similarities, are inherently biased towards the chosen similarity or similarities. Such a flaw is undesirable by the definition of the untargeted metabolomics.

The MassIVE database is a vast resource of undiscovered molecules. Consequently, training DreaMS in a self-supervised regime utlizing MassIVE allows the neural network to discover the concept of molecular similarity through their spectra without relying on human understanding of chemistry or the scope of known compounds. In particular, when evaluating DreaMS against MS2DeepScore and modified cosine distance, we identified examples of spectra where DreaMS exhibits a novel notion of similarity. Figure 6.4 demonstrates that our method encodes the resemblance of molecules that cannot be captured by either classic spec-

tral similarity or the molecular similarity imposed on MS2DEEPSCORE.

6.2.4 DreaMS induce molecular networks

Observing that DREAMS encode structural features of molecules and serve as a suitable similarity measure on spectra, we explore its potential as a novel approach to molecular networking. While existing techniques integrate multiple computational tools, one could construct molecular networks within a single, unified DREAMS framework. This framework is purely the geometry and topology of the DREAMS embedding space.

More precisely, utilizing 10,000 random spectra from MoNA, we construct a 3-NN graph with nodes representing individual spectra, each connected to its three nearest neighbors based on cosine distance within the DREAMS space. We then manually investigate its local and global structure by performing breadth-first searches (BFS) and depth-first-search-like (DFS) traversals, starting from randomly chosen spectra. Our analysis consistently reveals that local neighborhoods exhibit shared structural motifs in precursor molecules, yet develop anisotropically. In other words, spectra are not densely interconnected through 1-hop or 2-hop neighbors but rather evolve in distinct directions within the space of structural features (Figure 6.5).

To assess the 3-NN graph on a global scale, we generate cycle-free paths starting with random nodes and recursively selecting the nearest neighbor. Figure 6.6 illustrates an example of such a path. Analogous to the local analysis, we frequently observe a gradual morphing of molecules with respect to their structural properties. However, considering that we examine a space limited to 10,000 samples, transitions between spectra are often sharp in terms of molecular similarity.

6.3 Fine-tuning DreaMS

In the preceding section, we showed that DREAMS gain a general knowledge of the molecular structures through a course of self-supervised pre-training. In this section, we evaluate the fine-tuning of the model enabling the distillation of task-specific knowledge.

6.3.1 Self-supervised pre-training consistently improves the performance of DreaMS on the variety of downstream tasks

Employing the optimal pre-trained DreaMS model according to our ranking, we examine further training (i.e. fine-tuning) the model on two spectral libraries for a range of labeled downstream tasks (section Supervised fine-tuning): prediction of (i) 2048-bit ECFP fingerprints, (ii) the number of oxygen atoms, (iii) the presence of nitrogen, and (iv) the quantitative estimate of drug-likeliness (QED). By comparing the pre-trained model against the identical randomly initialized architecture, we observe a consistent increase in performance (Table 6.1).

In particular, the pre-trained model attains the highest validation metrics across all eight downstream tasks. As expected, we observe that fine-tuning benefits significantly from pre-training when conducted on a small GNPS dataset. Remarkably, the non-pre-trained model does not converge on the most challenging task of predicting QED. Although we only experiment with a single configuration for fine-tuning the pre-trained model, all our attempts to identify hyperparameters that enable the random model to converge on the task were unsuccessful. This finding strongly emphasizes the importance of pre-training. In general, both randomly initialized and pre-trained models significantly outperform the DeepSets baseline, emphasizing the strong inductive bias of the DreaMS architecture.

Task	**GNPS (SIRIUS split)**				**NIST20 (Murcko hist. split)**			
Model	**ECFP↑**	**#O↓**	**Has N↑**	**QED↓**	**ECFP↑**	**#O↓**	**Has N↑**	**QED↓**
DeepSets	0.26	2.27	0.44	0.52	0.12	2.65	0.6	0.46
DreaMS (random)	0.41	1.33	0.91	0.47	0.30	2.35	0.84	0.46
DreaMS (pre-trained)	**0.54**	**1.02**	**0.94**	**0.11**	**0.32**	**2.13**	**0.88**	**0.17**

Table 6.1: Pre-training consistently improves downstream supervised training. The values represent the best validation metrics achieved on various tasks and datasets introduced in section Supervised fine-tuning. "DreaMS (random)" refers to the randomly initialized DreaMS architecture, while "DreaMS (pre-trained)" denotes the DreaMS architecture pre-trained using self-supervised learning. The pre-trained model consistently outperforms both the randomly initialized DreaMS and the DeepSets baseline. The significance of pre-training is particularly noticeable in the most complex task of predicting QED.

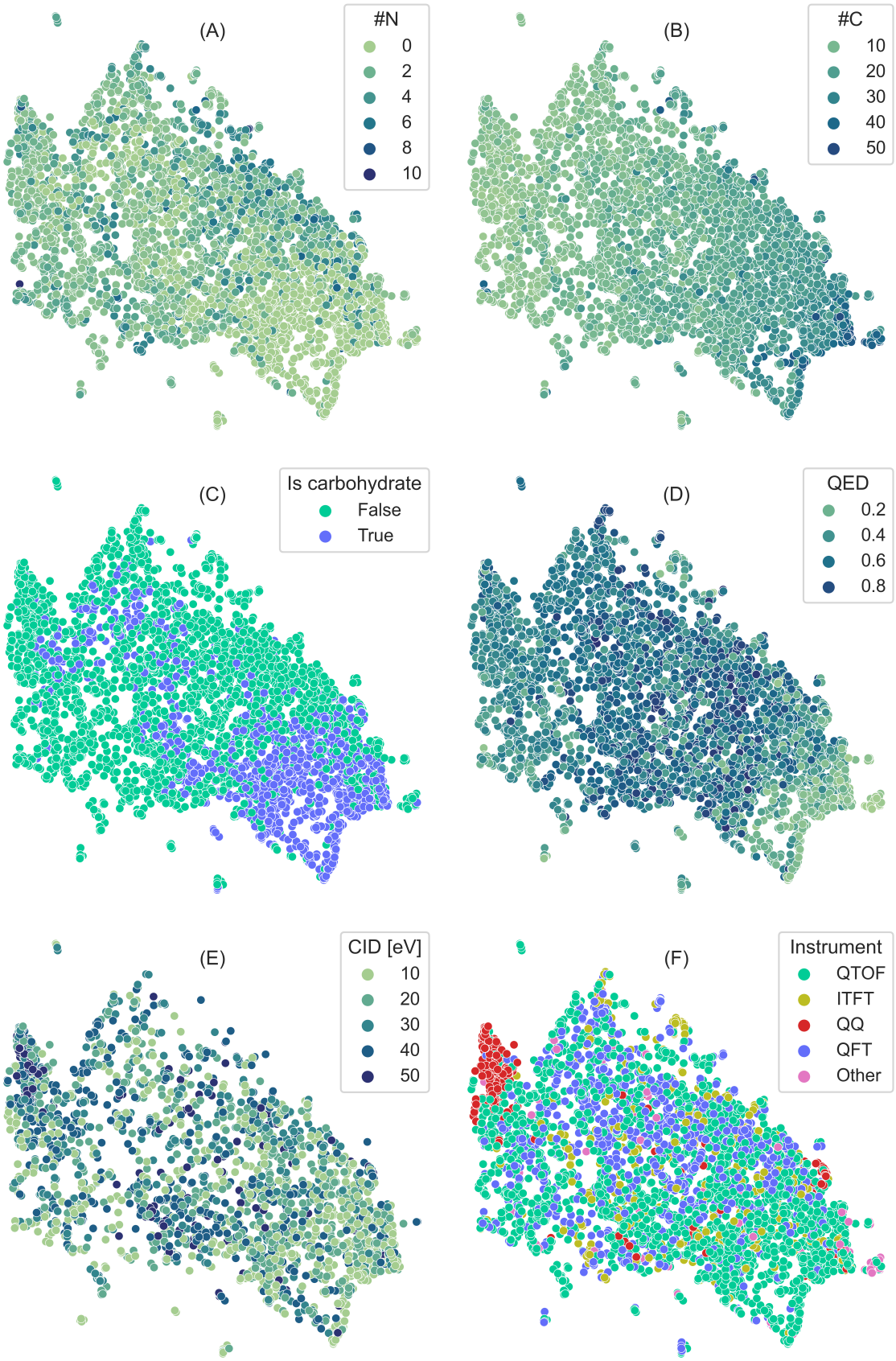


Figure 6.2: UMAP projections of the 10,000 DreaMS embeddings for random spectra from MoNA. Top row reveals the structural organization of DreaMS with respect to chemical formulas. Middle row demonstrates that DreaMS encode more intricate compositional properties of molecules such as being the carbohydrate or the quantitative estimation of drug-likeness (QED). Bottom row shows the sharp localization of DreaMS with respect to mass spectrometry setups.

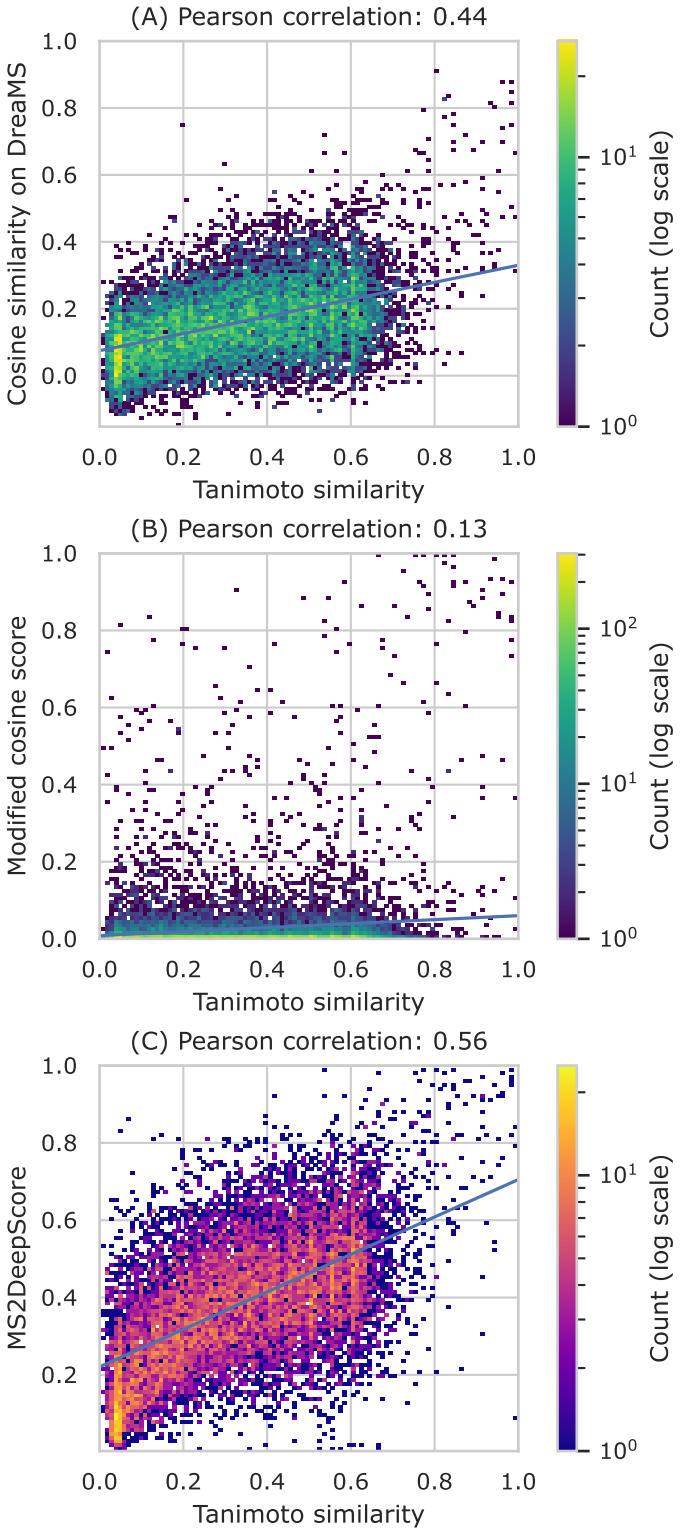


Figure 6.3: Correlation between spectral similarities and the Tanimoto similarity on underlying precursor molecules. The figure showcases three spectral similarity measures: cosine distance based on DreaMS, classic modified cosine distance, and MS2DeepScore. The green palette represents methods that do not employ spectral libraries, while the orange palette highlights the supervised MS2DeepScore, explicitly trained to maximize the correlation. Although the modified cosine distance fails to perform well in this evaluation, the self-supervised DreaMS demonstrates competitive performance compared to the supervised MS2DeepScore, even without utilizing any information about molecular structures during pre-training.

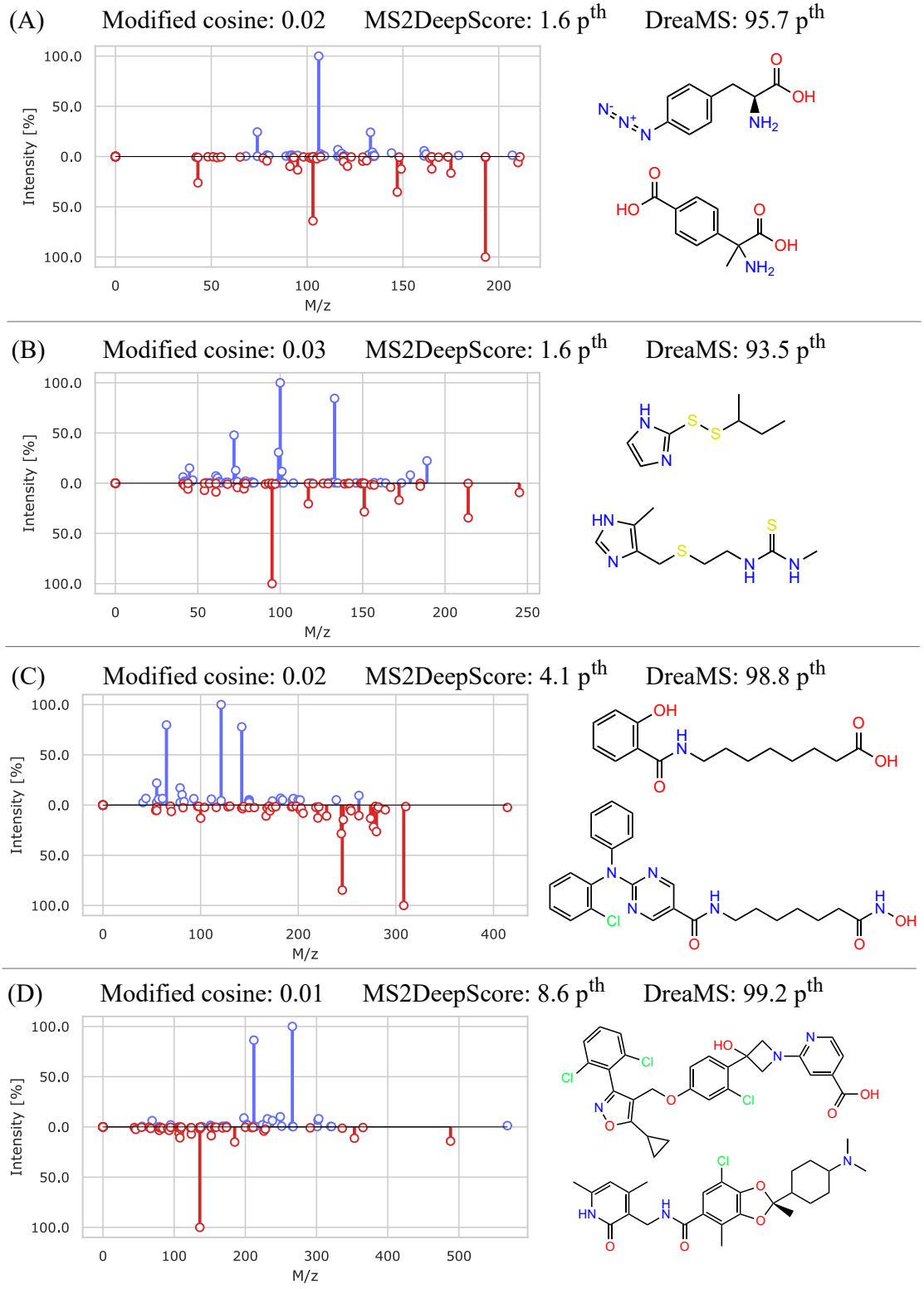


Figure 6.4: Examples of spectra with differing modified cosine similarity and MS2DeepScore values, but with similar DreaMS and shared structural features in precursor molecules. The figure illustrates that DreaMS encode latent properties of mass spectra not captured by either spectral similarity or the molecular similarity imposed on MS2DeepScore. The term "pth" denotes the percentile of the distance among the 25,836,889 pairwise distances between a sample of 5,083 spectra.

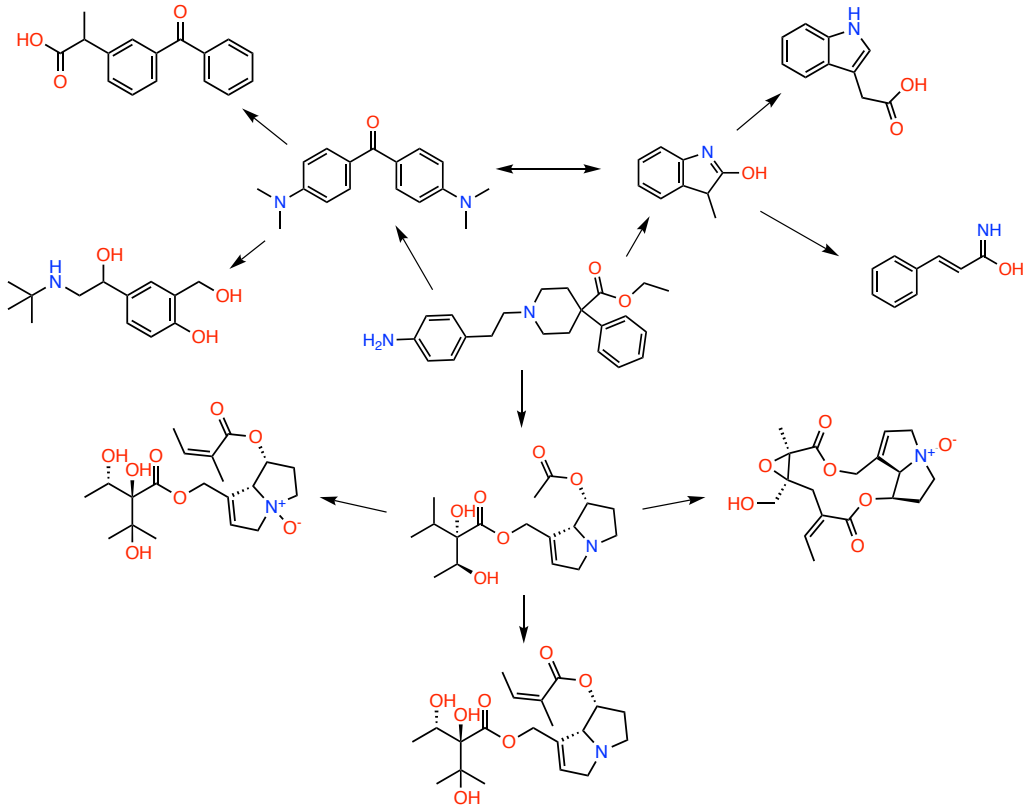


Figure 6.5: Example of a spectral neighborhood given by DreaMS embeddings of 10,000 random spectra from MoNA. The figure shows 2-hop neighbors of the central molecule in the DREAMS molecular network, where edges are given by the three closest embeddings in the sense of cosine distance. Notably, despite the proximity of nodes in the graph, only one pair of spectra is connected bidirectionally. Such an observation highlights the anisotropy of the embedding space with respect to molecular motifs.

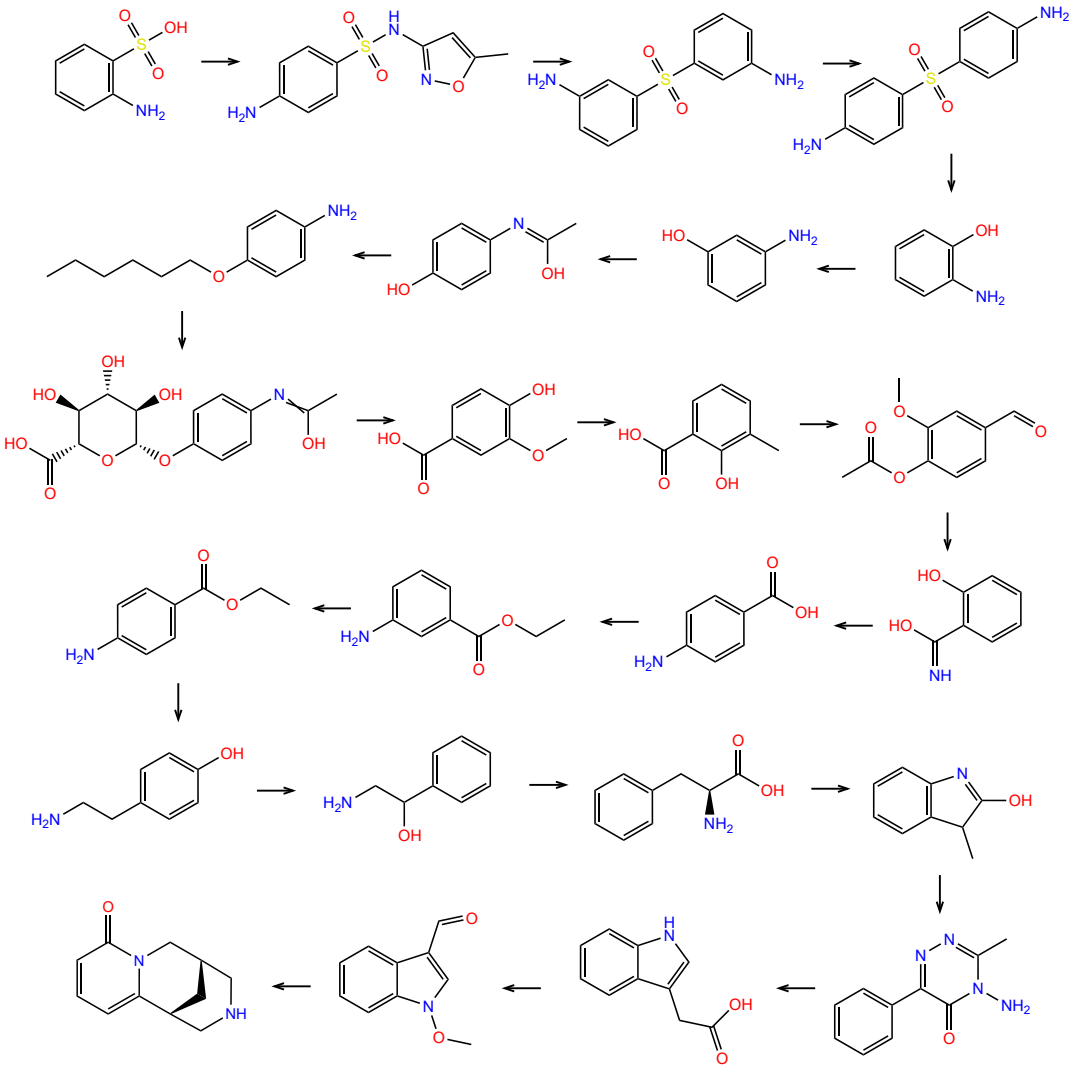


Figure 6.6: Fragment of a path in the molecular network induced by the 10,000 DREAMS embeddings. Starting with a spectrum corresponding to the top-left molecule, we recursive retrieve the other closest new spectrum with respect to the cosine distance on DREAMS. The associated precursors gradually morph along the space of structural motifs, yet with sharp transitions given by the limited variation of 10,000 spectra. Important to mention that the diameter of the entire network is equal to 19 and we permit the cycles when forming the path. Therefore, the depicted fragment most likely forms a convoluted knot-like structure in the space of DREAMS.

CHAPTER 7

Conclusions & Future work

Our study represents a pioneering effort in the development of a self-supervised foundation model for mass spectrometry. Obtained experimental findings demonstrate that by solely learning to predict masked m/z values of mass spectra, the proposed DREAMS neural network progressively extracts properties of molecular structures. Through the analysis of the learned representation space, we determined that it exhibits a complex organization capturing molecular properties and mass spectrometry configurations. Moreover, we discovered that the pre-trained model manifests a novel concept of spectral similarity, which is not attainable by existing methods. Lastly, we illustrated how DREAMS induces molecular networks and how end-to-end fine-tuning benefits from the pre-training on practical tasks.

The results obtained in this study underscore the potential of self-supervised learning to shift the paradigm of computational mass spectrometry and tackle the longstanding challenge of interpreting mass spectra. In contrast to current approaches that depend on human expertise or labor-intensive annotations, our method provides a way to extract knowledge purely from experimental mass spectrometry data. While simply scaling the DREAMS architecture and utilizing more training data from MassIVE or other repositories may further enhance the capacity of the representations [142], we acknowledge opportunities to make conceptual advancements in our methodology.

Firstly, the only mass spectrometry inductive bias considered in this work is Fourier features. We have not incorporated, for example, MS^1 spectra, nor have we equipped the attention mechanism with MS^n fragmentation priors. We believe that implementing such modifications could significantly improve the model archi-

tecture and enable us to benchmark our method against the state-of-the-art SIRIUS platform, which, for instance, derives chemical formulas and adducts leveraging MS^1 spectra.

Ultimately, our objective is to ensure that the DREAMS embeddings possess sufficient capacity for the *de novo* generation of molecular structures from spectra. Although annotated datasets are not extensive enough to solve the inverse annotation problem, generative modeling over the pre-trained embeddings may address this bottleneck. In particular, our plan involves investigating latent diffusion models [143] and GFlowNets [144] for proposing molecules from DREAMS embeddings. Successful experiments in this direction would imply an automated expansion of the discovered chemical space, and therefore, a breakthrough in life sciences.

Bibliography

[1] Gordon M Cragg and David J Newman. Natural products: a continuing source of novel drug leads. *Biochim Biophys Acta*, 1830(6):3670–3695, February 2013. doi: 10.1016/j.bbagen.2013.02.008. URL https://doi.org/10.1016/j.bbagen.2013.02.008.

[2] Atanas G. Atanasov, Sergey B. Zotchev, Verena M. Dirsch, Ilkay Erdogan Orhan, Maciej Banach, Judith M. Rollinger, Davide Barreca, Wolfram Weckwerth, Rudolf Bauer, Edward A. Bayer, Muhammed Majeed, Anupam Bishayee, Valery Bochkov, Günther K. Bonn, Nady Braidy, Franz Bucar, Alejandro Cifuentes, Grazia D'Onofrio, Michael Bodkin, Marc Diederich, Albena T. Dinkova-Kostova, Thomas Efferth, Khalid El Bairi, Nicolas Arkells, Tai-Ping Fan, Bernd L. Fiebich, Michael Freissmuth, Milen I. Georgiev, Simon Gibbons, Keith M. Godfrey, Christian W. Gruber, Jag Heer, Lukas A. Huber, Elena Ibanez, Anake Kijjoa, Anna K. Kiss, Aiping Lu, Francisco A. Macias, Mark J. S. Miller, Andrei Mocan, Rolf Müller, Ferdinando Nicoletti, George Perry, Valeria Pittalà, Luca Rastrelli, Michael Ristow, Gian Luigi Russo, Ana Sanches Silva, Daniela Schuster, Helen Sheridan, Krystyna Skalicka-Woźniak, Leandros Skaltsounis, Eduardo Sobarzo-Sánchez, David S. Bredt, Hermann Stuppner, Antoni Sureda, Nikolay T. Tzvetkov, Rosa Anna Vacca, Bharat B. Aggarwal, Maurizio Battino, Francesca Giampieri, Michael Wink, Jean-Luc Wolfender, Jianbo Xiao, Andy Wai Kan Yeung, Gérard Lizard, Michael A. Popp, Michael Heinrich, Ioana Berindan-Neagoe, Marc Stadler, Maria Daglia, Robert Verpoorte, Claudiu T. Supuran, and the International Natural Product Sciences Taskforce. Natu-

ral products in drug discovery: advances and opportunities. *Nature Reviews Drug Discovery*, 20(3):200–216, Mar 2021. ISSN 1474-1784. doi: 10.1038/s41573-020-00114-z. URL https://doi.org/10.1038/s41573-020-00114-z.

[3] Carmen Bedia, Paulo Cardoso, Núria Dalmau, Elba Garreta-Lara, Cristian Gómez-Canela, Eva Gorrochategui, Meritxell Navarro-Reig, Elena Ortiz-Villanueva, Francesc Puig-Castellví, and Romà Tauler. Chapter nineteen - applications of metabolomics analysis in environmental research. In Joaquim Jaumot, Carmen Bedia, and Romà Tauler, editors, *Data Analysis for Omic Sciences: Methods and Applications*, volume 82 of *Comprehensive Analytical Chemistry*, pages 533–582. Elsevier, 2018. doi: https://doi.org/10.1016/bs.coac.2018.07.006. URL https://www.sciencedirect.com/science/article/pii/S0166526X18300709.

[4] Ewa Szpyrka and Magdalena Słowik-Borowiec. Analysis of residues in environmental samples. *Molecules*, 28(7), 2023. ISSN 1420-3049. doi: 10.3390/molecules28073046. URL https://www.mdpi.com/1420-3049/28/7/3046.

[5] Royston Goodacre, Seetharaman Vaidyanathan, Warwick B. Dunn, George G. Harrigan, and Douglas B. Kell. Metabolomics by numbers: acquiring and understanding global metabolite data. *Trends in Biotechnology*, 22(5):245–252, 2004. ISSN 0167-7799. doi: https://doi.org/10.1016/j.tibtech.2004.03.007. URL https://www.sciencedirect.com/science/article/pii/S0167779904000812.

[6] Saleh Alseekh, Asaph Aharoni, Yariv Brotman, Kévin Contrepois, John D'Auria, Jan Ewald, Jennifer C. Ewald, Paul D. Fraser, Patrick Giavalisco, Robert D. Hall, Matthias Heinemann, Hannes Link, Jie Luo, Steffen Neumann, Jens Nielsen, Leonardo Perez de Souza, Kazuki Saito, Uwe Sauer, Frank C. Schroeder, Stefan Schuster, Gary Siuzdak, Aleksandra Skirycz, Lloyd W. Sumner, Michael P. Snyder, Huiru Tang, Takayuki Tohge, Yulan Wang, Weiwei Wen, Si Wu, Guowang Xu, Nicola Zamboni, and Alisdair R. Fernie. Mass spectrometry-based metabolomics: a guide for annotation, quantification and best reporting practices. *Nature Methods*, 18(7):747–756, Jul 2021. ISSN 1548-7105. doi: 10.1038/s41592-021-01197-1. URL https://doi.org/10.1038/s41592-021-01197-1.

[7] Saleh Alseekh and Alisdair R. Fernie. Metabolomics 20 years on: what have we learned and what hurdles remain? *The Plant Journal*, 94(6): 933–942, 2018. doi: https://doi.org/10.1111/tpj.13950. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/tpj.13950.

[8] Martin Giera, Oscar Yanes, and Gary Siuzdak. Metabolite discovery: Biochemistry's scientific driver. *Cell Metabolism*, 34(1): 21–34, 2022. ISSN 1550-4131. doi: https://doi.org/10.1016/j.cmet.2021.11.005. URL https://www.sciencedirect.com/science/article/pii/S1550413121005337.

[9] David S. Wishart. Metabolomics for investigating physiological and pathophysiological processes. *Physiological Reviews*, 99(4):1819–1875, 2019. doi: 10.1152/physrev.00035.2018. URL https://doi.org/10.1152/physrev.00035.2018. PMID: 31434538.

[10] Katja Dettmer, Pavel A Aronov, and Bruce D Hammock. Mass spectrometry-based metabolomics. *Mass Spectrom Rev*, 26(1):51–78, January 2007. doi: https://doi.org/10.1002/mas.20108. URL https://doi.org/10.1002/mas.20108.

[11] John B. Fenn, Matthias Mann, Chin Kai Meng, Shek Fu Wong, and Craig M. Whitehouse. Electrospray ionization for mass spectrometry of large biomolecules. *Science*, 246(4926):64–71, 1989. doi: 10.1126/science.2675315. URL https://www.science.org/doi/abs/10.1126/science.2675315.

[12] Fred W. McLafferty. Tandem mass spectrometry (ms/ms): a promising new analytical technique for specific component determination in complex mixtures. *Accounts of Chemical Research*, 13(2):33–39, Feb 1980. ISSN 0001-4842. doi: 10.1021/ar50146a001. URL https://doi.org/10.1021/ar50146a001.

[13] Ricardo R. da Silva, Pieter C. Dorrestein, and Robert A. Quinn. Illuminating the dark matter in metabolomics. *Proceedings of the National Academy of Sciences*, 112(41):12549–12550, 2015. doi: 10.1073/pnas.1516878112. URL https://www.pnas.org/doi/abs/10.1073/pnas.1516878112.

[14] Niek F. de Jonge, Kevin Mildau, David Meijer, Joris J. R. Louwen, Christoph Bueschl, Florian Huber, and Justin J. J. van der Hooft. Good prac-

tices and recommendations for using and benchmarking computational metabolomics metabolite annotation tools. *Metabolomics*, 18(12):103, Dec 2022. ISSN 1573-3890. doi: 10.1007/s11306-022-01963-y. URL https://doi.org/10.1007/s11306-022-01963-y.

[15] Kai Dührkop, Markus Fleischauer, Marcus Ludwig, Alexander A. Aksenov, Alexey V. Melnik, Marvin Meusel, Pieter C. Dorrestein, Juho Rousu, and Sebastian Böcker. Sirius 4: a rapid tool for turning tandem mass spectra into metabolite structure information. *Nature Methods*, 16(4):299–302, Apr 2019. ISSN 1548-7105. doi: 10.1038/s41592-019-0344-8. URL https://doi.org/10.1038/s41592-019-0344-8.

[16] Sebastian Böcker and Kai Dührkop. Fragmentation trees reloaded. *Journal of Cheminformatics*, 8(1):5, Feb 2016. ISSN 1758-2946. doi: 10.1186/s13321-016-0116-8. URL https://doi.org/10.1186/s13321-016-0116-8.

[17] Kai Dührkop, Huibin Shen, Marvin Meusel, Juho Rousu, and Sebastian Böcker. Searching molecular structure databases with tandem mass spectra using csi:fingerid. *Proceedings of the National Academy of Sciences*, 112 (41):12580–12585, 2015. doi: 10.1073/pnas.1509788112. URL https://www.pnas.org/doi/abs/10.1073/pnas.1509788112.

[18] Kai Dührkop, Louis-Félix Nothias, Markus Fleischauer, Raphael Reher, Marcus Ludwig, Martin A. Hoffmann, Daniel Petras, William H. Gerwick, Juho Rousu, Pieter C. Dorrestein, and Sebastian Böcker. Systematic classification of unknown metabolites using high-resolution fragmentation mass spectra. *Nature Biotechnology*, 39(4):462–471, Apr 2021. ISSN 1546-1696. doi: 10.1038/s41587-020-0740-8. URL https://doi.org/10.1038/s41587-020-0740-8.

[19] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and

Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, Aug 2021. ISSN 1476-4687. doi: 10.1038/s41586-021-03819-2. URL https://doi.org/10.1038/s41586-021-03819-2.

[20] J. Dauparas, I. Anishchenko, N. Bennett, H. Bai, R. J. Ragotte, L. F. Milles, B. I. M. Wicky, A. Courbet, R. J. de Haas, N. Bethel, P. J. Y. Leung, T. F. Huddy, S. Pellock, D. Tischer, F. Chan, B. Koepnick, H. Nguyen, A. Kang, B. Sankaran, A. K. Bera, N. P. King, and D. Baker. Robust deep learning-based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022. doi: 10.1126/science.add2187. URL https://www.science.org/doi/abs/10.1126/science.add2187.

[21] Andy Hsien-Wei Yeh, Christoffer Norn, Yakov Kipnis, Doug Tischer, Samuel J. Pellock, Declan Evans, Pengchen Ma, Gyu Rie Lee, Jason Z. Zhang, Ivan Anishchenko, Brian Coventry, Longxing Cao, Justas Dauparas, Samer Halabiya, Michelle DeWitt, Lauren Carter, K. N. Houk, and David Baker. De novo design of luciferases using deep learning. *Nature*, 614 (7949):774–780, Feb 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-05696-3. URL https://doi.org/10.1038/s41586-023-05696-3.

[22] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. URL http://www.deeplearningbook.org.

[23] Daniel Probst and Jean-Louis Reymond. Visualization of very large high-dimensional data sets as minimum spanning trees. *Journal of Cheminformatics*, 12(1):12, Feb 2020. ISSN 1758-2946. doi: 10.1186/s13321-020-0416-x. URL https://doi.org/10.1186/s13321-020-0416-x.

[24] National Institute of Standards and Technology. Nist standard reference database 1a. URL https://www.nist.gov/srd/.

[25] Hisayuki Horai, Masanori Arita, Shigehiko Kanaya, Yoshito Nihei, Tasuku Ikeda, Kazuhiro Suwa, Yuya Ojima, Kenichi Tanaka, Satoshi Tanaka, Ken Aoshima, Yoshiya Oda, Yuji Kakazu, Miyako Kusano, Takayuki Tohge, Fumio Matsuda, Yuji Sawada, Masami Yokota Hirai, Hiroki Nakanishi, Kazutaka Ikeda, Naoshige Akimoto, Takashi Maoka, Hiroki Takahashi, Takeshi Ara, Nozomu Sakurai, Hideyuki Suzuki, Daisuke Shibata, Steffen Neumann, Takashi Iida, Ken Tanaka, Kimito Funatsu, Fumito Matsuura, To-

moyoshi Soga, Ryo Taguchi, Kazuki Saito, and Takaaki Nishioka. MassBank: a public repository for sharing mass spectral data for life sciences. *J Mass Spectrom*, 45(7):703–714, July 2010. doi: https://doi.org/10.1002/jms.1777. URL https://doi.org/10.1002/jms.1777.

[26] Maria Sorokina, Peter Merseburger, Kohulan Rajan, Mehmet Aziz Yirik, and Christoph Steinbeck. Coconut online: Collection of open natural products database. *Journal of Cheminformatics*, 13(1):2, Jan 2021. ISSN 1758-2946. doi: 10.1186/s13321-020-00478-9. URL https://doi.org/10.1186/s13321-020-00478-9.

[27] A. Patrícia Bento, Anne Hersey, Eloy Félix, Greg Landrum, Anna Gaulton, Francis Atkinson, Louisa J. Bellis, Marleen De Veij, and Andrew R. Leach. An open source chemical structure curation pipeline using rdkit. *Journal of Cheminformatics*, 12(1):51, Sep 2020. ISSN 1758-2946. doi: 10.1186/s13321-020-00456-1. URL https://doi.org/10.1186/s13321-020-00456-1.

[28] J. Clayden, N. Greeves, and S. Warren. *Organic Chemistry*. OUP Oxford, 2012. ISBN 9780199270293. URL https://books.google.cz/books?id=G3wUe-2e-4MC.

[29] *Algorithmic Mass Spectrometry*. 2019. URL https://bio.informatik.uni-jena.de/textbook-algoms/.

[30] Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges, 2021. URL https://doi.org/10.48550/arXiv.2104.13478.

[31] Joseph L. Durant, Burton A. Leland, Douglas R. Henry, and James G. Nourse. Reoptimization of mdl keys for use in drug discovery. *Journal of Chemical Information and Computer Sciences*, 42(6):1273–1280, Nov 2002. ISSN 0095-2338. doi: 10.1021/ci010132r. URL https://doi.org/10.1021/ci010132r.

[32] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754, May 2010. ISSN 1549-9596. doi: 10.1021/ci100050t. URL https://doi.org/10.1021/ci100050t.

[33] Shandilya Mahamuni Baira, Srinivas Ragampeta, and M.V.N. Kumar Talluri. A comprehensive study on rearrangement reactions in collision-induced dissociation mass spectrometric fragmentation of protonated diphenyl and phenyl pyridyl ethers. *Rapid Communications in Mass Spectrometry*, 33 (18):1440–1448, 2019. doi: https://doi.org/10.1002/rcm.8488. URL https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/abs/10.1002/rcm.8488.

[34] Dejan Nikolić and David C. Lankin. Low energy collision-induced dissociation of azepine pictet–spengler adducts of nω-methylserotonin. *Journal of the American Society for Mass Spectrometry*, 34(2):182–192, Feb 2023. ISSN 1044-0305. doi: 10.1021/jasms.2c00247. URL https://doi.org/10.1021/jasms.2c00247.

[35] Kristina Mamko, Liudmila Kalichkina, Oleg Kotelnikov, Anna Nikulina, Natalia Dementeva, and Dmitry Novikov. Separation of cis/trans isomers of 4,5-dihydroxyimidazolidine-2-thione and 4,5-dimethoxyimidazolidine-2-thione by aqueous normal-phase hplc mode. *Chromatographia*, 83(9): 1087–1093, Sep 2020. ISSN 1612-1112. doi: 10.1007/s10337-020-03926-8. URL https://doi.org/10.1007/s10337-020-03926-8.

[36] Katarzyna Bus, Jerzy Sitkowski, Wojciech Bocian, Adam Zmysłowski, Karol Ofiara, and Arkadiusz Szterk. Separation of menaquinone-7 geometric isomers by semipreparative high-performance liquid chromatography with silver complexation and identification by nuclear magnetic resonance. *Food Chemistry*, 368:130890, 2022. ISSN 0308-8146. doi: https://doi.org/10.1016/j.foodchem.2021.130890. URL https://www.sciencedirect.com/science/article/pii/S0308814621018963.

[37] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2 (5):359–366, 1989. ISSN 0893-6080. doi: https://doi.org/10.1016/0893-6080(89)90020-8. URL https://www.sciencedirect.com/science/article/pii/0893608089900208.

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

[39] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

[40] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does bert look at? an analysis of bert's attention, 2019. URL https://doi.org/10.48550/arXiv.1906.04341.

[41] Patrick Schramowski, Cigdem Turan, Nico Andersen, Constantin A. Rothkopf, and Kristian Kersting. Large pre-trained language models contain human-like biases of what is right and wrong to do, 2022. URL https://doi.org/10.48550//arXiv.2103.11790.

[42] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023. doi: 10.1126/science.ade2574. URL https://www.science.org/doi/abs/10.1126/science.ade2574.

[43] Randall Balestriero, Mark Ibrahim, Vlad Sobal, Ari Morcos, Shashank Shekhar, Tom Goldstein, Florian Bordes, Adrien Bardes, Gregoire Mialon, Yuandong Tian, Avi Schwarzschild, Andrew Gordon Wilson, Jonas Geiping, Quentin Garrido, Pierre Fernandez, Amir Bar, Hamed Pirsiavash, Yann LeCun, and Micah Goldblum. A cookbook of self-supervised learning, 2023. URL https://doi.org/10.48550/arXiv.2304.12210.

[44] Yann LeCun and Ishan Misra. Self-supervised learning: The dark matter of intelligence, 2021. URL https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence/.

[45] Ruoxi Sun, Hanjun Dai, and Adams Wei Yu. Does gnn pretraining help molecular representation?, 2022. URL https://doi.org/10.48550/arXiv.2207.06010.

[46] Hang Hu, Jyothsna Padmakumar Bindu, and Julia Laskin. Self-supervised clustering of mass spectrometry imaging data using contrastive learning. *Chem. Sci.*, 13:90–98, 2022. doi: 10.1039/D1SC04077D. URL http://dx.doi.org/10.1039/D1SC04077D.

[47] Foivos Ntelemis, Yaochu Jin, and Spencer A. Thomas. A generic self-supervised framework of learning invariant discriminative features, 2022. URL https://doi.org/10.48550/arXiv.2202.06914.

[48] Henry Webel, Lili Niu, Annelaura Bach Nielsen, Marie Locard-Paulet, Matthias Mann, Lars Juhl Jensen, and Simon Rasmussen. Mass spectrometry-based proteomics imputation using self supervised deep learning. *bioRxiv*, 2023. doi: 10.1101/2023.01.12.523792. URL https://www.biorxiv.org/content/early/2023/02/01/2023.01.12.523792.

[49] Qiong Yang, Hongchao Ji, Hongmei Lu, and Zhimin Zhang. Prediction of liquid chromatographic retention time with graph neural networks to assist in small molecule identification. *Analytical Chemistry*, 93(4):2200–2206, Feb 2021. ISSN 0003-2700. doi: 10.1021/acs.analchem.0c04071. URL https://doi.org/10.1021/acs.analchem.0c04071.

[50] Constantino A. García, Alberto Gil-de-la Fuente, Coral Barbas, and Abraham Otero. Probabilistic metabolite annotation using retention time prediction and meta-learned projections. *Journal of Cheminformatics*, 14(1): 33, Jun 2022. ISSN 1758-2946. doi: 10.1186/s13321-022-00613-8. URL https://doi.org/10.1186/s13321-022-00613-8.

[51] Jesse G. Meyer. Deep learning neural network tools for proteomics. *Cell Reports Methods*, 1(2):100003, 2021. ISSN 2667-2375. doi: https://doi.org/10.1016/j.crmeth.2021.100003. URL https://www.sciencedirect.com/science/article/pii/S2667237521000035.

[52] Lars J. Kangas, Thomas O. Metz, Giorgis Isaac, Brian T. Schrom, Bojana Ginovska-Pangovska, Luning Wang, Li Tan, Robert R. Lewis, and John H. Miller. In silico identification software (ISIS): a machine learning approach to tandem mass spectral identification of lipids. *Bioinformatics*, 28(13): 1705–1713, 05 2012. ISSN 1367-4803. doi: 10.1093/bioinformatics/bts194. URL https://doi.org/10.1093/bioinformatics/bts194.

[53] Yuanyue Li, Tobias Kind, Jacob Folz, Arpana Vaniya, Sajjan Singh Mehta, and Oliver Fiehn. Spectral entropy outperforms MS/MS dot product similarity for small-molecule compound identification. *Nature Methods*, 18(12): 1524–1531, Dec 2021. ISSN 1548-7105. doi: 10.1038/s41592-021-01331-z. URL https://doi.org/10.1038/s41592-021-01331-z.

[54] Arun S. Moorthy and Anthony J. Kearsley. Pattern similarity measures applied to mass spectra. In Manuel Cruz, Carlos Parés, and Peregrina Quintela, editors, *Progress in Industrial Mathematics: Success Stories*, pages 43–53, Cham, 2021. Springer International Publishing. ISBN 978-3-030-61844-5. doi: 10.1007/978-3-030-61844-5_4. URL https://doi.org/10.1007/978-3-030-61844-5_4.

[55] Stephen E. Stein and Donald R. Scott. Optimization and testing of mass spectral library search algorithms for compound identification. *Journal of the American Society for Mass Spectrometry*, 5(9): 859–866, 1994. ISSN 1044-0305. doi: https://doi.org/10.1016/1044-0305(94)87009-8. URL https://www.sciencedirect.com/science/article/pii/1044030594870098.

[56] Jeramie Watrous, Patrick Roach, Theodore Alexandrov, Brandi S. Heath, Jane Y. Yang, Roland D. Kersten, Menno van der Voort, Kit Pogliano, Harald Gross, Jos M. Raaijmakers, Bradley S. Moore, Julia Laskin, Nuno Bandeira, and Pieter C. Dorrestein. Mass spectral molecular networking of living microbial colonies. *Proceedings of the National Academy of Sciences*, 109(26):E1743–E1752, 2012. doi: 10.1073/pnas.1203689109. URL https://www.pnas.org/doi/abs/10.1073/pnas.1203689109.

[57] Florian Huber, Sven van der Burg, Justin J. J. van der Hooft, and Lars Ridder. Ms2deepscore: a novel deep learning similarity measure to compare tandem mass spectra. *Journal of Cheminformatics*, 13(1):84, Oct 2021. ISSN 1758-2946. doi: 10.1186/s13321-021-00558-4. URL https://doi.org/10.1186/s13321-021-00558-4.

[58] Gennady Voronov, Rose Lightheart, Joe Davison, Christoph A. Krettler, David Healey, and Thomas Butler. Multi-scale sinusoidal embeddings enable learning on high resolution mass spectrometry data, 2022. URL http://arxiv.org/abs/2207.02980.

[59] Melih Yilmaz, William E. Fondrie, Wout Bittremieux, Sewoong Oh, and William Stafford Noble. De novo mass spectrometry peptide sequencing with a transformer model. *bioRxiv*, 2022. doi: 10.1101/2022.02.07.479481. URL https://www.biorxiv.org/content/early/2022/06/18/2022.02.07.479481.

[60] Wout Bittremieux, Damon H. May, Jeffrey Bilmes, and William Stafford Noble. A learned embedding for efficient joint analysis of millions of mass spectra. *Nature Methods*, 19(6):675–678, Jun 2022. ISSN 1548-7105. doi: 10.1038/s41592-022-01496-1. URL https://doi.org/10.1038/s41592-022-01496-1.

[61] Justin Johan Jozias van der Hooft, Joe Wandy, Michael P. Barrett, Karl E. V. Burgess, and Simon Rogers. Topic modeling for untargeted substructure exploration in metabolomics. *Proceedings of the National Academy of Sciences*, 113(48):13738–13743, 2016. doi: 10.1073/pnas.1608041113. URL https://www.pnas.org/doi/abs/10.1073/pnas.1608041113.

[62] Florian Huber, Lars Ridder, Stefan Verhoeven, Jurriaan H. Spaaks, Faruk Diblen, Simon Rogers, and Justin J. J. van der Hooft. Spec2vec: Improved mass spectral similarity scoring through learning of structural relationships. *PLOS Computational Biology*, 17(2):1–18, 02 2021. doi: 10.1371/journal.pcbi.1008724. URL https://doi.org/10.1371/journal.pcbi.1008724.

[63] Fei Wang, Jaanus Liigand, Siyang Tian, David Arndt, Russell Greiner, and David S. Wishart. Cfm-id 4.0: More accurate esi-ms/ms spectral prediction and compound identification. *Analytical Chemistry*, 93(34):11692–11700, Aug 2021. ISSN 0003-2700. doi: 10.1021/acs.analchem.1c01465. URL https://doi.org/10.1021/acs.analchem.1c01465.

[64] Lars Ridder, Justin J. J. van der Hooft, Stefan Verhoeven, Ric C. H. de Vos, René van Schaik, and Jacques Vervoort. Substructure-based annotation of high-resolution multistage msn spectral trees. *Rapid Communications in Mass Spectrometry*, 26(20): 2461–2471, 2012. doi: https://doi.org/10.1002/rcm.6364. URL https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/abs/10.1002/rcm.6364.

[65] Christoph Ruttkies, Emma L. Schymanski, Sebastian Wolf, Juliane Hollender, and Steffen Neumann. Metfrag relaunched: incorporating strategies beyond in silico fragmentation. *Journal of Cheminformatics*, 8(1): 3, Jan 2016. ISSN 1758-2946. doi: 10.1186/s13321-016-0115-9. URL https://doi.org/10.1186/s13321-016-0115-9.

[66] Jennifer N. Wei, David Belanger, Ryan P. Adams, and D. Sculley. Rapid prediction of electron–ionization mass spectrometry using neural networks. *ACS Central Science*, 5(4):700–708, Apr 2019. ISSN 2374-7943. doi: 10.1021/acscentsci.9b00085. URL https://doi.org/10.1021/acscentsci.9b00085.

[67] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks, 2017. URL https://doi.org/10.48550/1609.02907.

[68] Hao Zhu, Liping Liu, and Soha Hassoun. Using graph neural networks for mass spectrometry prediction, 2020. URL https://doi.org/10.1021/2010.04661.

[69] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks, 2018. URL https://doi.org/10.48550/1710.10903.

[70] Richard Overstreet, Ethan King, Julia Nguyen, and Danielle Ciesielski. Qc-gn2oms2: a graph neural net for high resolution mass spectra prediction. *bioRxiv*, 2023. doi: 10.1101/2023.01.16.524269. URL https://www.biorxiv.org/content/early/2023/01/19/2023.01.16.524269.

[71] Adamo Young, Bo Wang, and Hannes Röst. Massformer: Tandem mass spectrum prediction with graph transformers, 2021. URL https://doi.org/10.48550/2111.04824.

[72] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform bad for graph representation?, 2021. URL https://doi.org/10.48550/arXiv.2106.05234.

[73] Xinmeng Li, Hao Zhu, Li ping Liu, and Soha Hassoun. Ensemble spectral prediction (esp) model for metabolite annotation, 2022.

[74] Samuel Goldman, Janet Li, and Connor W. Coley. Generating molecular fragmentation graphs with autoregressive neural networks, 2023. URL https://doi.org/10.48550/arXiv.2304.13136.

[75] Michael Murphy, Stefanie Jegelka, Ernest Fraenkel, Tobias Kind, David Healey, and Thomas Butler. Efficiently predicting high resolution mass spectra with graph neural networks, 2023. URL https://doi.org/10.48550/arXiv.2301.11419.

[76] Kai Dührkop, Marcus Ludwig, Marvin Meusel, and Sebastian Böcker. Faster mass decomposition. In Aaron Darling and Jens Stoye, editors, *Algorithms in Bioinformatics*, pages 45–58, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-40453-5. doi: https://doi.org/10.1007/978-3-642-40453-5_5. URL https://doi.org/10.1007/978-3-642-40453-5_5.

[77] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, Sep 1995. ISSN 1573-0565. doi: 10.1007/BF00994018. URL https://doi.org/10.1007/BF00994018.

[78] Kai Dührkop. Deep kernel learning improves molecular fingerprint prediction from tandem mass spectra. *Bioinformatics*, 38(Suppl 1):i342–i349, June 2022.

[79] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, Leonid Zaslavsky, Jian Zhang, and Evan E Bolton. PubChem 2023 update. *Nucleic Acids Research*, 51(D1):D1373–D1380, 10 2022. ISSN 0305-1048. doi: 10.1093/nar/gkac956. URL https://doi.org/10.1093/nar/gkac956.

[80] Ziling Fan, Amber Alley, Kian Ghaffari, and Habtom W. Ressom. Metfid: artificial neural network-based compound fingerprint prediction for metabolite annotation. *Metabolomics*, 16(10):104, Sep 2020. ISSN 1573-3890. doi: 10.1007/s11306-020-01726-7. URL https://doi.org/10.1007/s11306-020-01726-7.

[81] Viviana Consonni, Fabio Gosetti, Veronica Termopoli, Roberto Todeschini, Cecile Valsecchi, and Davide Ballabio. Multi-task neural networks and molecular fingerprints to enhance compound identification from lc-ms/ms data. *Molecules*, 27(18), 2022. ISSN 1420-3049. doi: 10.3390/molecules27185827. URL https://www.mdpi.com/1420-3049/27/18/5827.

[82] Samuel Goldman, Jeremy Wohlwend, Martin Stražar, Guy Haroush, Ramnik J. Xavier, and Connor W. Coley. Annotating metabolite mass spectra

with domain-inspired chemical formula transformers. *bioRxiv*, 2022. doi: 10.1101/2022.12.30.522318. URL https://www.biorxiv.org/content/early/2022/12/31/2022.12.30.522318.

[83] Gennady Voronov, Abe Frandsen, Brian Bargh, David Healey, Rose Lightheart, Tobias Kind, Pieter Dorrestein, Viswa Colluru, and Thomas Butler. Ms2prop: A machine learning model that directly predicts chemical properties from mass spectrometry data for novel compounds. *bioRxiv*, 2022. doi: 10.1101/2022.10.09.511482. URL https://www.biorxiv.org/content/early/2022/10/11/2022.10.09.511482.

[84] G. Richard Bickerton, Gaia V. Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L. Hopkins. Quantifying the chemical beauty of drugs. *Nature Chemistry*, 4(2):90–98, Feb 2012. ISSN 1755-4349. doi: 10.1038/nchem.1243. URL https://doi.org/10.1038/nchem.1243.

[85] Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of Cheminformatics*, 1(1):8, Jun 2009. ISSN 1758-2946. doi: 10.1186/1758-2946-1-8. URL https://doi.org/10.1186/1758-2946-1-8.

[86] Jian Gong Jaden J. A. Hastings G. Matthew Fricke Nathalie Cabrol Scott Sandford Michael Phillips Kimberley Warren-Rhodes Atılım Güneş Baydin Timothy D. Gebhard, Aaron C. Bell. Inferring molecular complexity from mass spectrometry data using machine learning. 12 2022. URL https://ml4physicalsciences.github.io/2022/files/NeurIPS_ML4PS_2022_94.pdf.

[87] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939785. URL https://doi.org/10.1145/2939672.2939785.

[88] Thomas Böttcher. An additive definition of molecular complexity. *Journal of Chemical Information and Modeling*, 56(3):462–470, Mar 2016. ISSN 1549-

9596. doi: 10.1021/acs.jcim.5b00723. URL https://doi.org/10.1021/acs.jcim.5b00723.

[89] Mengji Zhang, Yingce Xia, Nian Wu, Kun Qian, and Jianyang Zeng. MS^2-transformer: An end-to-end model for MS/MS-assisted molecule identification, 2022. URL https://openreview.net/forum?id=XK4GN6UCTfH.

[90] Soumitra Samanta, Steve O'Hagan, Neil Swainston, Timothy J. Roberts, and Douglas B. Kell. Vae-sim: A novel molecular similarity measure based on a variational autoencoder. *Molecules*, 25(15), 2020. ISSN 1420-3049. doi: 10.3390/molecules25153446. URL https://www.mdpi.com/1420-3049/25/15/3446.

[91] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. URL https://doi.org/10.48550/arXiv.1312.6114.

[92] Aditya Divyakant Shrivastava, Neil Swainston, Soumitra Samanta, Ivayla Roberts, Marina Wright Muelas, and Douglas B Kell. MassGenie: A Transformer-Based deep learning method for identifying small molecules from their mass spectra. *Biomolecules*, 11(12), November 2021. doi: https://doi.org/10.1002/jms.1777. URL https://doi.org/10.1002/jms.1777.

[93] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry, 2017. URL https://doi.org/10.48550/arXiv.1704.01212.

[94] Michael A. Stravs, Kai Dührkop, Sebastian Böcker, and Nicola Zamboni. Msnovelist: de novo structure generation from mass spectra. *Nature Methods*, 19(7):865–870, Jul 2022. ISSN 1548-7105. doi: 10.1038/s41592-022-01486-3. URL https://doi.org/10.1038/s41592-022-01486-3.

[95] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, nov 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

[96] Florent Olivon, Nicolas Elie, Gwendal Grelier, Fanny Roussi, Marc Litaudon, and David Touboul. Metgem software for the generation of molecular networks based on the t-sne algorithm. *Analytical Chemistry*, 90(23):13900–

13908, Dec 2018. ISSN 0003-2700. doi: 10.1021/acs.analchem.8b03099. URL https://doi.org/10.1021/acs.analchem.8b03099.

[97] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

[98] Louis-Félix Nothias, Daniel Petras, Robin Schmid, Kai Dührkop, Johannes Rainer, Abinesh Sarvepalli, Ivan Protsyuk, Madeleine Ernst, Hiroshi Tsugawa, Markus Fleischauer, Fabian Aicheler, Alexander A. Aksenov, Oliver Alka, Pierre-Marie Allard, Aiko Barsch, Xavier Cachet, Andres Mauricio Caraballo-Rodriguez, Ricardo R. Da Silva, Tam Dang, Neha Garg, Julia M. Gauglitz, Alexey Gurevich, Giorgis Isaac, Alan K. Jarmusch, Zdeněk Kameník, Kyo Bin Kang, Nikolas Kessler, Irina Koester, Ansgar Korf, Audrey Le Gouellec, Marcus Ludwig, Christian Martin H., Laura-Isobel McCall, Jonathan McSayles, Sven W. Meyer, Hosein Mohimani, Mustafa Morsy, Oriane Moyne, Steffen Neumann, Heiko Neuweger, Ngoc Hung Nguyen, Melissa Nothias-Esposito, Julien Paolini, Vanessa V. Phelan, Tomáš Pluskal, Robert A. Quinn, Simon Rogers, Bindesh Shrestha, Anupriya Tripathi, Justin J. J. van der Hooft, Fernando Vargas, Kelly C. Weldon, Michael Witting, Heejung Yang, Zheng Zhang, Florian Zubeil, Oliver Kohlbacher, Sebastian Böcker, Theodore Alexandrov, Nuno Bandeira, Mingxun Wang, and Pieter C. Dorrestein. Feature-based molecular networking in the gnps analysis environment. *Nature Methods*, 17(9):905–908, Sep 2020. ISSN 1548-7105. doi: 10.1038/s41592-020-0933-6. URL https://doi.org/10.1038/s41592-020-0933-6.

[99] Robin Schmid, Daniel Petras, Louis-Félix Nothias, Mingxun Wang, Allegra T. Aron, Annika Jagels, Hiroshi Tsugawa, Johannes Rainer, Mar Garcia-Aloy, Kai Dührkop, Ansgar Korf, Tomáš Pluskal, Zdeněk Kameník, Alan K. Jarmusch, Andrés Mauricio Caraballo-Rodríguez, Kelly C. Weldon, Melissa Nothias-Esposito, Alexander A. Aksenov, Anelize Bauermeister, Andrea Albarracin Orio, Carlismari O. Grundmann, Fernando Vargas, Irina Koester, Julia M. Gauglitz, Emily C. Gentry, Yannick Hövelmann, Svetlana A. Kalinina, Matthew A. Pendergraft, Morgan Panitchpakdi, Richard Tehan, Audrey Le Gouellec, Gajender Aleti, Helena Mannochio Russo, Birgit Arndt, Florian Hübner, Heiko Hayen, Hui Zhi, Manuela Raffatellu, Kimberly A. Prather, Li-

hini I. Aluwihare, Sebastian Böcker, Kerry L. McPhail, Hans-Ulrich Humpf, Uwe Karst, and Pieter C. Dorrestein. Ion identity molecular networking for mass spectrometry-based metabolomics in the gnps environment. *Nature Communications*, 12(1):3832, Jun 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-23953-9. URL https://doi.org/10.1038/s41467-021-23953-9.

[100] Allegra T. Aron, Emily C. Gentry, Kerry L. McPhail, Louis-Félix Nothias, Mélissa Nothias-Esposito, Amina Bouslimani, Daniel Petras, Julia M. Gauglitz, Nicole Sikora, Fernando Vargas, Justin J. J. van der Hooft, Madeleine Ernst, Kyo Bin Kang, Christine M. Aceves, Andrés Mauricio Caraballo-Rodríguez, Irina Koester, Kelly C. Weldon, Samuel Bertrand, Catherine Roullier, Kunyang Sun, Richard M. Tehan, Cristopher A. Boya P., Martin H. Christian, Marcelino Gutiérrez, Aldo Moreno Ulloa, Javier Andres Tejeda Mora, Randy Mojica-Flores, Johant Lakey-Beitia, Victor Vásquez-Chaves, Yilue Zhang, Angela I. Calderón, Nicole Tayler, Robert A. Keyzers, Fidele Tugizimana, Nombuso Ndlovu, Alexander A. Aksenov, Alan K. Jarmusch, Robin Schmid, Andrew W. Truman, Nuno Bandeira, Mingxun Wang, and Pieter C. Dorrestein. Reproducible molecular networking of untargeted mass spectrometry data using gnps. *Nature Protocols*, 15(6):1954–1991, Jun 2020. ISSN 1750-2799. doi: 10.1038/s41596-020-0317-5. URL https://doi.org/10.1038/s41596-020-0317-5.

[101] Flaminia Vincenti, Camilla Montesano, Francesca Di Ottavio, Adolfo Gregori, Dario Compagnone, Manuel Sergi, and Pieter Dorrestein. Molecular networking: A useful tool for the identification of new psychoactive substances in seizures by lc–hrms. *Frontiers in Chemistry*, 8, 2020. ISSN 2296-2646. doi: 10.3389/fchem.2020.572952. URL https://www.frontiersin.org/articles/10.3389/fchem.2020.572952.

[102] Nicholas J. Morehouse, Trevor N. Clark, Emily J. McMann, Jeffrey A. van Santen, F. P. Jake Haeckl, Christopher A. Gray, and Roger G. Linington. Annotation of natural product compound families using molecular networking topology and structural similarity fingerprinting. *Nature Communications*, 14(1):308, Jan 2023. ISSN 2041-1723. doi: 10.1038/s41467-022-35734-z. URL https://doi.org/10.1038/s41467-022-35734-z.

[103] Louis-Félix Nothias, Mélissa Nothias-Esposito, Ricardo da Silva, Mingxun Wang, Ivan Protsyuk, Zheng Zhang, Abi Sarvepalli, Pieter Leyssen, David Touboul, Jean Costa, Julien Paolini, Theodore Alexandrov, Marc Litaudon, and Pieter C. Dorrestein. Bioactivity-based molecular networking for the discovery of drug leads in natural product bioassay-guided fractionation. *Journal of Natural Products*, 81(4):758–767, 2018. doi: 10.1021/acs.jnatprod.7b00737. URL https://doi.org/10.1021/acs.jnatprod.7b00737. PMID: 29498278.

[104] Lerato Nephali, Paul Steenkamp, Karl Burgess, Johan Huyser, Margaretha Brand, Justin J. J. van der Hooft, and Fidele Tugizimana. Mass spectral molecular networking to profile the metabolome of biostimulant bacillus strains. *Frontiers in Plant Science*, 13, 2022. ISSN 1664-462X. doi: 10.3389/fpls.2022.920963. URL https://www.frontiersin.org/articles/10.3389/fpls.2022.920963.

[105] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, mar 2003. ISSN 1532-4435. URL https://proceedings.neurips.cc/paper_files/paper/2001/file/296472c9542ad4d4788d543508116cbc-Paper.pdf.

[106] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013. URL https://doi.org/10.48550/arXiv.1301.3781.

[107] Svetlana Kutuzova, Christian Igel, Mads Nielsen, and Douglas McCloskey. Bi-modal variational autoencoders for metabolite identification using tandem mass spectrometry. *bioRxiv*, 2021. doi: 10.1101/2021.08.03.454944. URL https://www.biorxiv.org/content/early/2021/08/04/2021.08.03.454944.

[108] Yuge Shi, N. Siddharth, Brooks Paige, and Philip H. S. Torr. Variational mixture-of-experts autoencoders for multi-modal deep generative models, 2019. URL https://doi.org/10.48550/1911.03393.

[109] Emma L Schymanski and Steffen Neumann. The critical assessment of small molecule identification (CASMI): Challenges and solutions. *Metabolites*, 3(3):517–538, June 2013. doi: 10.3390/metabo3030517. URL https://doi.org/10.3390/metabo3030517.

[110] Yujia Bao and Regina Barzilay. Learning to split for automatic bias detection, 2022. URL https://doi.org/10.48550/arXiv.2204.13749.

[111] Guy W. Bemis and Mark A. Murcko. The properties of known drugs. 1. molecular frameworks. *Journal of Medicinal Chemistry*, 39(15):2887–2893, Jan 1996. ISSN 0022-2623. doi: 10.1021/jm9602928. URL https://doi.org/10.1021/jm9602928.

[112] A. Lehman B. Weisfeiler. The reduction of a graph to canonical form and the algebra which appears therein, 1968. URL https://www.iti.zcu.cz/wl2018/pdf/wl_paper_translation.pdf.

[113] Lennart Martens, Matthew Chambers, Marc Sturm, Darren Kessner, Fredrik Levander, Jim Shofstahl, Wilfred Tang, Andreas Römpp, Steffen Neumann, Angel Pizarro, Luisa Montecchi-Palazzi, Natalie Tasman, Mike Coleman, Florian Reisinger, Puneet Souda, Henning Hermjakob, Pierre-Alain Binz, and Eric Deutsch. mzml—a community standard for mass spectrometry data. *Molecular cellular proteomics : MCP*, 10:R110.000133, 01 2011. doi: 10.1074/mcp.R110.000133. URL https://doi.org/10.1074/mcp.R110.000133.

[114] Patrick G. A. Pedrioli, Jimmy K. Eng, Robert Hubley, Mathijs Vogelzang, Eric W. Deutsch, Brian Raught, Brian Pratt, Erik Nilsson, Ruth H. Angeletti, Rolf Apweiler, Kei Cheung, Catherine E. Costello, Henning Hermjakob, Sequin Huang, Randall K. Julian, Eugene Kapp, Mark E. McComb, Stephen G. Oliver, Gilbert Omenn, Norman W. Paton, Richard Simpson, Richard Smith, Chris F. Taylor, Weimin Zhu, and Ruedi Aebersold. A common open representation of mass spectrometry data and its application to proteomics research. *Nature Biotechnology*, 22(11):1459–1466, Nov 2004. ISSN 1546-1696. doi: 10.1038/nbt1031. URL https://doi.org/10.1038/nbt1031.

[115] Robin Schmid, Steffen Heuckeroth, Ansgar Korf, Aleksandr Smirnov, Owen Myers, Thomas S. Dyrlund, Roman Bushuiev, Kevin J. Murray, Nils Hoffmann, Miaoshan Lu, Abinesh Sarvepalli, Zheng Zhang, Markus Fleischauer, Kai Dührkop, Mark Wesner, Shawn J. Hoogstra, Edward Rudt, Olena Mokshyna, Corinna Brungs, Kirill Ponomarov, Lana Mutabdžija, Tito Damiani, Chris J. Pudney, Mark Earll, Patrick O. Helmer, Timothy R. Fallon, Tobias Schulze, Albert Rivas-Ubach, Aivett Bilbao, Henning Richter, Louis-Félix

Nothias, Mingxun Wang, Matej Orešič, Jing-Ke Weng, Sebastian Böcker, Astrid Jeibmann, Heiko Hayen, Uwe Karst, Pieter C. Dorrestein, Daniel Petras, Xiuxia Du, and Tomáš Pluskal. Integrative analysis of multimodal mass spectrometry data in mzmine 3. *Nature Biotechnology*, 41(4):447–449, Apr 2023. ISSN 1546-1696. doi: 10.1038/s41587-023-01690-2. URL https://doi.org/10.1038/s41587-023-01690-2.

[116] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt, 2019. URL https://doi.org/10.48550/1912.02292.

[117] Su Jung Kim, Su Hee Kim, Ji Hyun Kim, Shin Hwang, and Hyun Ju Yoo. Understanding metabolomics in biomedical research. *Endocrinol Metab (Seoul)*, 31(1):7–16, March 2016.

[118] Dominika Strzelecka, Sebastian Chmielinski, Sylwia Bednarek, Jacek Jemielity, and Joanna Kowalska. Analysis of mononucleotides by tandem mass spectrometry: investigation of fragmentation pathways for phosphate- and ribose-modified nucleotide analogues. *Scientific Reports*, 7 (1):8931, Aug 2017. ISSN 2045-2322. doi: 10.1038/s41598-017-09416-6. URL https://doi.org/10.1038/s41598-017-09416-6.

[119] Yousheng Hua, Samuel B Wainhaus, Yanan Yang, Lixin Shen, Yansan Xiong, Xiaoying Xu, Fagen Zhang, Judy L Bolton, and Richard B van Breemen. Comparison of negative and positive ion electrospray tandem mass spectrometry for the liquid chromatography tandem mass spectrometry analysis of oxidized deoxynucleosides. *Journal of the American Society for Mass Spectrometry*, 12(1):80–87, 2001. ISSN 1044-0305. doi: https://doi.org/10.1016/S1044-0305(00)00191-4. URL https://www.sciencedirect.com/science/article/pii/S1044030500001914.

[120] Moses S. Charikar. Similarity estimation techniques from rounding algorithms. In *Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing*, STOC '02, page 380–388, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 1581134959. doi: 10.1145/509907.509965. URL https://doi.org/10.1145/509907.509965.

[121] Peter W. Battaglia, Jessica B. Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David

Raposo, Adam Santoro, Ryan Faulkner, Caglar Gulcehre, Francis Song, Andrew Ballard, Justin Gilmer, George Dahl, Ashish Vaswani, Kelsey Allen, Charles Nash, Victoria Langston, Chris Dyer, Nicolas Heess, Daan Wierstra, Pushmeet Kohli, Matt Botvinick, Oriol Vinyals, Yujia Li, and Razvan Pascanu. Relational inductive biases, deep learning, and graph networks, 2018. URL https://doi.org/10.48550/arXiv.1806.01261.

[122] Petar Veličković. Message passing all the way up, 2022. URL https://doi.org/10.48550/arXiv.2202.11097.

[123] Yan Ma, Tobias Kind, Dawei Yang, Carlos Leon, and Oliver Fiehn. Ms2analyzer: A software for small molecule substructure annotations from accurate tandem mass spectra. *Analytical Chemistry*, 86(21):10724–10731, Nov 2014. ISSN 0003-2700. doi: 10.1021/ac502818e. URL https://doi.org/10.1021/ac502818e.

[124] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains, 2020. URL https://doi.org/10.48550/arXiv.2006.10739.

[125] Sunghwan Kim, Ryan P. Rodgers, and Alan G. Marshall. Truly "exact" mass: Elemental composition can be determined uniquely from molecular mass measurement at 0.1mda accuracy for molecules up to 500da. *International Journal of Mass Spectrometry*, 251(2): 260–265, 2006. ISSN 1387-3806. doi: https://doi.org/10.1016/j.ijms.2006.02.001. URL https://www.sciencedirect.com/science/article/pii/S1387380606000856. ULTRA-ACCURATE MASS SPECTROMETRY AND RELATED TOPICS Dedicated to H.-J. Kluge on the occasion of his 65th birthday anniversary.

[126] Mary Phuong and Marcus Hutter. Formal algorithms for transformers, 2022. URL https://doi.org/10.48550/arXiv.2207.09238.

[127] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus), 2020. URL https://doi.org/10.48550/arXiv.1606.08415.

[128] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. URL https://doi.org/10.48550/arXiv.1512.03385.

[129] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016. URL https://doi.org/10.48550/arXiv.1607.06450.

[130] Toan Q. Nguyen and Julian Salazar. Transformers without tears: Improving the normalization of self-attention. 2019. doi: 10.5281/ZENODO.3525484. URL https://zenodo.org/record/3525484.

[131] Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F. Wong, and Lidia S. Chao. Learning deep transformer models for machine translation, 2019. URL https://doi.org/10.48550/arXiv.1906.01787.

[132] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding, 2020. URL https://doi.org/10.48550/arXiv.1906.08237.

[133] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. Electra: Pre-training text encoders as discriminators rather than generators, 2020. URL https://doi.org/10.48550/arXiv.2003.10555.

[134] Rainer Breitling, Patrick Armengaud, Anna Amtmann, and Pawel Herzyk. Rank products: a simple, yet powerful, new method to detect differentially regulated genes in replicated microarray experiments. *FEBS Lett*, 573(1-3):83–92, August 2004. doi: 10.1016/j.febslet.2004.07.055. URL https://doi.org/10.1016/j.febslet.2004.07.055.

[135] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL https://doi.org/10.48550/arXiv.1412.6980.

[136] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014. URL http://jmlr.org/papers/v15/srivastava14a.html.

[137] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. URL https://doi.org/10.48550/arXiv.1711.05101.

[138] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020. URL https://doi.org/10.48550/arXiv.1802.03426.

[139] Noel M. O'Boyle and Roger A. Sayle. Comparing structural fingerprints using a literature-based similarity benchmark. *Journal of Cheminformatics*, 8(1):36, Jul 2016. ISSN 1758-2946. doi: 10.1186/s13321-016-0148-0. URL https://doi.org/10.1186/s13321-016-0148-0.

[140] Vishwesh Venkatraman, Jeremiah Gaiser, Amitava Roy, and Travis J. Wheeler. Molecular fingerprints are not useful in large-scale search for similarly active compounds†. *bioRxiv*, 2022. doi: 10.1101/2022.09.20.508800. URL https://www.biorxiv.org/content/early/2022/09/22/2022.09.20.508800.

[141] Jacob Townsend, Cassie Putman Micucci, John H. Hymel, Vasileios Maroulas, and Konstantinos D. Vogiatzis. Representation of molecular structures with persistent homology for machine learning applications in chemistry. *Nature Communications*, 11(1):3230, Jun 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-17035-5. URL https://doi.org/10.1038/s41467-020-17035-5.

[142] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. URL https://doi.org/10.48550/arXiv.2001.08361.

[143] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. URL https://doi.org/10.48550/arXiv.2112.10752.

[144] Yoshua Bengio, Salem Lahlou, Tristan Deleu, Edward J. Hu, Mo Tiwari, and Emmanuel Bengio. Gflownet foundations, 2022. URL https://doi.org/10.48550/arXiv.2111.09266.

APPENDIX A

Acronyms

BCE Binary cross-entropy

CE Cross-entropy

CID Collision-induced dissociation

DreaMS Deep representations empowering the annotation of mass spectra

ECFP Extended-connectivity fingerprint

FFN Feed forward neural network

GELU Gaussian error linear Unit

GNN Graph neural network

GNPS The global natural product social molecular networking

LC Liquid chromatography

LSH Locality-sensitive hashing

MLP Natural language processing

MoNA Mass bank of North America

MS Mass spectrometry

NIST National Institute of Standards and Technology

QED Quantitative estimation of drug-likeness

ReLU Rectified linear unit

RT Retention time

SSL Self-supervised learning

TMAP Tree MAP

UMAP Uniform manifold approximation and projection

Appendix B

Contents of enclosed CD

- `thesis` — contents of enclosed CD
  - `msml` — Python package for the thesis
    - `algorithms` — directory with the source code for algorithms
    - `data` — directory with the source code for data
    - `models` — directory with the source code for neural networks
    - `experiments` — directory with the source code for experiments
    - `utils` — directory with the source code for utility functions
    - `definitions.py` — definitons of global variables
  - `setup.py` — installation file for the `msml` package
  - `tex` — directory with the LaTeXsource code