Learning Metabolite Tandem Mass Spectra Predictors From Molecular Graph Structure

by

Fei Wang

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

Abstract

In the field of metabolomics, mass spectrometry (MS) is the most widely adopted method for identifying metabolites. Conventionally, metabolite identification involves matching the target mass spectrum against experimentally acquired reference mass spectral libraries. However, the limited coverage of these reference libraries has created a major bottleneck to this approach. In the past few decades, several alternative approaches have been developed to address this issue of limited coverage of experimental MS reference libraries. These include *in-silico* fragmentation methods, which are capable of generating reference mass spectra from chemical structures, and so can extend existing MS reference libraries with synthetic spectra. While traditional *in-silico* fragmentation methods rely on hand-crafted rules, many recent approaches use machine learning methods to extract MS fragmentation rules.

This dissertation extends a state-of-art machine learning process, Competitive Fragmentation Modeling (CFM-ID), which uses a learned model to simulate the MS fragmentation process that occurs in a tandem mass spectrometer. While CFM-ID is an important step forward from hand-coded rule-based approaches, it still is unable to produce sufficiently accurate MS spectra, therefore it cannot yet be seen as a reliable alternative to laboratory mass spectrometry. My primary research contribution is to extend Competitive Fragmentation Modeling methods by learning parameters from

the topological structure of a molecule. In the tandem mass spectrum prediction task, our models showed significant improvement compared to the original CFM-ID models across multiple data sets. Furthermore, we also developed several sampling methods that greatly reduced the computational cost of training the model, yet still surpassed legacy CFM-ID models by a significant margin in spectrum prediction tasks.

Acknowledgements

I would like to express my most sincere gratitude to my supervisor, Dr. Russell Greiner, for his guidance and endless patience throughout the journey. In addition, I am very grateful to my co-director, Dr. David Wishart, for his guidance in bioinformatics and mass spectrometry. Special thanks to Dr. Felicity Allen for laying the foundation for this work.

Many thanks to my colleagues, Robert Vega and Siyang Tian for their helpful discussions and suggestions during this research, Dr. Yannick Djoumbou and Dr. Maheswor Gautam, for sharing their chemistry and mass spectrometry expertise, and David Arndt for his help in supporting this work.

Many thanks to all my family and friends for their consistent patience, support and encouragement.

Last but not least, thanks for Compute Canada's Westgrid facility for providing countless hours of computational resources to make this work possible.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivations

Metabolomics is the scientific study of chemical processes involving small molecule metabolites. Metabolomics is playing a key role in many disciplines such as life sciences, food science, drug development, and medical diagnostics [9], [30].

The study of metabolites via metabolomics can help us to extend our knowledge of disease mechanisms and drug effects, as well as improve our ability to predict personal disease progression or variation in drug response phenotypes [20]. The identification and quantification of metabolites in human cells offer avenues for understanding, diagnosing, and managing human diseases; assessing disease risk factors associated with drugs, identifying potential toxins in the environment; and ultimately developing treatment options [22].

Metabolism is a general term for all the chemical reactions that occur in organisms [51], [77]. It includes two categories of chemical reactions: **catabolism**, which decomposes large molecules to gain energy, and **anabolism**, which consumes energy to synthesize components in cells. These complex biochemical processes directly contribute a range of organism activities including growth, and reproduction. They also allow organisms to maintain their structure and react to their external environment.

Metabolites are the intermediate and final small molecule products of metabolism. These small molecules are typically found within cells, tissues, and biofluids and have a restricted molecular weight (under 1500 Da [86]) [1]. As downstream products of genetic and environmental effects, metabolites provide sensitive measurements of the state of the upstream events or processes, including the effects of diseases, drugs, toxins, and the environment on the on cells, tissues or entire organisms. Through the overall analysis of metabolites, key insights regarding the physiological, pathological, and biochemical status of an organism can be gained.

The purpose of metabolomics is to provide quantitative and qualitative analysis of large numbers of metabolites in tissues or biofluids, which is often called metabolite profiling. The core challenge of metabolomics is the identification of metabolites. Common technologies for profiling metabolites include mass spectrometry (MS) and nuclear magnetic resonance spectroscopy (NMR) [9]. Between these two types of technologies, mass spectrometry has become the platform of choice [10] due to its sensitivity, reproducibility, and versatility. Mass spectrometers reveal the structure of input molecules by fragmenting their ionized forms into smaller ions and neutral particles and measuring the observed abundance of charged fragments grouped by their mass-to-charge ratio (m/z). The output of a mass spectrometer is called a mass spectrum or an MS spectrum (Figure 1.1). The x-axis of this graph is the mass-to-charge ratio, and the y-axis represents the relative intensity or relative abundance of the ions. The relative intensity is the ratio between the given peak height and height of the highest peak in the MS spectrum.

One of the most common goals of mass spectrometry-based metabolite profiling is determining the chemical identity of an unknown compound via its mass spectrum. Conventionally, given a mass spectrum of an unknown compound, the best way to determine its corresponding chemical structure is to find the matching spec-

[1] In perspective, mass of a Carbon-12 atom is 12 Da

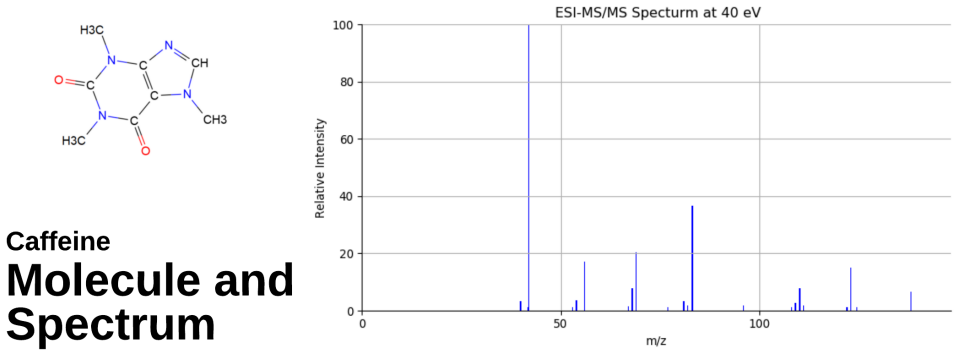


Figure 1.1: Diagram of a caffeine molecule and its ESI-MS/MS mass spectrum [67].

trum from a reference spectral library of known compounds (Figure 1.2). Currently, a mass spectrometry spectral database search is the fastest and also the most accurate method for chemical compound identification [4]. However, this only works when the molecule and its spectrum are included in the reference spectral library. The limited coverage and limited availability of these reference spectral libraries has become a major bottleneck to this approach [16], [79]. Moreover, since unreported metabolites are not included in any MS reference database, identifying novel or unknown metabolites using mass spectrometry has been considered as one of the most challenging issues in computational mass spectrometry [78].

To overcome this bottleneck, *in-silico* fragmentation methods have been developed to identify the chemical structure of an unknown compound with a previously unseen MS spectrum without having to directly query an experimentally derived reference spectrum database. These *in-silico* methods can be divided into two broad categories: (1) those that extract the chemical and physical characteristics of the input compound from its measured MS spectrum and then find the corresponding chemical structure with such characteristics from a compound database [27], [32], [45], [64], [65]; and (2) those that attempt to identify an unknown molecule

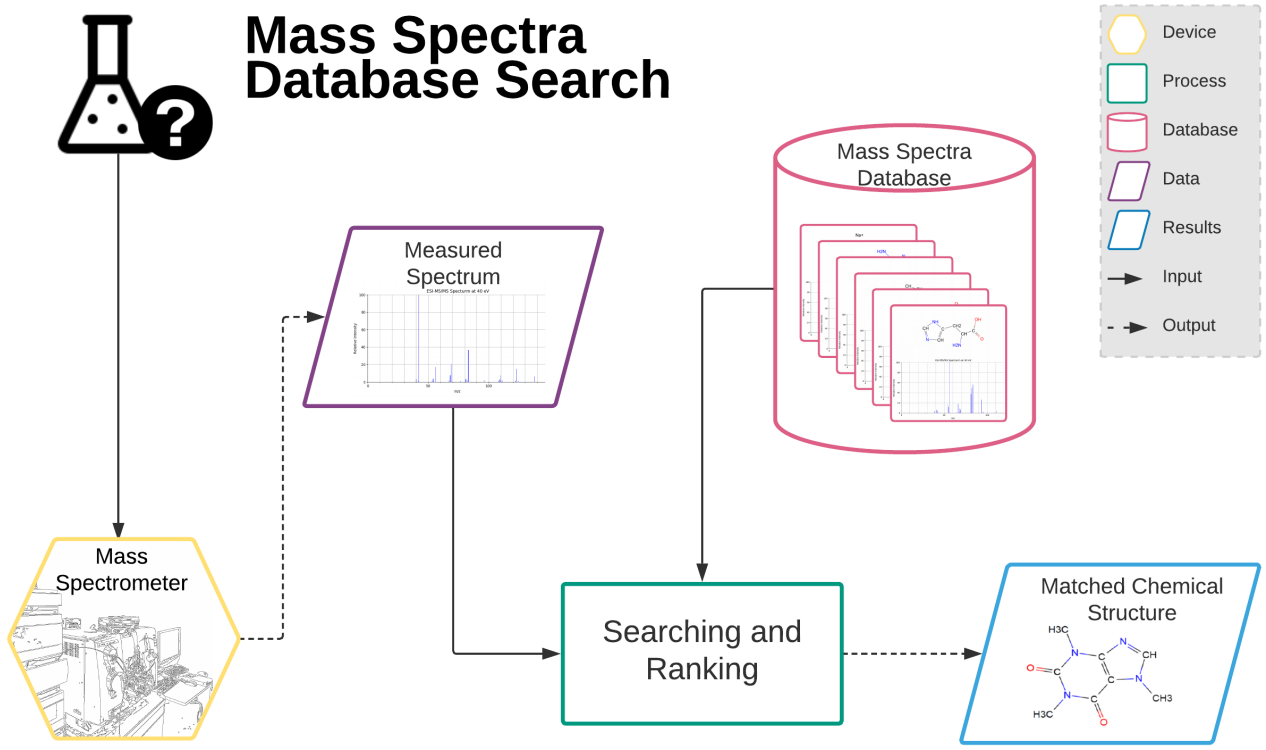


Figure 1.2: Diagram of a conventional MS spectra search.

by comparing its experimental spectrum with predicted candidate spectra that are computationally generated for molecules from a compound database [1], [2], [21], [60]. Regardless of the types of *in-silico* methods being used, the more accurate these methods can predict, the better the compound identification performance will be. Furthermore, since the number of known chemical compounds is much larger than the number of experimentally collected MS spectra (hundreds to thousands of times larger in terms of the number of compounds); *in-silico* fragmentation methods should be more capable of identifying new or unknown compounds than by searching through a small set of experimental MS spectra.

The state-of-the-art methods in this field are using a wide array of different techniques, ranging from rule-based expert systems to the latest deep learning algorithms [1], [2], [27], [44], [64], [65]. However, when many of theses methods were independently tested using the Critical Assessment of Small Molecule Identification (CASMI) challenge [63], including the four top such top algorithms, MS-Finder [80], CFM-ID [1], [2], MetFrag [60] and MAGMa+ [83], the results showed that these pure *in-silico* algorithmic approaches could only correctly identify about a quarter of the testing compounds without the help of meta data. [5]. This test shows that there is still significant room for improvement for all these *in-silico* methods.

In mass spectrometry, especially tandem mass spectrometry (MS/MS) a chemical structure transforms into a smaller chemical structure through the cleavage of one or more chemical bonds that links the two parts of structure together. Although the fragmentation mechanism of ESI-MS/MS has not been fully explained it is believed to be highly correlated to the strength of a chemical bond that describes how strongly any two atoms are joined to another, thus indirectly indicating the tendency toward bond cleavage. Over the years, various *in-silico* algorithms have used a range of different chemical and physical measurements as an approximation of a bond's strength, such as bond energy, bond dissociation energy, and cleavage activation energy [34],

[60], [74]. While accurately computing these measurements for an instance usually involves complicated quantum chemical computations, many researchers have found that similar chemical and physical properties can be estimated using the compound's structural information [17], [23], [35], [61], [90].

Inspired by these efforts, we developed a process that improves the existing CFM-ID method [1], [2] to learn a model that can more accurately predict the Electrospray Ionization Mass Spectrometry (ESI-MS) spectrum of chemical compounds. This new system learns its parameters from a detailed representation of the molecule's chemical bond topological. In contrast, this topology information was largely ignored by the original CFM-ID sapproach.

1.2 My Contributions

The main hypothesis driving this work is that mass spectral prediction, especially tandem mass spectral prediction for small molecule metabolites can be further improved by learning the spectral fragmentation model from the molecule's topological information.

My primary research contribution is the development of various methods to enhance and extend the original CFM-ID approach to small molecule MS spectral prediction. The proposed Connectivity Matrix Features (CMF) is a representation of a chemical bond's topological surroundings based on the adjacent matrix of the molecular graph (Figure 1.1). This CMF enables the new CFM-ID methods to learn their parameters from the topological characteristics of the input chemical structures. With regard to the spectral prediction task, our empirical results showed that the spectral fragmentation models learned by this new approach perform significantly better than the legacy CFM-ID models on ESI-MS/MS data sets. When measured by the Dice coefficient (a method for measuring the similarity of predicted and observed spectra), we saw approximately a $15-20\%$ improvement (depending on the datasets). Furthermore, in addition to the improved performance, our proposed model used far fewer parameters compared to the legacy CFM-ID models.

Moreover, we also developed several sampling methodsthat can be used to more efficiently train the model. These improvements led to significant computational cost reduction of (at least 10-fold) compared to the legacy methods. This greatly increased the scalability of this class of CFM-ID models.

1.3 Document Organization

The rest of Chapter 1 covers the related works in mass spectrometry and the necessary chemical background respectively. Chapter 2 describes the methods developed in this study, and Chapter 3 presents the empirical evaluation of the results of the modified CFM-ID methods. Finally, Chapter 4 discusses the implications of this study and future work. This document is closed off with a Glossary and Appendix.

I will close this first chapter with three more sections that provide a more detailed background on mass spectrometry (Section: 1.4) , more details on *in-silico* fragmentation methods (Section: 1.5), and a review of CFM-ID methods (Section: 1.6).

1.4 Mass Spectrometry

Mass spectrometry is a widely adopted and well established analytical technique for analyzing chemical compounds. It can reveal structural information about a molecule by measuring the mass-to-charge ratio of that molecule or the mass-to-charge ratio of its constituent fragments.. Given a purified molecule, its mass spectrum (mass spectra in some cases) is often sufficient to determine the partial structure of the molecule and sometimes even the identity of the metabolite.

The theoretical basis of mass spectrometry (MS) can be traced back to the end of the 1890s [30], [53]. In 1912, J.J. Thomson implemented the technique and used it to demonstrate the presence of Neon-22 (a rare isotope of Neon gas) in Neon-20 (the common form of Neon) samples, thus demonstrating the existence of isotopes [30], [53]. During the Second World War, the Manhattan Project used MS technology in the task of enriching uranium separation [47]. Throughout the second half of the 20th century, mass spectrometry has become increasingly widely used and far more widely available to scientific researchers interested in characterizing chemicals.

A mass spectrometer is a specially designed instrument used to determine the mass-to-charge ratio of charged or ionized molecules. An MS Instrument consists of three main components: an ionization source, a mass analyzer and a detector (Fig. 1.3, which all operate under high vacuum conditions). The chemical sample of interest enters the mass spectrometer through an inlet which is connected to the ionization source. The ionization source converts the neutral molecules into ions, which are the charged form of molecules or atoms. Once in the ionization source, the MS instrument then applies an electric field to accelerate them. During this acceleration phase, the ionized molecules can enter into a collision cell (found in tandem mass spectrometers), containing neutral gas molecules which collide with the ionized molecules leading them the break down into smaller ions and neutral

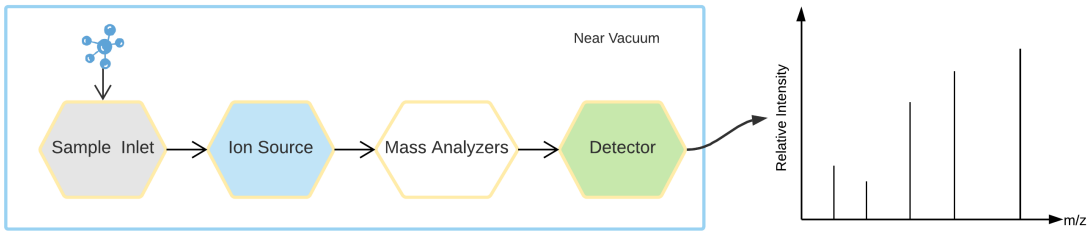


Figure 1.3: Diagram shows 4 major components of a mass spectrometer.

molecular fragments (called neutral losses). These ion fragments then proceed to the next component of the MS instrument, the mass analyzer, which uses electric (or magnetic) fields to separates the ions based on their mass-to-charge ratios. Finally, the ions reach the detector, which counts the ions of different mass-to-charge ratios. The mass spectrometer can only detect charged particles, uncharged particles cannot be directly observed through an MS instrument.

Mass spectrometers are usually categorized by their ionization source or mass analyzer. Most mass spectrometers use Electron Ionization (EI), Electrospray Ionization (ESI) or matrix assisted laser desorption ionization sources (MALDI). The most common mass analyzers used by mass spectrometers are quadrupole analyzers (Q), triple quadrupole analyzers (QqQ), time-of-flight analyzers (TOF) or Orbitrap mass analyzers. A more detailed discussion of ionization sources and mass analyzers can be found in the section 1.4.2 and section 1.4.3 of this chapter.

1.4.1 Chromatography

Chromatography is a standard pre-processing step used in mass spectrometry. It is a technique for separating or extracting pure compounds from complicated biological or chemical mixtures. This separation process is done by passing the mixture through a specially designed column containing either small particles or a specially designed liner. Mass spectra of pure compounds are much easier to interpret than

impure or mixed compounds. Gas Chromatography (GC) and Liquid Chromatography (LC) are the most widely used chromatographic methods in mass spectrometry. The former is usually used in conjunction with an Electron Ionization (EI) mass spectrometer while the latter is used in conjunction with an Electrospary Ionization (ESI) mass spectrometer.

1.4.2 Ionization Source

The ionization source is the first core component inside a mass spectrometer and its primary purpose is to convert input neutral molecules into charged ions. Common ionization methods include, but are not limited to, Electron Ionization (EI), Chemical Ionization (CI), Atmospheric Pressure Ionization (API), Matrix Assisted Laser Desorption ionization (MALDI) or electrospray ionization (ESI). ESI, MALDI and API are considered soft ionization techniques, while EI and CI are considered as "hard" ionization techniques.

Since this work only uses data collected from ESI mass spectrometers, we will not discuss EI, CI, API or MALDI methods further Electrospray ionization (ESI) creates charged molecular ions, also known as adducts, through spraying an aqueous solution of the chemical or chemicals of interest through a small metal capillary placed under a strong electric field. The sprayed solution exists at the end of the capillary as droplets and enters a heating chamber where the droplets travel toward the next MS component against a gas flowing in the opposite direction. The combined effect of the gas stream and the accumulated charge on each droplet eventually creates gas-phase ions. In particular, this spraying and evaporation process allows the molecule(s) to become ionized by picking up (or losing) electrons or protons.

Electrospray Ionization can produce both positive and negative ions. The $[M + H]^+$ correspond to a positively charged ions generated by adding a proton to the molecule of interest. The $[M - H]^-$ ions correspond to negatively charged ions gen-

erated by removing a proton from the molecule of interest. After being ionized and sent through the ionizer,the gas-phase ions eventually reach the end of the chamber and proceeds to the mass analyzer.

As noted earlier, Electrospray Ionization (ESI) is a soft ionization technique for small molecules is considered as a soft ionization method, as most ions are still largely intact when they enter the mass analyzer. If the parent ion remains intact and is sent through a simple quadrupole analyzer and is detected, then the total mass (or mass-to-charge ratio) of the molecule can be easily determined. If one knows the mass precisely enough, it is possible to narrow down the identity of molecule by comparing its mass to a list of candidate chemical structures sorted by their calculated masses. Unfortunately, this approach to identifying molecules based solely on their mass is often insufficient to uniquely determine what the molecule is. However, if the molecular ion (or parent ion) can be fragmented into smaller parts and the masses of those fragments determined, then it is possible to determine the structure (and even the identity) of the molecule with much greater confidence. When ESI is combined with tandem mass spectrometers (MS/MS), it is possible to fragment molecules, even using a soft ionization technique like ESI. Tandem mass spectrometers consist of multiple mass spectrometers or mass analyzers with a specially designed collision cell that allows the parent ions to collide with inert gas (argon or helium) molecules and fragment.

1.4.3 Mass Analyzer

The mass analyzer in a mass spectrometer separates ions by their mass-to-charge ratio. There are two fundamentally different mass analyzers that are of interest for the work described in this study: the Time-of-Flight (ToF) analyzer and the Quadrupole analyzer. As the name suggested, the ToF mass analyzer separates ions by measuring the time it takes for ions to travel over a well-defined distance. A ToF

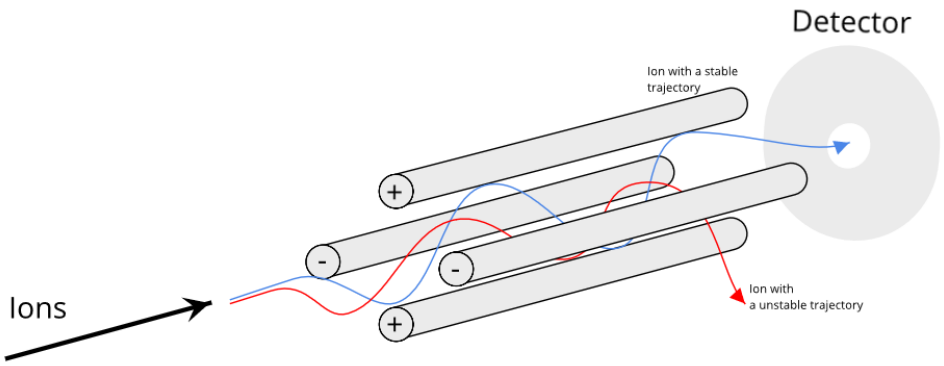


Figure 1.4: Diagram of a quadrupole mass analyzer.

analyzer first accelerates all ions simultaneously through the same electric field to ensure that each ion obtains the same amount of kinetic energy. In doing so, one can be certain that the ion travel speed depends only on its mass-to-charge ratio. For instance, as ions with a lower mass-to-charge ratio will fly slower, it takes them longer to reach the detector. Because all ions start at the same time, mass spectrometers using ToF mass analyzer are capable of fast data acquisition.

The quadrupole (Q) mass analyzer consists of four parallel electrodes that are evenly distributed around the ion flight path. During a quadrupole mass analyzer's operation, an oscillating radio-frequency field is created between the four electrodes, causing the ions to travel along helical trajectories (Figure 1.4). Only ions within a certain range of mass-to-charge ratio can pass through the mass analyzer and reach the detector, while the remaining ions are separated due to their unstable travel trajectories. This design gives the quadrupole mass analyzer a powerful mass selection capability. But because only ions with a given mass-to-charge ratio can pass through at a time, this type of device is much slower in data collection speed compared to its ToF counterparts.

1.4.4 Tandem Mass Spectrometry with Collision Induced Dissociation

Tandem mass spectrometry (MS/MS or MS^2) [48] as the name suggests, is a form of mass spectrometry where the input compounds are subjected to a sequence of two or more mass spectrometry analyses by two (or more) mass spectrometers placed in tandem [24], [30].

Most ESI tandem mass spectrometers contain a collision cell, filled with inert gas that allow molecular ions (also called parent ions) coming from the ionizer to collide with the gas atoms and to fragment into product ions. The first mass spectrometer (often a quadrupole type) allows users to select molecular ions to be fed into the collision cell. The second mass spectrometer takes the fragmented product ions and sends them to the detector. It is also possible to redirect the product ions back into the collision cell (or to another collision cell, depending on the instrument design) and to get further fragmentation of the product ions (i.e. product-of-the-product ions). This step can be repeated many times. If a mass spectrum consists only of the product ions and some remnants of the molecular or parent ion, it is called a MS^2 spectrum. If a mass spectrum consists of product-of-the-product ions, it is called a MS^3 spectrum. Therefore, a spectrum generated after multiple (n) passes through a collision cell is called a MS^n spectrum.

The MS data used in this work (for both training and testing the algorithms) are all ESI-MS/MS tandem mass spectra collected by on Q-ToF instruments. A Q-ToF mass spectrometer typically includes an ESI ionizer or source, a quadrupole (Q) mass analyzer, a fragmentation cell or collision cell, and a second ToF mass analyzer. As shown in Figure 1.5 , for a given chemical sample, molecules enter the instrument, become ionized, and then proceed to the quadrupole mass analyzer. Here, the ions undergo a selection process that allows only those ions within a given mass-to-charge ratio range to progress to the fragmentation device or collision cell. Selected ions

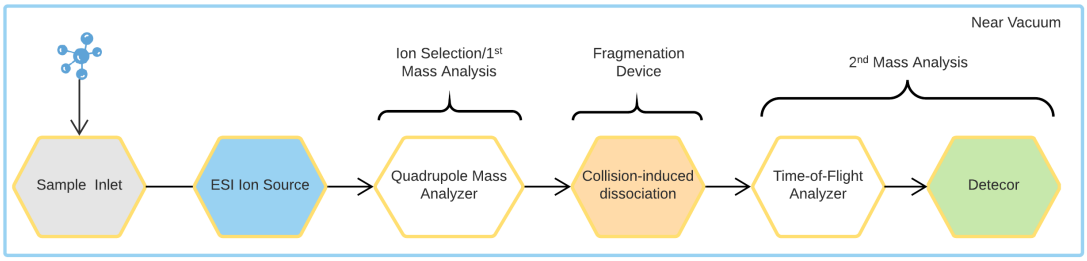


Figure 1.5: Digram of a Q-ToF tandem mass spectrometer.

(chosen by the instrument operator) then enter the collision cell that further breaks existing molecular ions into smaller fragment or product ions. Subsequently, those product ions arrive at the ToF mass analyzer where they are further separated and detected by the detector. The final output this instrument is then the ESI-MS/MS spectra.

The specific type of fragmentation device used by most Q-ToF instruments is a collision cell. Collision cells enable a process called collision-induced-dissociation (CID) which leads to ion fragmentation via the collision of ions with inert gas molecules. The fragmentation power of CID is controlled by the collision energy which is determined by the kinetic energy given to the ions before entering the collision cell. The collision energy in collision cells is usually measured in Electron Volts or eV. The higher the collision energy is, the more fragmentation will occur. Tandem mass spectrometry requires an appropriate amount of fragmentation to fully reveal the structural characteristics of the input molecule. Too much fragmentation will generate too many tiny, uninformative ion fragments generating uninformative MS spectra while too little fragmentation, will not generate enough fragment ions to fully characterize the parent molecule/ion of interest. As a result the spectra will lose the detailed characteristics. Often, to fully reveal the structural information of a parent molecule/ion, multiple tandem MS spectra collected at multiple collision energies

need to be collected. The tandem mass spectra used in this work were collected at three collision energy levels, 10 eV, 20 eV and 40 eV.

1.5 *In-silico* fragmentation methods

The main purpose of this thesis is to develop machine-learning techniques that can learn molecular fragmentation models from the topological information derived from chemical structures. This fragmentation information and the probabilities associated with this fragmentation can then be used to generate theoretical or *in-silico* Mass SpectrometryMass Spectrometry (MS/MS) is a mass spectrometry setup consists of two mass spectrometers in tandem spectra from the structure (known or hypothesised) of chemical compounds. *In-silico* fragmentation and *in-silico* MS/MS prediction offers a route to the identification of compounds when no reference MS/MS spectra of the compound exists. This section discusses a number of MS-based metabolite identification methods.

A number of methods for automatic MS-based metabolite identification have been developed and implemented using MS spectra database searches [15], [32], [38], [39], [69], [75]. Given an MS or MS/MS spectrum or spectra measured on the compound of interest, candidate spectra from the reference MS or MS/MS spectral database (which may have thousands of MS/MS spectra from thousands of compounds) are ranked according to how close each is to the query spectrum. Once ranked the molecules corresponding to the closest matched MS spectrum are returned as the answer to the search query. Note the number of returned possible molecules or MS spectral matches can be adjusted according to the user's need. Different spectral matching algorithms exist, but all are attempt to quantify how many and how closely the observed MS peaks match to the database MS peaks. These similarity measurements range from simple Euclidean distance, Absolute Value Distance [70] to more complicated probability-based matching methods [75].

In the case of ESI-MS/MS, Tautenhahn et al. [75] showed that a modified X-Rank algorithm can correctly identify 90 out of the 101 experimentally measured MS/MS

spectra collected from 23 different metabolites from a range of MS/MS instruments. In the cases of Electron Ionization Mass Spectrometry (EI-MS), Stein and Scott [70] queried 12,592 low resolution EI-MS spectra collected from 8000 compounds against the NIST-EPA-NIH Mass Spectral Database [71]. They reported that their algorithm achieved a 75% rank- 1 accuracy. Rank-1 accuracy is a measure which indicates the frequency with which the top ranked candidate was the correct match or correct answer. The best performing similarity metric that Stein and Scott used in this work measures a dot-product between two spectra represented as vectors. MS-database searching against a database of experimentally measured MS spectra only works if the database includes the target molecule. Therefore the success of such an approach depends heavily on the size and quality of the referential MS database.

In 2005, The Scripps Research Institute published an MS spectral database named METLIN [67]. METLIN is among the largest ESI-MS/MS databases containing MS/MS spectra of metabolites; as of July 2017, the METLIN database contains over 72,000 experimentally collected MS/MS spectra for over 14,000 different chemical compounds [49]. More recently, Guijas et al. reported that the METLIN database now has collected and analyzed MS/MS spectra for over 22,000 metabolites [26]from a wide range of organisms or samples. METLIN is now one of the largest repositories of MS data.

In 2007, a more human-centric metabolome database, The Human Metabolome Database (HMDB) [87], was published. The HMDB attempts to cover all metabolites found in the human body. At the time of this writing, HMDB [88] (v4.0; May 2019) contains 114,100 entries, with 22,198 experimental MS/MS spectra for 2265 compounds and 279,972 predicted MS/MS spectra for 98,601 compounds. Over the last decade, an increasing number of reference mass spectrometry databases have appeared or have become publicly available. These include MassBank, [31], Lipid maps, [76], MassBank of North America [42], and the National NIST/EPA/NIH

Mass Spectral Library [72].

While the efforts of the mass spectrometry community to build out spectral MS/MS databases is impressive, the number of compounds and MS spectra in these databases is far fewer than the number of known chemicals found in publicly accessible chemical databases such as PubChem [6], [12], and ChemSpider [54]. These databases include 96 million, and 67 million compounds respectively at the time of this writing, and are growing at a rate of 2-3 million compounds each year. The rate of expansion of chemical databases is far greater than the rate of expansion of MS spectral databases. For example, METLIN has only gained 12,000 new compounds in the past 5 years [26]. At this rate of expansion, it is likely that the number of known compounds will always outnumber the number of known MS spectra.

To overcome this limitation, a wide variety of *in-silico* methods were introduced over the years. Those methods can be loosely divided into two categories: spectra-to-structure prediction approaches (Section 1.5.1)), and *in-silico* fragmentation methods (Section 1.5.2).

1.5.1 *In-silico* Spectra to Structure predictions

In-silico spectra to structure prediction involves taking an MS (or MS/MS) spectrum and predicting what chemical structure this spectrum most likely corresponds to. An illustration of the concept is shown in Figure 1.6.

The first such MS spectrum to structure method described in the literature was, the DENDRAL project [19], [45]. Led by a team of interdisciplinary scientists at Stanford University in 1965, the DENDRAL project was one of the first applications of artificial intelligence (AI). It not only a pioneered many aspects of computational metabolite identification methods but also is considered as the "grandfather of expert systems" [45]. The original DENDRAL program, aka Heuristic DENDRAL, took experimental mass spectra as the primary input, then performed a heuristic search

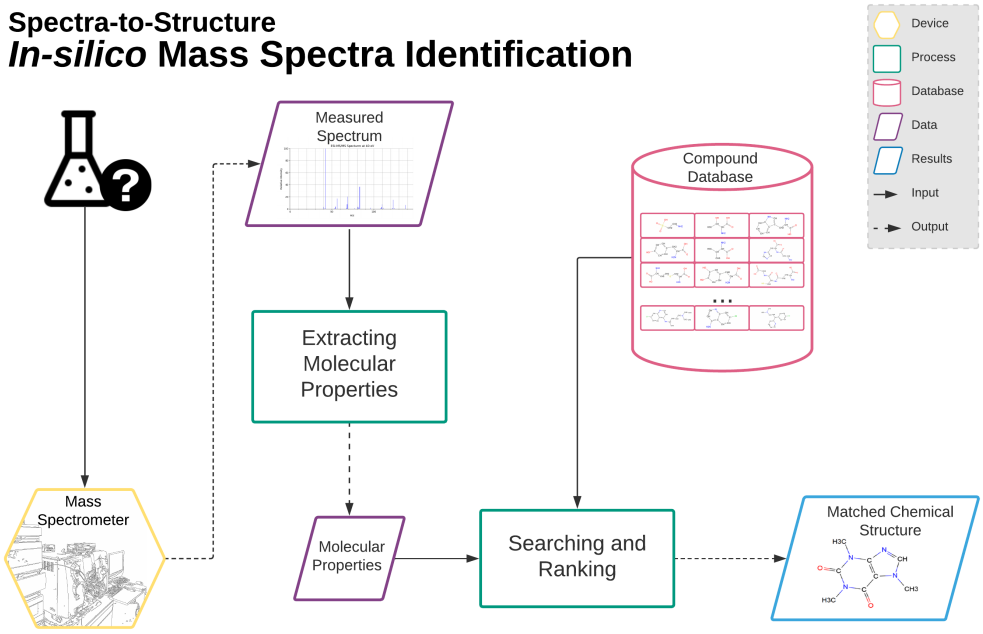


Figure 1.6: Diagram of Spectra-to-Structure prediction Methods.

guided by hand-engineered chemistry rules to explain the features of the spectra and associate them with a library of known chemicals. The final outcome of this program was a set of possible chemical structures that matched to the given spectrum. The performance task of this program was organized into three stages: *Plan*, *Generate*, and *Test*. The Plan stage found the constraints from an input spectrum that narrowed down the chemical search space. After that, the *Generate* stage created a large number of candidate solutions within the refined search space. Finally, the *Test* stag filtered and ranked candidates by certain chemical criteria. A later extension called META-DENDRAL [45] was a machine learning system that outputs a learned set of fragmentation rules for input mass spectrum and the corresponding molecules. Those rules could be later used in the *Plan* and *Test* stages of Herusitic DENDRAL as chemical constraint criteria.

Since the publication of DENDRAL other programs have appeared. However, most of these programs have focused on the easier task of spectra-to-chemical-class classification instead of the harder task of spectra-to-chemical-structure classification [11], [18], [33], [41], [82]. Chemical class is a loosely defined term, wherein molecules within the same chemical class all share the same pre-defined structural or functional criteria. These criteria may include the presence of a particular chemical substructure, a specific physical characteristic or a certain biological chemical property [32]. More recently, with the publication of a number of large publicly accessible mass spectral databases, a new breed of metabolite identification methods has appeared. In contrast to older attempts, such as DENRAL, the majority of these new methods use chemical fingerprints as an intermediate search space instead of directly searching in chemical space. Chemical fingerprints are hashed one-dimensional representations for chemical structures. For a given molecule, its chemical fingerprint can be extracted by examining a series of molecular properties (Figure 1.7). Each bit in this molecule-specific chemical fingerprint indicates the presence of a particular

characteristic of the compound's structure, those characteristics are similar to the chemical class. Chemical fingerprints often include the presence of specific atoms, chemical properties, or chemical substructures. Some chemical fingerprint variants also include encoding for topological features [43], [59]. Chemical fingerprints are widely used in querying chemical compound databases, especially in structure similarity searches [6]. Although chemical fingerprints are hashed, they have gained some popularity in machine learning applications.

Among the first works in this line of spectrum-to-fingerprint prediction is Heinonen et al.'s FingerID program [27], [65]. FingerID used ESI-MS/MS spectra to predict chemical fingerprints via a set of trained binary support vector machines (SVMs). For a given MS/MS spectrum, each SVM is responsible for predicting the value of a special bit in its corresponding molecule's chemical fingerprints. It then ranks the candidates by measuring the distance between the predicted fingerprint and the candidates' chemical fingerprint. This method was later extended by Shen et al. [64], who replaced the kernel SVMs used in the previous approach with multi-kernel SVMs. In addition, Shen et al.'s approach extracted a fragmentation tree from the spectra together with spectra as the input of multi-kernel support vector machines. A further extension of FingerID, called CSI:FingerID [14], combined the previous methods with 8 additional kernels and longer chemical fingerprints. Another work with a similar framework has been recently described by Brouard et al. [7]. This work replaced the set of SVMs by a single Input Output Kernel Regression (IOKR) method [8]. This method not only outperforms CSI:FingerID in terms of metabolite identification accuracy but also greatly improved both training and inference speed compared to earlier works. Lastly, Ludwig et al. [46] extended CSI:FingerID with a Bayesian network and achieved further improvements in compound identification rates. According to results of CASMI 2016 challenge [63], both IOKR [8] and CSI:FingerID [14] achieved impressive results, and ranked first and second in the

In-silico fragmentation only category. However, these spectra-to-structure prediction approaches lack the capability to help chemists and physicians to uncover and understand fragmentation mechanism of mass spectrometry.

1.5.2 *In-silico* Spectra Generation

Another set of approaches to compound identification involves *in-silico* spectral generation. This concept, shown in Figure 1.8 tackles the same problem of compound identification from the opposite direction of spectra-to-compound prediction.

As the name suggested, *in-silico* MS spectra generation attempts to simulate the compound fragmentation process in a mass spectrometer to some degree, and then to predict the spectrum of given input chemical structure. As illustrated in Figure 1.8, when applied to the compound classification task, an *in-silico* spectral prediction program first requires a database of known or hypothesized chemical structures. The program then predicts the MS spectra for every chemical structure in the databases. With the spectrum/structure database in hand, given a new spectrum, its structure candidates are compared, scored and ranked based on their closeness between the input MS spectrum and the corresponding predicted MS spectrum. This searching and ranking approach typically uses the same searching approach used for the conventional spectra identification in databases of experimentally generated MS spectra. Note that, if a set of candidates is given, only spectra for each candidate chemical structure are going to be predicted,This further reduces the size of the MS Spectra search space.

Like their spectra-to-structure prediction counterparts, most early *in-silico* spectra generation algorithms were rule-based systems that relied on manually-curated rules for fragmentation. Each rule answers a true or false question of whether a chemical bond with certain characteristics is going to break. For instance, a fragmentation rule can be as simple as "Aromatic rings never fragment".

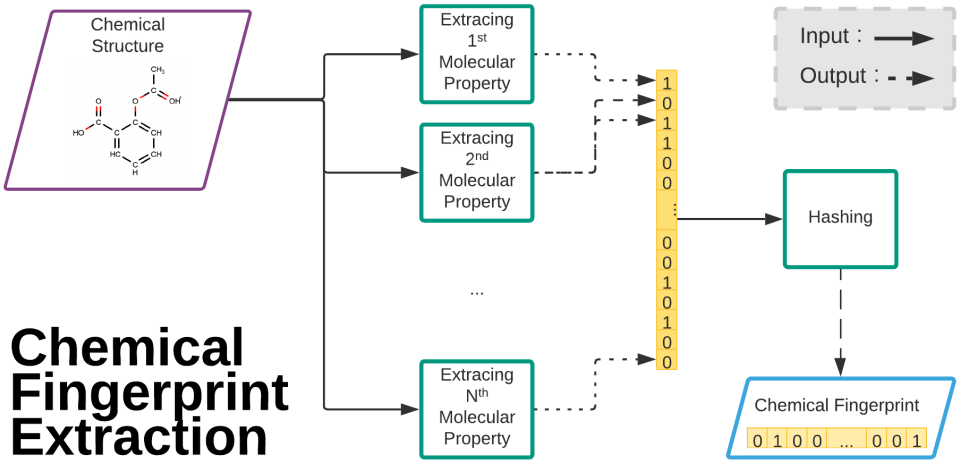


Figure 1.7: Diagram of chemical fingerprint extraction.

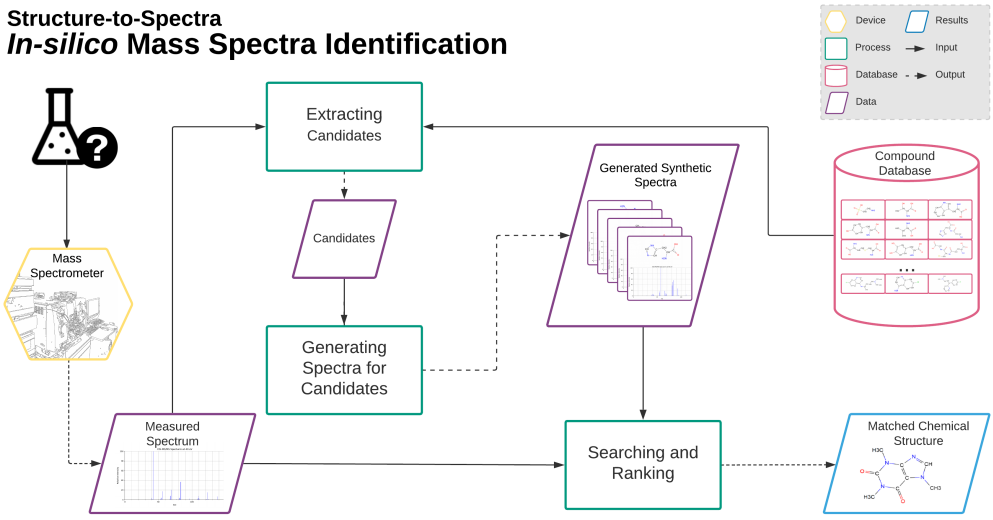


Figure 1.8: Diagram of structure-to-spectra prediction methods in compound classification tasks

Early rule-based programs were only capable of producing what are called "bar-code" spectra where all the peaks in the predicted spectra have the same height. These rule-based methods often used thousands of hand-engineered fragmentation rules, guided by experts' domain knowledge and experience. Their prediction capability largely depended on the number of rules in the system. In theory, the more rules, the better the prediction. However, the highest prediction precision cannot be reached by simply adding more rules. Schymanski et al [62] compared three widely used rule based solutions: ACD MS Fragmenter, Mass Frontier and MOLGEN-MS [36]. This study showed that when ranking candidate structures for 100 randomly selected EI-MS spectra, the best performance was achieved by "the simplest and quickest of the program and settings combinations".

Rather than relying on a hand-engineered, yet very complicated library of fragmentation rules, a number of newer methods use combinatorial approaches to fragment a chemical structure. This type of fragmenter maps the chemical structure to MS spectrum peaks via a collection of activities related to chemical bonds –a peak in the spectrum is a result of a sequence of bond cleavages with possible fragmentation rearrangement. Therefore, an MS spectrum can be generated by applying possible fragments of the original structure in a systematic manner.

Early combinatorial fragmenters [28], [29] enumerated all possible fragmentation in a purely systematic, recursive, and exhaustive way. However,this naive approach significantly limited the method to be used only in handling relatively small input molecules. Later works such as LipidBlast [37], MetFrag [60], [89], MIDAS [84], and MAGMa [58], performed combinatorial generation under the guidance of a heuristic function. In particular, these tools calculate and assign a heuristic "cost value" to each chemical bond for every input compound. After this cost assignment is made, then fragments are generated under the guidance of those heuristic values, which prioritize some fragmentation events over others. With a branch-cutting mechanism,

these methods do not need to go through all combinatorially possible fragments for a given chemical structure. Therefore, they are fast enough to handle more complicated input both in terms of the size of input molecular as well as the numbers of input candidates. However, like all other heuristic-based algorithms, choosing a suitable heuristic function to compute the "cost value" remains a challenging problem. Over the years, a wide array of heuristic functions have been proposed and applied to this domain, ranging from the functions tracking the types chemical bonds to more complicated bond-dissociation energy functions. To overcome the limitation of hand-picked heuristic functions, Kangas et al. [34] proposed the development of a data driven fragmenters for lipids. This method used a neural net within a kinetic Monte Carlo simulation to provide more accurate estimate of bond dissociation energies. Compared to the previous methods, this work was able to output more realistic-looking spectra where the predicted peak intensity are continuous values in contrast to the uniform height produced by its predecessor.

With regard to spectra prediction of ESI-MS/MS spectra, the state-of-art work in this area is the competitive fragmentation modeling method [1], [2] also known as CFM-ID. This method has also been later extended with rule-based enhancement to better handle lipids [13]. A more detailed review of the CFM-ID can be found in Section 1.6.

Most recently, Jennifer et al. [85] proposed a bi-directional neural network model for predicting EI-MS spectra. Their model was trained on the NIST/EPA/NIH Mass Spectral Library v14 [71] and evaluated on a replicate set from the same data library. In spectra classification tasks, their approach had an accuracy of 77% when only considering rank-1 prediction, which surpassed reported accuracy of the EI-MS version of CFM-ID [2]. Moreover, this new method also achieved an impressive speed-up in classification time. It was determined to be a million times faster than CFM-ID. However, unlike CFM-ID, this approach is only able to predict MS spectra

with integer precision at the mass-to-charge ratio. In another recent development, Guijas et al [26] mentioned they had devised a way to reverse the input-output kernel regression approach used in CSI:FingerID [14] to produce predicted ESI-MS/MS spectra for the METLIN database. However, the details of this work have not been published.

1.6 Competitive Fragmentation Modeling

For ESI-MS/MS, the Competitive Fragmentation Modeling (CFM-ID) method learns its parameters from input molecules and their ESI-MS/MS spectra. It uses the assumption that input molecules are in their most common isotope forms. Therefore, all CFM-ID predicted spectra are free of isotope peaks. While there is an extension of the CFM-ID for predicting isotope spectra, however it is beyond the scope this work.

1.6.1 Transition Model and Observation Model

The CFM-ID method models the fragmentation process inside mass spectrometer as a stochastic and homogeneous Markov process. More formally, given a molecule as input and its molecular or parent ion f_0, this process is defined as a fixed-length sequence of Fragment States F_0 to F_d. Here F_0 and F_d denote the initial state and final state respectively. Each state F_i consists of all theoretically possible ions fragments as described a set: $F_i = \{f_e^i\}$. For a fragment state $F_i = \{f_e^i\}$, next state F_{i+1} includes subsequent fragments produced from F_i. The process of generating fragments in F_{i+1} from F_i is guided by a fragmentation graph, which is a rooted directed graph that starts with the molecular ion of the input compound. The directed graph's vertices represent possible fragments from the input compound. And the edge connects from fragments f_e^i to f_e^{i+1} indicates a fragment transition $T(f_e^i, f_e^{i+1})$ from $f_l \in F_i$ to $f_e^{i+1} \in F_{i+1}$ is possible. Furthermore, a probability value $\rho = Pr(f_e^i \rightarrow f_e^{i+1}$ is assigned to each transition. This probability represents how likely fragment f_e^i will turn into f_e^{i+1} during the fragment state transition process. An example of this fragmentation scheme can be found in Figure 1.9, using acetic acid as an example. The top half of the figure shows a 2-depth fragmentation graph created for its $[\mathrm{M} + \mathrm{H}]^+$ ion, and the bottom half demonstrates a 3-step Markov process for

fragmenting this compound. Note that the sub-sequence fragment state contains all the fragments that have been seen in its predecessors, and it requires an $n+1$-step Markov process to cover all possible fragments from a n-depth fragmentation graph.
.

Starting from the molecular ion, the fragmentation graph is created systematically and recursively by CFM-ID. For any vertex in this graph, CFM-ID first adds a self-pointing transition (persistence transition) to allow the current structure to remain unchanged during the fragmentation state transition. It then generates the rest of each vertex's direct children vertices through iterative chemical bond cleavage.

Using the aspirin ion as an example, as illustrated in Figure 1.10, in a one-bond cleavage case, CFM-ID selects and disconnects a chemical bond in the current chemical structure, and then uses the results of this chemical bond breakage to create child vertices. In the case of a ring structure cleavage, it chooses a pair of bonds to break, rather than just select a single bond in contrast to normal one-bond cleavage cases. The CFM-ID approach assumes no mass or charge will be lost in the event of bond cleavage, and a fragmentation event always results in exactly two fragments. This rule effectively breaks any given ion into two fragments, the charged one becomes the one with children vertices, while another is a neutral loss. Because of the different possible positioning of the unpaired valence electron, more than one possible child fragment can be generated from the same bond dissociation event.

The CFM-ID program assigns a tendency value $\Theta_{i,m}$ to each possible transition $T(f_l, f_m)$ which represents how likely a particular transition occurs. When applied to the spectra prediction task, the CFM-ID program computes a break tendency for each transition through learned parameters and the transition's feature vector. It then applies those break tendency values to determine the transitional probability between fragments. For each vertex, a modified softmax function is used to map the break tendency values into conditional probabilities, as shown in Equation 1.1. The

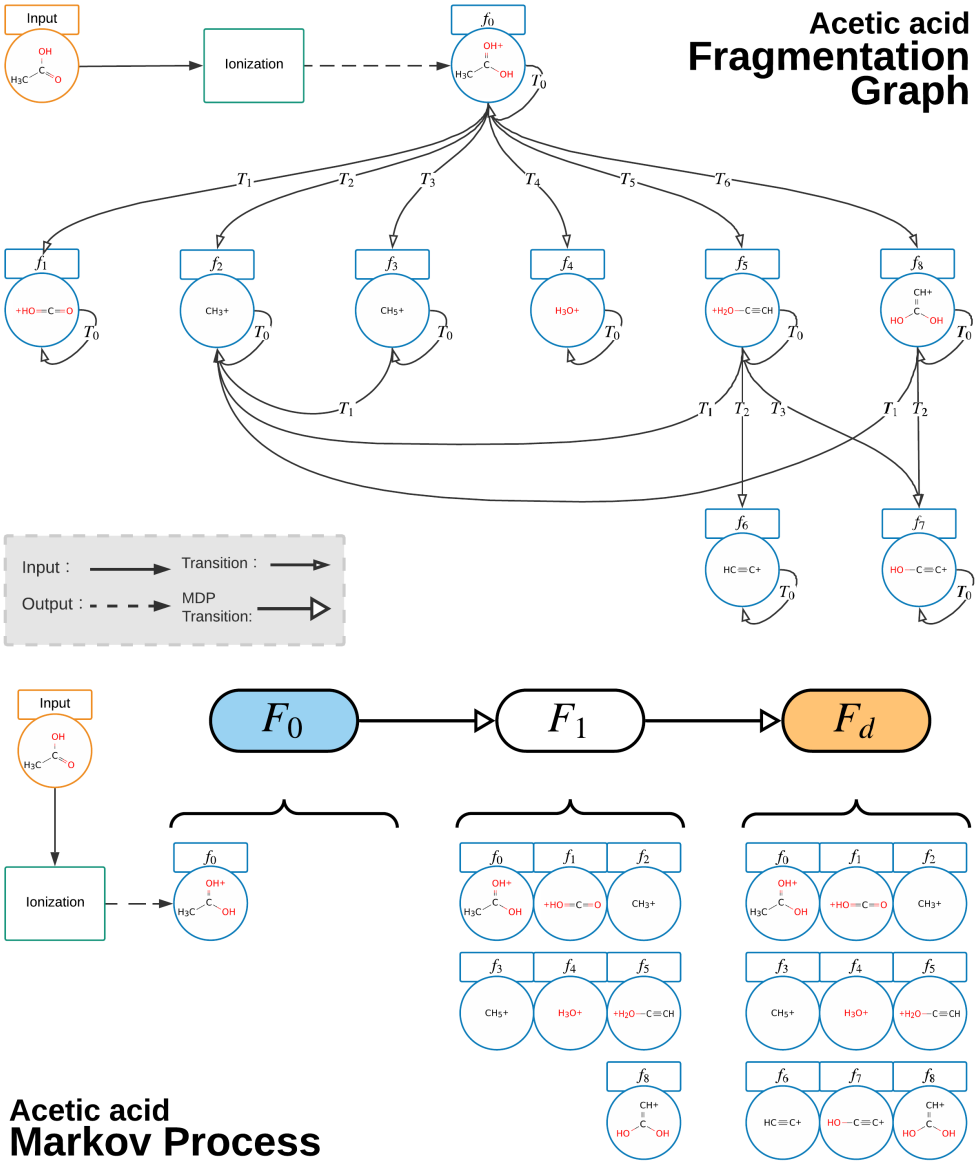


Figure 1.9: A fragmentation graph and a Markov process for the acetic acid $[M+H]^+$ ion.

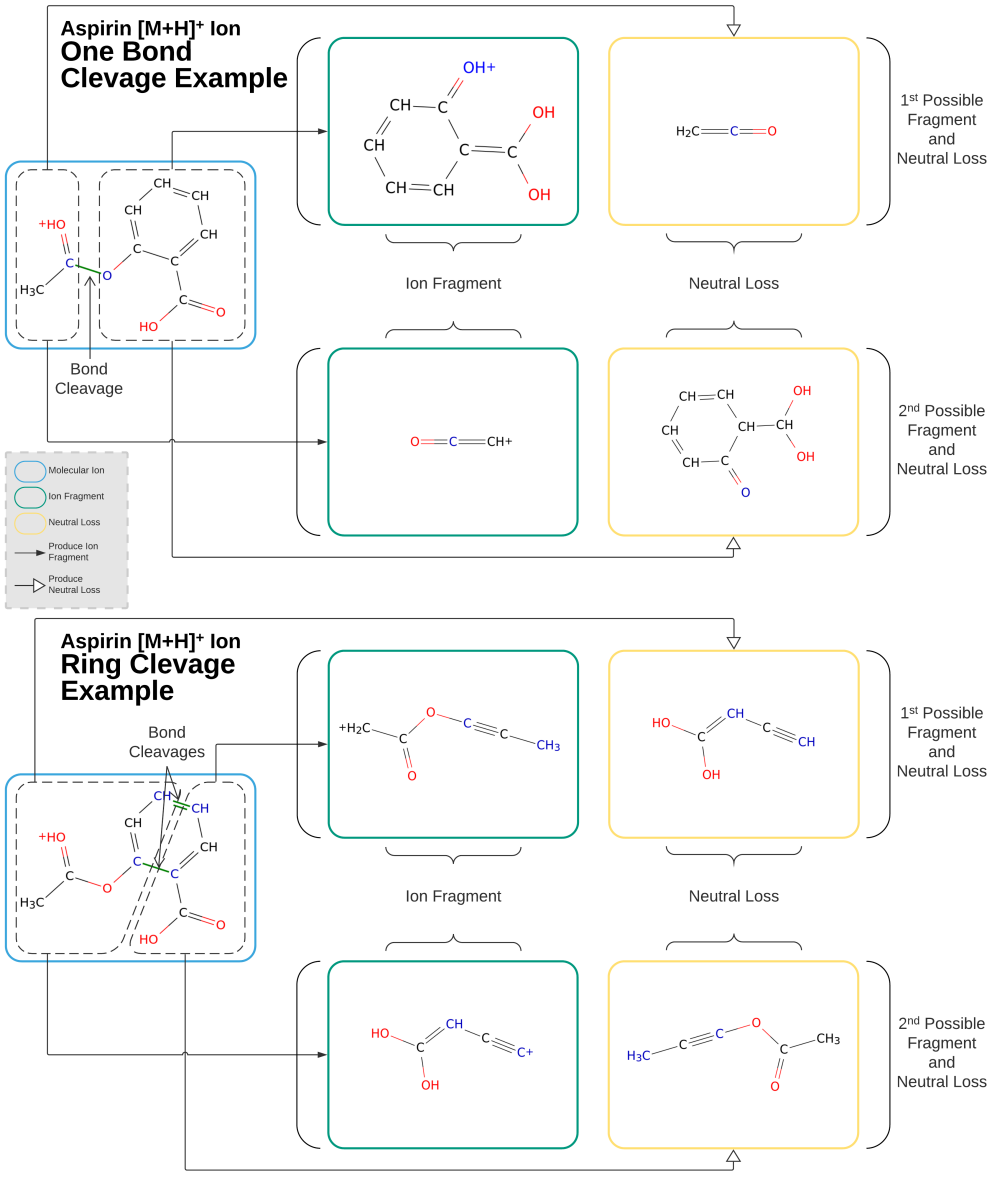


Figure 1.10: One-bond cleavage and ring cleavage examples for the Aspirin $[M+H]^+$ ion. When describing molecules, vertices represent atoms, and edges represent the chemical bonds. The bond types are represented by the number of lines on each edge. The bond breaks are coloured in green, and root atoms are highlighted in blue.

break tendency value of the persistence transition is fixed to 1. This softmax setup enables the CFM-ID model to capture the competitive nature of the fragmentation process as seen in real mass spectrometers.

$$\rho\left(f_i \rightarrow f_j \mid f_j \in Child(f_i)\right) = \begin{cases} \dfrac{\exp\theta_{i,j}}{1+\sum_k \exp\theta_{i,k}} & : f_i \neq f_j \text{ and } f_i \rightarrow f_j \text{ is possible} \\ \dfrac{1}{1+\sum_k \exp\theta_{i,k}} & : f_i = f_j \\ 0 & : f_i \rightarrow f_j \text{ is not possible} \end{cases} \tag{1.1}$$

The final fragment state F_d contains all theoretically possible chemical structures that can be observed and measured in a mass spectrum of the input compound. In the performance time, CFM-ID treats a spectrum as a mixture of Gaussian distributions where mass and intensity values of peaks are treated as the means and weights of those Gaussian functions. Hence, CFM-ID generates a predicted spectrum from the F_d by computing marginal probabilities of each mass-to-charge ratio.

To model the relationship between a predicted spectrum and experimental mass spectrum, CFM-ID employs a narrow Gaussian distribution model to measure the closeness between a predicted peak and an experimental peak. This observational model is centred around the mass of predicted peak, and its variance is set to address the difference in spectrometer's mass accuracy. For instance, given a measured peak $P_{measured}$ and a predicted peak $P_{predicted}$, and their mass difference is $\Delta\mathrm{m}, = Mass(P_{predicted}) - Mass(P_{measured})$. The probability of $P_{measured}$ and $P_{predicted}$ are a matching pair can be determined by a observational function shown as Eq. 1.2.

$$\mathrm{g}\left(\Delta\mathrm{m}, \mathrm{F_d}; \sigma\right) \quad = \quad \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(\frac{\Delta\mathrm{m}}{\sigma}\right)^2\right\} \tag{1.2}$$

1.6.2 Parameter Estimation

For a given data set χ of molecules, with both the chemical structure and their associated MS/MS spectra, CFM-ID pre-possesses each spectrum as follows, it first composes the relative intensity value of each peak into a range $[0, 100]$ and then normalizes them such that $\sum_{(m,h)\in x} h = 100$. Those values are treated as the frequency of each mass-to-charge ratio. During its learning phase, CFM-ID seeks the parameters w that optimize the cost function [2], Eq.1.3, where w is the learned weights of the model, Child $(\mathrm{f_i})$ is the set that contains all children of fragment f_i including f_i itself, ρ and g are transition functions (Eq. 1.1) and observation functions (Eq. 1.2) respectively.

$$L(\chi, w) = \prod_{x\in\chi} \prod_{(m,h)\in x} \sum_{F_1\in Child(x)} \rho(x, f_1; w) \sum_{f_2\in Child(f_2)} \rho(f_1, f_2; w) \ldots \sum_{f_d\in Child(f_{d-1})} \rho(f_{d-1} \to f_d; w) g(m, f_d; \sigma)^h \tag{1.3}$$

Since the cost function cannot be solved in a closed form, the parameter estimation is done by applying an Expectation Maximization (EM) algorithm [50] to the model.

The binary feature vector used by the original version of the CFM-ID algorithm consists of a chemical feature component and a quadratic feature component. For each fragment transition, the CFM-ID method extracts the former from the molecular structure and then creates the latter by using a quadratic polynomial function on the chemical features. For every pair of features in a chemical feature vector, the quadratic feature component has a feature indicating whether they occur together. If necessary, this quadratic polynomial method can be further expanded to use a higher-order polynomial function. However, it brings the obvious disadvantage that leads to a rapidly increasing number of dimensions. To learn a better representation

from the raw features, Allen et al10. [2] proposed an advanced method to address this problem that incorporated a neural network into the M-step of the EM method and a CFM-ID specific back-propagation method for learning the model.

There are other extensions that can be applied to enchance the CFM-ID model. For details of those methods please refer to the original publications [1], [2]. This work extends from the basic CFM-ID model with its neural network enhancement. For the purpose of this thesis, the term competitive fragmentation modeling (CFM or CFM-ID) refers to this specific machine learning approach in both learning and performance time.

1.6.3 Structure Feature Representation in CFM-ID

For a given molecule, CFM-ID extracts a feature representation from its molecular graph (Figure 1.11) which corresponds to a labelled graph whose vertices and edges are used, respectively, to represent the atoms and chemical bonds of a compound. Labels assigned to vertices record the atom types and labels assigned to edges record the bond types. In the context of CFM-ID, all hydrogen vertices all removed from the graph, only non-hydrogen atoms are taken into consideration.

Recall that in the context of CFM-ID, a fragment transition can involve up to two bond breaks. For each bond break, CFM-ID extracts a feature vector fv and then concatenates those vectors into a *fragment-transition-feature-vector* FV. We denoted these two *bond-break-feature-vectors* as fv_1 and fv_2 respectively. In the case of a one-bond breakage fragment transition, only fv_1 is used while fv_2 is populated with zeroes. In the case of a ring cleavage fragment transition, both fv_1 and fv_2 are populated with meaningful features.

A bond-breakage-feature-vector describes the bond cleavage event using the following three parts: 1) the chemical bonds that break, 2) the related structure on the ion fragment, and 3) the related structure on the neural fragment. The dissociated

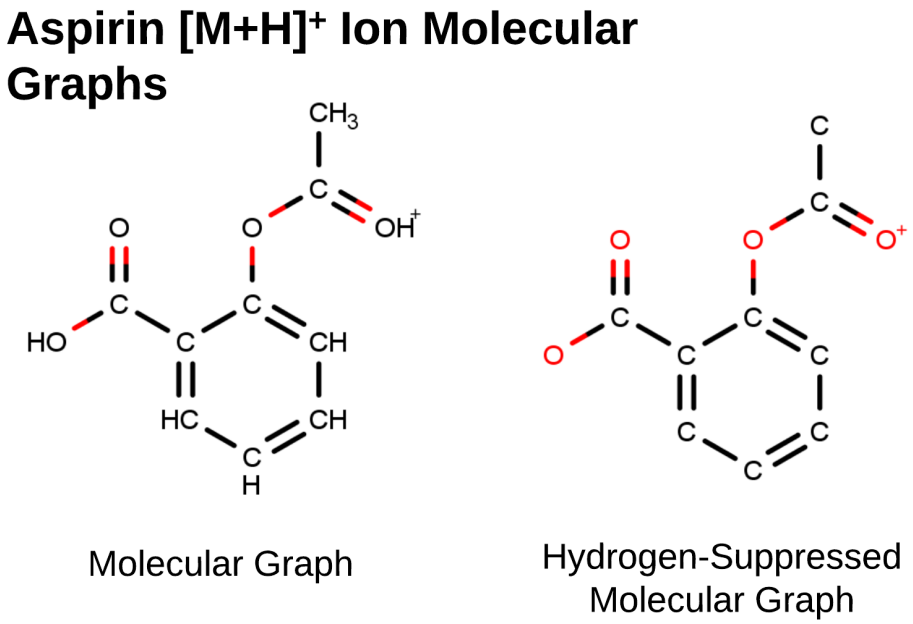


Figure 1.11: Molecular graphs for the Aspirin $[M+H]^+$ ion. Vertices represent atoms, and edges represent chemical bonds. The bond types are represented by the different number of lines on each edge. The graph on the left is the original molecular graph of the Aspirin ion. The right graph has all hydrogen vertices deleted, it is called a hydrogen-suppressed molecular graph.

bond is an important yet simple element to represent. For a given chemical bond, there are only two pieces of information needed: the bond order and the types atoms attached to it. The more interesting part is to represent two fragments after the bond break. For ease of description, we define the atoms involved in the bond dissociated as root atoms or roots. Therefore, the term *Ion Root* and *NL Root* refers to the root atom of an ion fragment and neural loss fragment respectively.

The characteristic of chemical structures is mainly captured by the following two one-hot encoded features.

- *Break Atom Pair features*, is a binary vector that indicates *Ion Root* or *NL Root* atoms types. Atom types can be one of Carbon, Nitrogen, Oxygen, Phosphorus, Sulfur or others [1].

- *Ion and NL Root Path features*, is a binary vector whose i^{th} bit indicates whether a particular atomic sequence occurs in a path that begins with either *Ion Root* or *NL Root*. AMore specifically, an atomic sequence is defined as a particular atom type pairs or tuple. For instance Carbon-Carbon pair is one atomic sequence , and Carbon-Carbon-Carbon is another. Atoms types of interests are Carbon, Nitrogen, Oxygen, Phosphorus, and Sulfur; all other heavy atoms are classified into the Other type [1].

Figure 1.12 provides a simple figure to illustrate the feature extraction process from a fragment transition $T(f_0, f_1)$ for fragment f_0 to f_1. First, the *Break Atom Pair features* are computed from the green chemical bond located within f_0. In this case, a carbon and an oxygen atom are attached to the bond, therefore the bit corresponding to this Carbon-Oxygen pair is set to 1 while everything else remains at 0. Then, the *Ion and NL Root Path features* are extracted from the ion fragment f_1 and the neutral loss fragment. This feature only covers a part of each fragment that starts from the root atom that is labelled and highlighted in blue. In this case, a

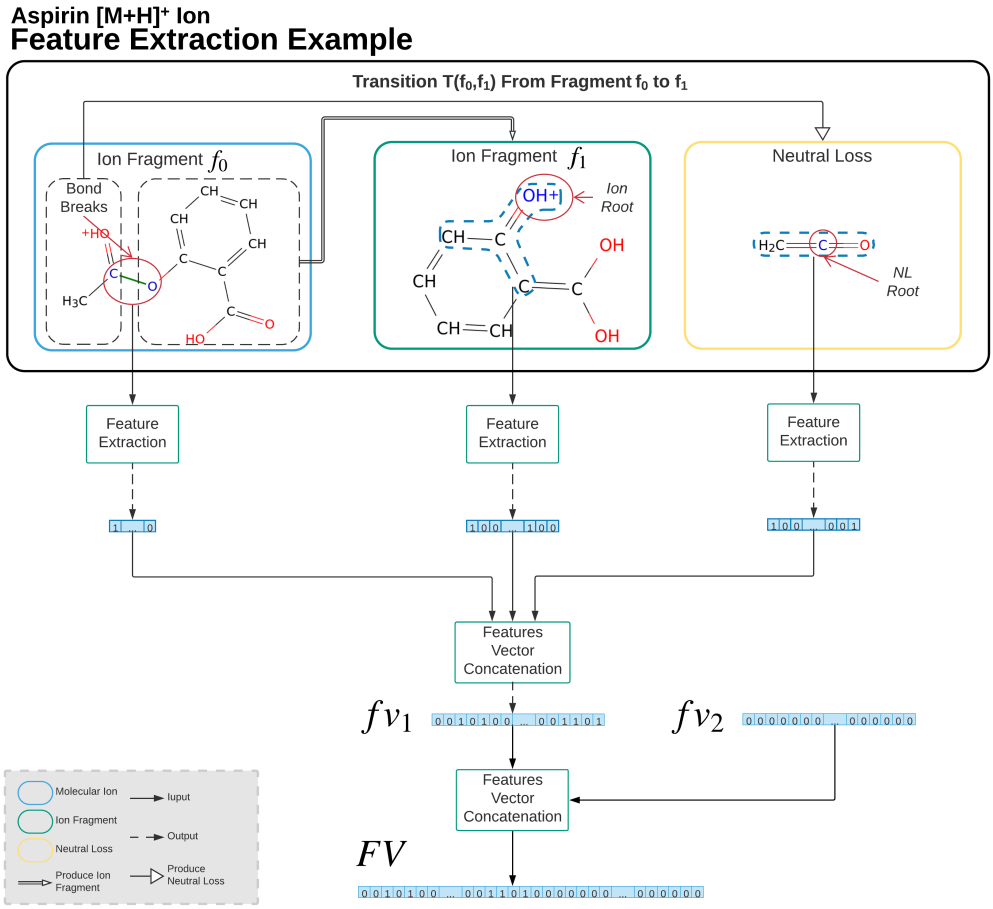


Figure 1.12: CFM-ID chemical feature extraction of a fragment transition. Chemical feature vectors are extracted from each part of this transition, then concatenated into the full feature representation. Only the bond-breakage feature vector fv_1 is populated while fv_2 is the all zero vector in this case.

blue-dash area highlights atoms within a 2-bond radius from the root atom within its respected fragment. Although the ion fragment f_1 has two different paths starting from the root atom, the atom sequences of each path are the same, both are an Oxygen-Carbon-Carbon tuple, therefore the *Ion Root Path feature* only has one bit set for this tuple. On the neutral loss fragment part, *NL Root Path feature* has a bit set for a Carbon-Carbon pair and another for a Carbon-Oxygen pair. Finally, all three separated feature vectors are concatenated to form the feature vector of this transition.

On the positive side, these two features are easy to calculate and easily avoid the feature mismatch caused by graph isomorphism. That is, features extracted by these methods are invariant to a vertex index change in the molecular graph. However, their limitations are obvious. Since both the bond order and topology information are discarded, even drastically distinct structures will end up with an identical feature representation. Figure 1.13 shows an example using 3,4-Diethylhex-1-ene and decane which are two structurally very different compounds. The former has a large branching factor, while the latter is a carbon chain without branches. When breaking the center bond of each molecular ion, even though the product ion and neutral loss fragments are structurally very different, both transitions show on the figure share the same feature vector: a bit for the Carbon-Carbon pair in terms of Break Atom Pair feature, and a bit for the Carbon-Carbon-Carbon tuple in terms of *Ion and NL Root Path feature*. Detailed topology features such as bond order, bond location, as well as the branching factor of each vertex have, unfortunately, been discarded using this representation.

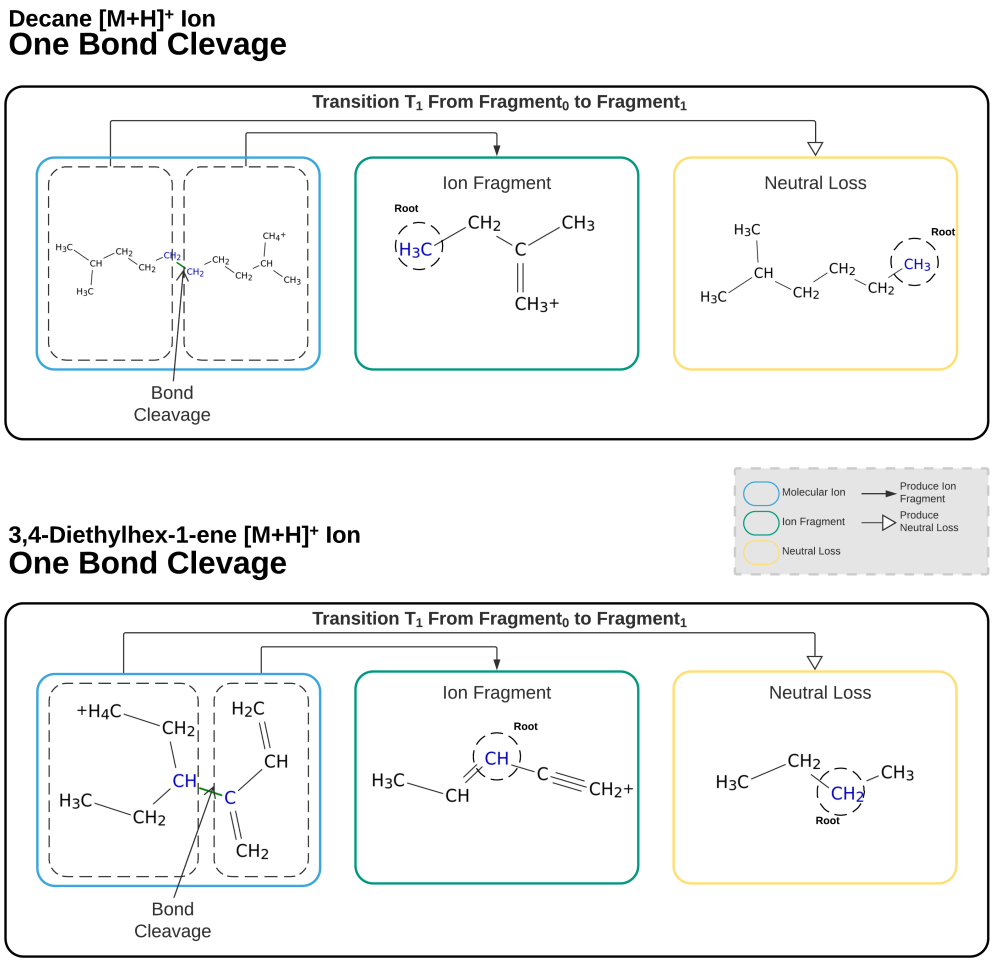


Figure 1.13: Fragment transitions from the Decane $[M + H]^+$ Ion and the 3,4-Diethylhex-1-ene Decane $[M + H]^+$ Ion. Although two transitions are formed by structural different chemical components, their feature representations are still the same. Start from the root atom of each fragment (highlighted in blue), each fragment has the presence of the Carbon-Carbon pair and the Carbon-Carbon-Carbon tuple.

Chapter 2

Methodology

This chapter presents the proposed remodel of ring structure cleavage (Section 2.1) and feature representations (Section 2.2) which we named as Connectivity Matrix Features, in CFM-ID. This chapter also describes several sampling methods (Section 2.3) used to boost training speed.

2.1 Sequential Ring Breakage Modeling

The first modification we applied to the original CFM-ID model was a re-modeling of the a ring breakage process as a *two-step sequential process*. Recall that CFM-ID models each fragment transition with up to two bonds, it extracts features from each bond cleavage and then concatenates two individual *bond-break-feature-vectors* fv_1 and fv_2 to a single feature vector FV. In a one chemical bond cleavage case, it fills the fv_1 with meaningful features and fv_2 with zeros. In a ring breakage case, the feature extraction process will populate both fv_1 and fv_2. However, the order of the two feature vectors after concatenation is determined in a first-come-first-serve basis. Therefore, for a ring beak transition, the old CFM-ID methods can extract two distinct feature representations. Since the second feature vector in the case of a non-ring breakage is always filled with zeroes, the parameters learned by those cleavages can not be fully shared with its ring break counterparts. Furthermore, this setup effectively increases the feature vector size by a factor of two. Thereby, it increases the computing resources required during training and prediction.

To address these issues, we re-modelled the ring break transition as a sequence of two individual one-bond fragment transitions. This idea removed the need to include a second bond feature vector from the original model. Figure 2.1 provides a simple comparison between the old CFM-ID ring model and the sequential ring breakage model. Given the same ion fragment f_0, the legacy CFM-ID approach created the subsequent fragment f_1 and its neutral loss by breaking two bonds highlighted in green. The sequential ring cleavage model breaks the only bond from f_0 and created an intermediate fragment f_{inter} with no neutral loss. It then produces the same fragment f_1 and neutral loss by breaking anther chemical bond.

The fragmentation graph generation process for a given molecule remains largely unchanged. For each vertex in the graph, our proposed method creates its child

vertices in the same way as the legacy CFM-ID method except the ring structure breaks. Given a fragment f_0 that has a ring structure, the proposed method first breaks one bond on the f_0 to create the intermediate fragment f_{inter}. To reduce the computational complexity, an original ring bond of fragment f_0 is then disconnected on fragment f_{inter} to create fragment f_1.

On top of this change, during graph depth calculation (and to compensate for the extracted depth introduced by this change), our methods treat a ring break sequence as a single vertex in the fragmentation graph. Therefore, the existing depth limitation can be left unchanged with this re-modeling. Moreover, during both the training and prediction process, our method sets the persistence transitional probability of the intermediate fragment to zero, which means unstable fragments will never become end outcomes. The sequential ring break modeling can be further extended to handle fragment transitions involving more than two bonds. However, ring breaks are already an unusual event, the chance of a triple or higher number of bond break at once is even rarer. To simplify our model, and the associated computations, we decided not to cover these rare cases.

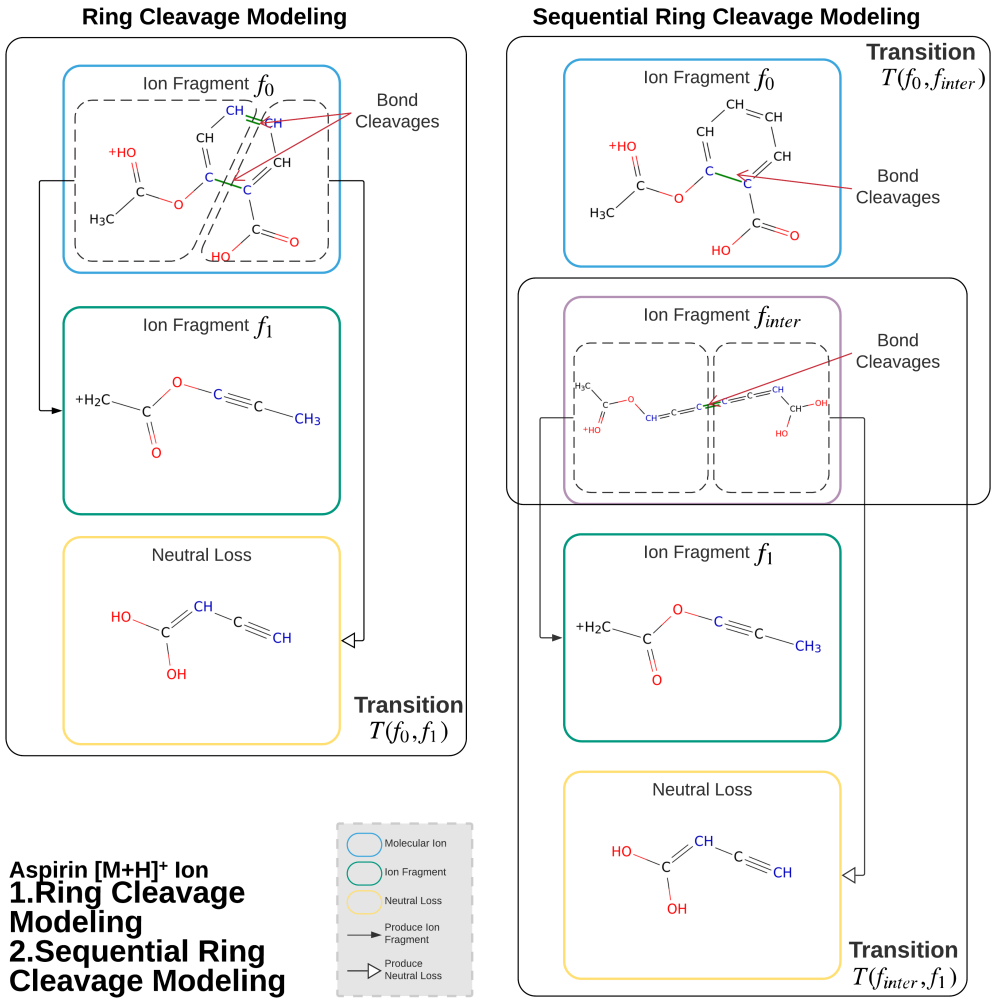


Figure 2.1: Diagram of ring cleavage models. The left side is the legacy (original CFM-ID) ring cleavage model, and the right side is the sequential ring cleavage model of the same ring cleavage.

2.2 Connectivity Matrix Features

In the updated version of CFM-ID, we chose to inherit most of the settings in the original CFM-ID [2], two of which are directly relevant to feature extraction and representation. First, feature extraction is always performed on a hydrogen-suppressed molecular graph (Section 1.6.3). Secondly, for each fragment transition, we also extract feature representations from three components: the bond that breaks, the structure of the ion fragment and the structure of the neutral loss fragment. The final outcome is then the concatenation of all three feature vectors. As with the original CFM-ID, the broken bond is still represented by the *atom pair features* (Section 1.6.3).

2.2.1 Basic Feature Representation

In the context of CFM-ID, the occurrence of a fragment transition depends on two factors: whether a chemical bond will break, and which fragment will be charged after the bond breaks. Our model assumes the answer to both questions lies in the structures around the disassociated chemical bond. The base version of the proposed feature representation methods encodes the entire structure of a given fragment. For a given fragment, our method transforms a given fragment from a chemical structure (Figure 2.2a) to a labelled undirected graph (Figure 2.2b). Specifically, we define each fragment structure as a graph $G = (V, E, VL, EL, R)$, where V is a set of vertices whose elements each corresponds to an atom and E is a set of chemical bonds between each pair of atoms in the fragment. VL and EL are the set of labels assigned to each vertex and edge, respectively. R denotes the root vertex of the graph, which is one of the two atoms connected to the bond before it breaks.

Our method then use tensors derived from an adjacency matrix (Figure 2.2c) of a fragment to create its feature representation. By definition, the adjacency matrix is a

square matrix whose (i, j) entry is 1 if there is an atom at $vertex_i$ that is connected to the atom at $vertex_j$. Such a representation method can be easily extended to handle more complicated labels via additional vectors or in the form of tensors. In our case, we represent a fragment graph G through a combination of two tensors. Let N_v represent the number of vertices and D_v represent the length of one-hot encoded feature vector per bond. Therefore a $N_v \times N_v \times D_v$ tensor T_{adj} holds the connectivity information of the given graph as well as the label of each chemical bond. To represent each atom, the second tensor, T_{vertex}, has the size $N_e \times D_e$ where N_e is the number of vertices and N_e is the size of the one-hot encoded labels for each atom.

Tensor T_{adj} is created from the adjacency matrix M_{adj}. Its first two axes resemble the M_{adj}, and the third axis is used to store features per edge. For every pair of vertices V_i and V_j, the feature vector $T_{adj}(i, j)$ is set to an all zeros vector if there is no edge. Otherwise, $T_{adj}(i, j)$ stores its associated chemical bond type as a one-hot encoded feature vector. Bond types can belong to one of the following categories, single, double, triple, quadruple and even higher bond orders, aromatic, and conjugated. In practice, it is very rare to encounter chemical compounds with bond order higher than triple, however, these higher bond orders still needs to exist because CFM-ID generates all theoretical possible chemical structures during its fragmentation graph generation. Tensor T_{vertex} is used to store feature vectors for atoms. For the i^{th} vertex and its associated atom in T_{vertex}, its feature vector is stored by the i^{th} vector of T_{vertex}. This vertex feature vector consists of two components. The first component describes the vertex's atom type as being one of the following: {Carbon, Nitrogen, Oxygen, Phosphorus, Sulfur, Others}. The second component indicates the current vertex's neighbourhood atom counts in one of following scales: {1, 2, 3, 4, 5 or More}. Before feeding this information into the CFM-ID training algorithm, the un-duplicated part of T_{adj} and all of T_{vertex} are fattened into 1-dimension vectors

and then concatenated into a single vector. Note that, since G is an undirected graph, only $Element_{i,j}$ of T_{vertex} are needed such that $i < j$. Figure 2.2d provides an illustration of this tensor to feature vector conversion process.

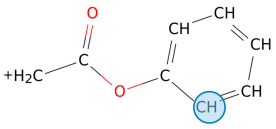
2.2.2 Feature Representation with A Subgraph Selection

This section discuss the two main drawbacks of the basic tensor representation method. The first drawback is that the larger the structure, the larger the adjacency matrix, and therefore the larger the size of the feature vector (Section 2.2.2.1). The second disadvantage is that our proposed method is not indexing invariant (Section 2.2.2.2).

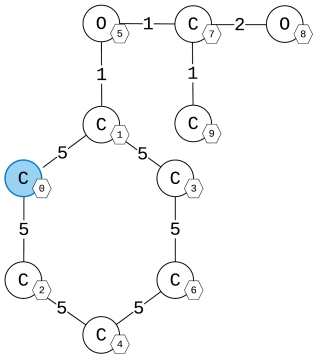
2.2.2.1 Challenges of Handling Feature Vector Dimensions

There are several challenges with regard to handling our proposed feature representation. First, the size of the adjacency matrix depends on the number of vertices in the graph. The larger the graph, the larger the matrix. Although the entire graph segment can be encoded using a sufficiently large matrix, this is considered unnecessary and problematic. In general, machine learning methods can benefit from more detailed feature representations, but this representation qincreases the risk of over-fitting. It also increases the consumption of computing resources. In the case of the original version of CFM-ID, when using a depth-2 fragmentation graph for a given sample, that molecule can have an average of more than 10,000 fragment transitions. At the same time, each feature vector consists of thousands to millions of bits, depending on the type of feature used and the training algorithm. The original version of CFM-ID itself was a very resource-intensive method already. Certainly, by implementing a larger and denser feature representation it will inevitably require even more computational resources.

To address this issue, we assume it is sufficient to only encode the local chemical



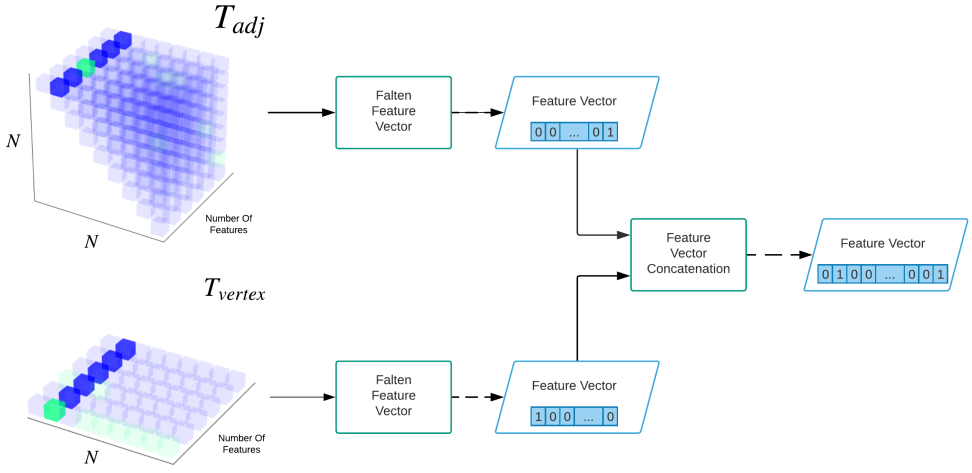
(a) An ion fragment, its root atom is highlighted by the blue circle.



(b) A graph extracted from the ion fragment in Figure 2.2a. Vertex's and bond's label indicate atom type and its bond order respectively. Number inside each hexagon shows vertex's index. Note that this graph does not include any hydrogen atoms.

$$
\begin{array}{c}
 \\
V_0 \\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \\ V_7 \\ V_8 \\ V_9
\end{array}
\begin{array}{c}
\begin{array}{cccccccccc} V_0 & V_1 & V_2 & V_3 & V_4 & V_5 & V_6 & V_7 & V_8 & V_9 \end{array} \\
\begin{bmatrix}
0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}
\end{array}
$$

(c) An adjacent matrix of graph in Figure 2.2b.



(d) Tensors are created from sub-figure(b) and (c), then they are turned into final feature vector.

Figure 2.2: Diagram of basic connectivity matrix feature (CMF).

structure of a chemical bond to determine how likely a fragment transition will occur. The intuition to do this is that for a given atom, its interaction with a distant neighbouring atom (far from a potential cleavage site) is less than that of a closer one, so it is safe to assume that distant atoms are less important. Recall that the question of whether a fragment transition will occur can be divided into two smaller questions: 1) Will the chemical bond break ? and 2) which fragment will be charged after the bond breaks ?. For the first question, research in the field of physical chemistry shows that the chance of bond rupture may largely depend on the chemical bond's local structure, which means that our hypothesis does make physicochemical sense. In recent work in this domain, Tanaka at el. [74] predicted whether a chemical bond will break by checking its bonding patterns, where a chemical bond is located, two adjacent atoms and chemical groups attached to it. For the second question, our limited observation from the annotated MS data show atoms such as sulphur and nitrogen have a higher tendency to get an unpaired valence electron and becomes a radical than other atom types such as carbon. This suggests the position of the charges after the bond breaks may be related to the type of atom in each fragment. To simplify the problem, we assume that the atoms near the bond dissociation are able to provide sufficient information to answer the question. This assumption may not be totally correct from a pure chemical or physical perspective, however it provides a rationale for the necessary trade-off from the computational point of view.

2.2.2.2 The Challenge of Graph Indexing

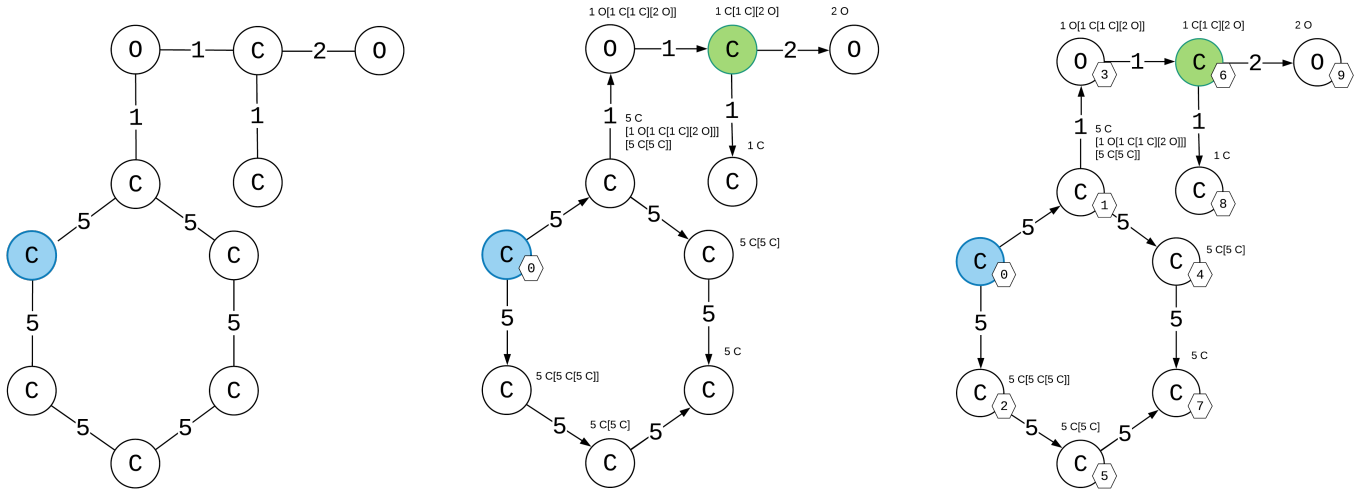
The second drawback of our proposed method is the fact that it is not indexing invariant. For an N-vertices graph, it's vertex indexing can be permuted without changing its underlying structure, therefore such a graph has $N!$ equivalent adjacency matrices. Each adjacency matrix corresponds to a different ordering of the graph's vertex indexing. Thus, although each of the adjacency matrix based feature

representations describe the same underlying structure, they may differ from each other. In a similar tensor representation approach, Simonovsky el al.'s method [66] tackled this problem by checking whether two adjacency matrices correspond to the same underlying structure via an approximate graph matching algorithm. Despite using an approximation algorithm, this approach still requires a rather intensive amount of computing resources. Fortunately, for a rooted graph, it is possible to assign an index to each vertex through a technique known as graph traversal such that the same structure always has the same indexing. Note that graph traversal is much faster than any graph matching algorithm. We proposed a graph indexing approach that is intended to achieve two objectives: (1) graphs with the same underlining structure are going to have the same indexing, therefore the same adjacent matrix; (2) Indexing of this graph needs to indicate the topological relationship between each vertex and root (that is, the vertex with the larger index number is farther from the graph root). Combined with the local structure encoding mentioned in the previous section (Section 2.2.2.1), our proposed method should select a partition from a chemical structure based on the index of each atom, and the selected partition should contain all the relevant topological information to predict the tendency of a fragment transition. Details of our proposed method can be found in the next section (Section 2.2.2.3).

2.2.2.3 Subgraph Selection

The selection method includes two graph breadth-first traversal algorithms, the first traversal is to assign a string heuristic value to each vertex in the graph, and the second traversal is responsible for selecting and indexing vertices using those heuristic values.

Illustrated in Figure 2.3a, for a given molecular graph, in the first pass, a heuristic string is created and assigned to each vertex trough a breadth-first traversal. First,



(a) A graph extracted from the ion fragment and it is used as input for subgraph selection.

(b) Phase one of subgraph selection, heuristic values are assigned to vertices. The arrows in the graph indicate vertices visiting orders during graph traversal.

(c) Phase two of subgraph selection, indices are assigned to vertices based on their heuristic values. The arrows in the graph also indicate vertices visiting orders during graph traversal.

Figure 2.3: Diagram of subgraph selection.

we define a (bond-order atom-type) pair as a 3-character string, the first character uses a number to indicate the order of the chemical bond, which leads to the current vertex and the remaining two characters indicate atom types of the current vertex using atomic symbols. For the one character atom symbols, an empty string is inserted between the atom type and bond order. For instance, a single bond leading to a carbon atom will have "1 C" as its bond-order-atom-type pair while string for a double bond leading to chlorine is "2Cl". For any non-root vertex in the graph, its heuristic string consists of the following two parts: (1) a bond-order-atom-type pair to describe the type of chemical bond leading to the current vertex and the atom type associated to the current vertex, and (2) a set of alphabetically heuristic strings from its children vertices, each placed within a bracket. For instance, the green vertex in Figure 2.3b, its associated atom is carbon and is connected to its parent vertex via an aromatic bond, therefore its own (bond-order atom-type) pair is "5 C" which forms the first part of the heuristic label. Its child vertex has a double-bond-oxygen pair and a single-bond-carbon pair. If we sort those two labels alphabetically the second part of the current heuristic label is "[1 C][2 O]". Once joined together, the heuristic label of the green vertex is "5 C[1 C][2 O]". In the situation where multiple edges lead to the same vertex, our method chooses the edge on the shorter path to the root vertex to create the heuristic label. If there are still ties, we use the edge with the lowest bond order to break the tie.

Figure: 2.3c provides an example of the second breadth-first traversal that assigns a new index to each vertex based on their labels. Starting from the root node, at each step the heuristic label created in the first traversal will be used to determine which child vertex to visit next, which is a vertex that has the alphabetical smaller label sort will be visited next. For example, after visiting the green vertex in the figure, our traversal process will visit $Vertex_8$ and then $Vertex_9$ because "1 C" is alphabetically smaller than "2 O".

Once the input graph is indexed, our method extracts the first N vertices in the graph based on their index in order to create an adjacency matrix, and then, feature vectors. Because the index number also reflects the distance between the current vertex to the root, the first N vertices are also the closest N vertices to the root. Since we are unable to determine the appropriate threshold, multiple versions of the same selection method are implemented with a threshold value of 6, 8, 10 or 16 atoms. Note that a threshold of size of 10 is actually sufficient to cover all atoms in our data-set with a radius of 2 bonds. Compared to the distance-based filtering that selects all neighbouring vertices within a given distance, this closest N-neighbourhood approach not only has a better feature space utilization but also is able to provide a better representation for a more linear chemical structure. For example, when using distance-based selection, long-chain structures with various lengths will have a lower chance of obtaining a unique feature vector. For example, a fixed-range filter with up to 3 atoms or 2 chemical bonds, we see that some fragments produced by propane and n-butylamine will share the same feature representation while they are structurally different. In comparison, the same problem is less severe in a neighbourhood limited setting.

Through this indexing and selecting process, the importance of atoms with respect to the current bond is captured by the topological distance between the atom pairs in the chemical space and labelling order inside feature space. More closely located atoms will be represented on the one end of the adjacency matrix as well the tensors T_{adj} and T_{vertex}, while father more distant ones will be represented on the other end.

2.3 Accelerated Parallel Training with Sampling

The original CFM-ID model was found to consume a huge amount of computational resources during its training phase. For instance, when training a model from 1500 chemical samples, the old CFM-ID model consumed around 9000 CPU core hours (Intel Xeon X5675 processors) on a cluster. Due to various factors such as memory requirements, programming complexity and hardware availability, CFM-ID methods employ a CPU-based parallel approach during their training phase which is built on the concept of data-driven parallelism. In the data preparation step, the training program randomly sorts and then evenly distributes training samples across all sessions. Each session is an independent process that has its own dedicated CPU core. During training, a training process only performs computations on the samples assigned. Data such as gradient values and model parameters are exchanged between processes through the Message Passing Interface (MPI). In this way, each training process can make full use of its dedicated CPU core. In addition, this framework allows training programs to be executed on distributed systems to overcome the limitations of the number of CPU cores on a single machine.

Recall that the CFM-ID approach learns its model parameters through an Expectation-Maximization (EM) algorithm, which consists of two alternating steps. The E-step estimates the expected value of the unknown variables based on parameters learned from the current model and the M-step updates parameters using variables estimated by the E-Step. Figure 2.4 provides an overview of the CFM-ID training process, for each alternating iteration in EM, the E-step calculation can be finished within minutes, while its M-step calculations will take hours if not days to complete. As illustrated in the same figure, an M-step consists of several periods, each of which consists of mini-batch calculations. The mini-batch calculation consists of 4 operating phases, gradient calculation, gradient aggregation, parameter update and

parameter distribution. To ensure model parameters are properly shared between training sessions, the gradients computed during each individual training session are sent to the main training session, then the model parameters are updated and distributed back. This means each of 4 operations has to start at the same time across all processes, and the operation is considered as completed only if each training session finishes its part. For instance, the parameter updates operation can only begin after all training sessions have sent their gradients value to the main process.

[!p]

In a situation where each input sample requires a similar amount of computing resources, this data-driven framework has excellent scalability. In such a model training speed can be increased in a nearly linear manner with a growing number of processes. Unfortunately, in the case with the CFM-ID, this situation cannot be met. As shown in Figure 2.5, the samples used in this work cover a wide array of very different molecules. There is a significant difference in fragment transition counts between samples when using a depth-2 fragmentation graph. Indeed, this difference will be more pronounced as the depth of the fragmentation graph increases. During the model training process, this enormous difference means that some processes will be required to perform a much larger amount of computation than others in the gradient computation. This difference creates a serious bottleneck during model training as faster processes must wait for the slowest process to complete before entering the next phase. In some cases, the wasted computation time is more than the actual time spent in the calculation (Figure 2.4).

With our proposed modifications, especially the new representation method, the new CFM-ID approach consumes significantly more computational resources in both its training and prediction time. Preliminary results show that the model inference time is still comparable to the original CFM-ID method, but the time to train the model, with the same sample data set, is not feasible, even when using the latest

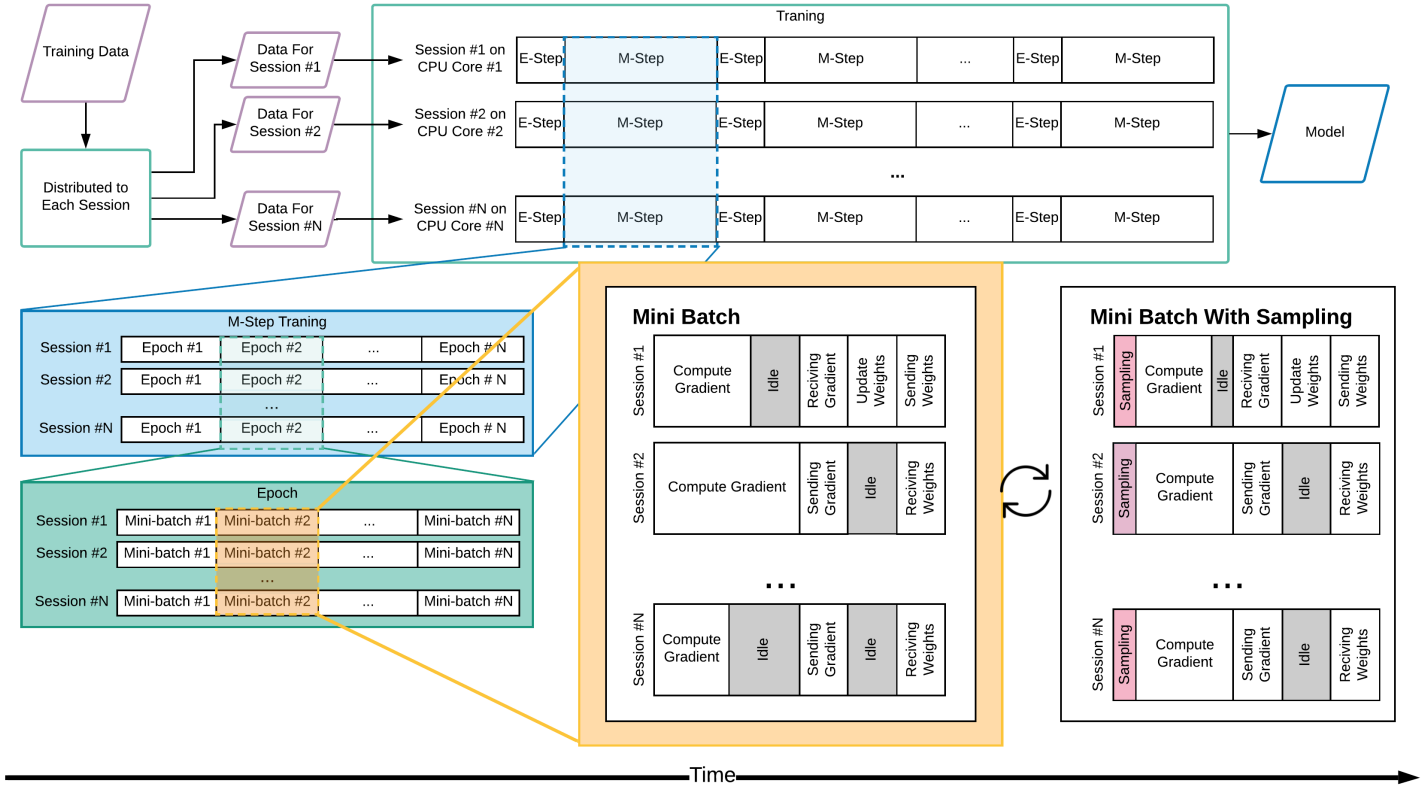


Figure 2.4: Diagram of CFM-ID training workflow. The mini-batch block within the orange background is used by the legacy version of CFM-ID, it has been replaced by the mini-batch block on the right with sampling methods.

high-performance computing hardware such as the Compute Canada Cedar Facility. The estimated computational resources needed to train the newer version of CFM-ID model are approximately ten to a hundred times more than the original one.

To make the training of our updated CFM-ID model computationally feasible, we proposed a sampling framework to reduce process idle time and increase the training speed. At the beginning of each mini-batch computation, a sampling step is performed on each sample before the gradient calculation begins. This operation selects a subset from all possible fragment transitions for a given molecule and uses the subset to compute the gradient and update the model (Figure 2.4). The same sampling step is repeated for every mini-batch, with different outcomes for the same input at each iteration. By applying a maximum threshold on the number of transitions that can be selected each time, this ensures each training session has a similar amount of computations during the gradient computation phase, which in turn reduces the process idle time mentioned above. Note that sampled subsets can be only used during the gradient computation, not in either model evaluation or loss computation. In total, three sampling methods are proposed. These include random sampling (Section 2.3.1), random walk sampling (Section 2.3.1), and peak difference sampling (Section 2.3.2).

2.3.1 Random Sampling and Random Walk Sampling

The most straightforward method to limit the number of transitions during each gradient calculation is to use a random sampling method. As the name suggests, for a given training sample, this approach first randomly (uniform distribution) selects a subset of non-self-point fragment transitions from the original sample and associated fragments, then the corresponding persistence transitions are added for each selected fragment. At first glance, this seems to be a good approach because of its speed of execution. Recall that, for a given training input, each of its fragment transitions

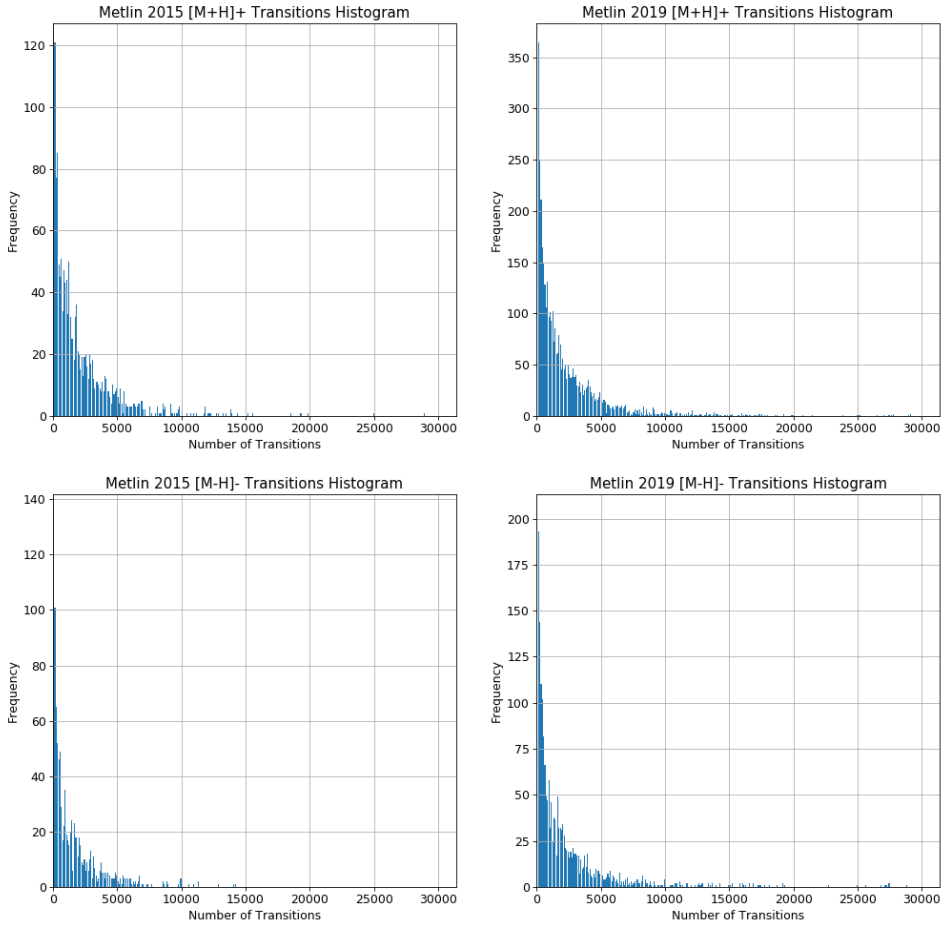


Figure 2.5: Histogram for the Meltin 2015 $[M + H]^+$set and the Meltin 2019 $[M + H]^+$set sample transitions

are organized by the input chemical compound's fragmentation graph. Therefore sampling a set of fragment transitions is equal to select a set of edges and associated vertices from the fragmentation graph. Besides, fragment transitions on the lower level of a fragmentation graph that are closer to the graph root have a higher chance to make more of an impact on the overall performance of the model because they are shared by downstream fragments on the graph. Since the random selection approach completely ignores the fragmentation graph, as demonstrated in the top half of Figure 2.6, there is always a risk of missing these important fragment transitions using this random sampling method during the learning process.

Evolved from a random sampling method, and inspired by similar random walk approaches applied in the other domains of machine learning [25], [55]–[57], we proposed a random walk sampling approach that selects a set of transitions with respect to the structure of the fragment graph. A random walk iteration begins at the root of the graph, and each sub-sequent step random (uniform distribution) selects a downstream vertex as a destination until it reaches the leaf vertex of the graph. Persistence transitions are excluded during this random walking process to avoid any chance of an infinite loop. In addition, relevant persistence transitions for selected fragments are added into results after the random walks. By doing so, the sampled transitions and fragments form a valid fragmentation graph of input samples, but they are smaller them the original. The total number of sampled transitions can be loosely controlled by limiting the number of walks performed during the sampling process. As shown in Figure 2.6, because a random walk sampling iteration always starts from the root vertex and includes all downstream vertices in its pass, the output of this selection is always a single graph. Moreover, the more important lower edges now have a higher chance of being selected compare to the random sampling approach.

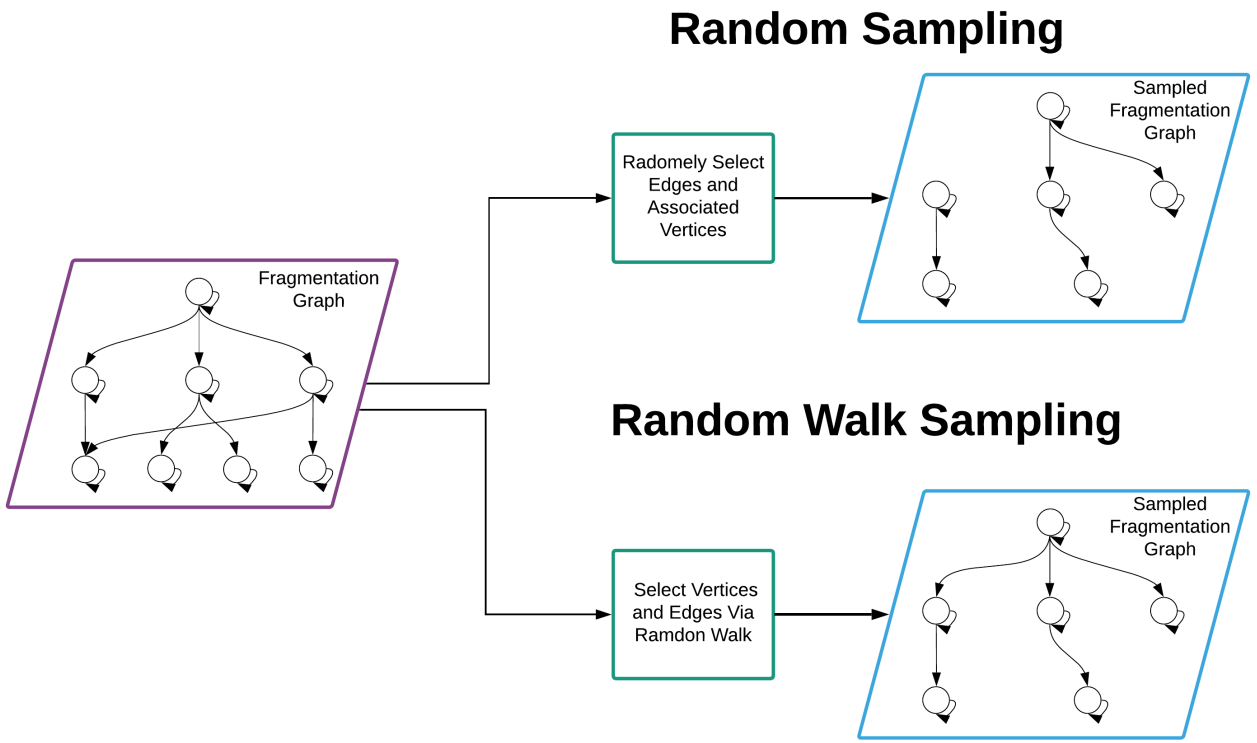


Figure 2.6: Diagram of random sampling method and random walk sampling method.

2.3.2 Peak Difference Sampling

We also proposed a more complicated sampling approach, namely peak difference sampling, which was inspired by the Markov Chain Monte Carlo [3] algorithm. It was also inspired by the other bagging and boosting methods used in machine learning. The intuition behind this method is to force the learning method to update CFM-ID model only through the most significant fragment transitions at the given time. First, we match peaks between a predicted spectrum and measured spectrum, and then the prediction error is measured by the absolute difference in intensity between the predicted and measured peak that is $P_{error} = |Intensity_P^{pred} - Intensity_P^{measured}|$. In the situation where the predicted peak has no counterpart in the measured spectrum, the intensity of measured peaks is considered as zero, and vice versa. This way the most incorrectly predicted peaks can be identified through the peak error value and the most significant fragment transitions at the given time are defined as the fragment transitions that form the path from the root of fragmentation graph to the peak producing fragment.

This process first uses the current learned model to generate the predicted spectra for the chemical samples at the beginning of a mini-batch, then, as illustrated in Figure 2.7, for an given chemical, it computes the errors between the predicted and the measured spectrum, and then determines which peaks are important by sorting them on the basis of the prediction errors. The top-N most severely miss-predicted peaks are then picked for further computation. To find the corresponding fragment transitions for those miss predicted peaks, the algorithm traveres the fragmentation graph and records a list of vertices by matching the observed peak's mass and the predicted fragments' mass. Next, all the transitions between the root vertex and selected fragments vertices are selected by the sampling methods. Finally, self-pointing transitions are added in the same manner as the random walk sampling counterpart.

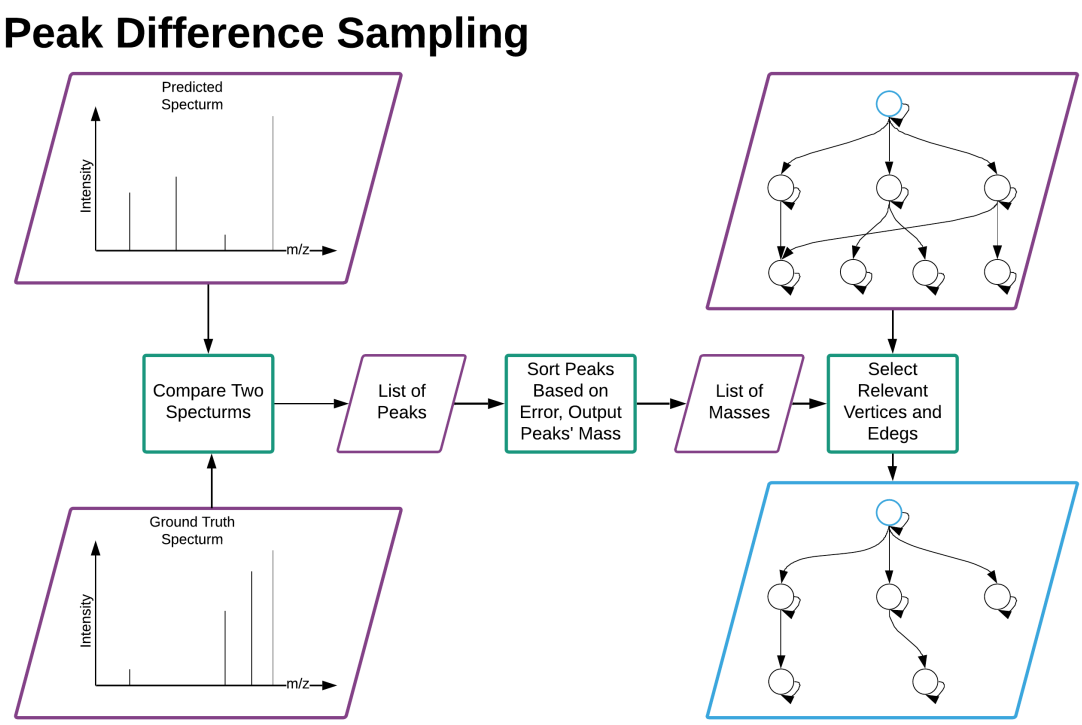


Figure 2.7: Diagram of peak difference sampling method.

Chapter 3

Empirical Evaluations

In this section, we present the results on several MS/MS spectrum prediction tasks using the feature representation and sampling methods described in Section: **??** and Section: 2.3.

3.1 Data Collection and Preparation

This study used experimental ESI-MS/MS data from the Scripps Research Institute's Metlin MS database [26], [67]. These MS/MS spectra were all collected by a Quadrupole Time-of-Flight (QToF) instrument in positive $[M + H]^+$ ionization and negative $[M - H]^-$ ionization modes. The spectra were then grouped by their molecular type, and ionization mode. Each training/testing sample consists of a molecule whose mass is less than or equal to 1000 Dalton (Da), and its ESI-MS/MS spectra measured at three different collision energy levels: 10 eV, 20 eV and 40 eV. Note that, only the most common isotope peaks are presented in these spectra. In total, 7 data sets were used in this study; the two legacy data sets (dataset No.1 and No.2) were used in the original CFM-ID study, and 5 recent data sets (No.3-7) were newly collected for this work. These 7 data sets are the following:

1. Metlin Metabolites 2015 ($[M+H]^+$) - 4473 spectra for 1491 non-peptide metabolites with spectra measured in the$[M + H]^+$ mode ionization from the Metlin database.

2. Metlin Metabolites 2015 ($[M-H]^-$) - 2928 spectra for 976 non-peptide metabolites with spectra measured in the $[M - H]^-$ ionization mode from the Metlin database.

3. Metlin Metabolites 2019 ($[M + H]^+$) - 12165 spectra for 4055 non-peptide metabolites from the Metlin database, all of which are collected in the$[M+H]^+$ ionization mode.

4. Metlin Metabolites 2019 Common($[M+H]^+$) - 4449 spectra for 1483 molecules and their spectra. All molecules are overlapped with Metlin Metablites 2015 ($[M + H]^+$) and Metlin Metabolites 2019 ($[M + H]^+$) set, while their spectra

are newly collected. A detailed description of the sample distribution can be found at Section 3.4.3.

5. Metlin Metabolites 2019 ($[M-H]^-$) - 6120 spectra for 2040 non-peptide metabolites from the Metlin database all collected in the$[M-H]^-$ ionization model.

6. Metlin Metabolites 2019 Common ($[M-H]^-$) - This set contains 913 molecules and their spectra. All molecules are overlapped with Metlin Metabolites 2015 ($[M-H]^-$) and Metlin Metabolites 2019 ($[M-H]^-$) set, while their spectra are newly collected. A detailed description of the sample distribution can be found at Section 3.4.3.

7. Metlin Metabolites Training Speed Test Set - 240 $[M+H]^+$ spectra for 240 randomly selected metabolites all measured at 10 eV. This data set was used to determine speed gains when applying different sampling methods during training time. Note that, this is only data-set does not have 3 spectra per sample.

A raw Metlin ESI-MS/MS Q-ToF spectrum typically carries a large number of noise peaks caused by instrument noise. This noise manifests in two forms: one type of noise peak has a very low relative intensity (less than 1%) and the other type of noise peak is a lower intensity peak close to a major peak (but often less than 1 Da away from that major peak). We used the following criteria to clean up each MS spectrum and to leave legacy MS data sets untouched. For the newly acquired data, data prepossessing removes the first type noise by filtering out every peak whose relative intensity is lower than 1%. The second type noise peaks are then iteratively removed such that the peaks within the 0.9 Da mass difference from a major peak are filtered out. This threshold value is chosen since it is relatively safe to assume peaks cannot be separated by less than 1 Da in an MS spectrum, as

this mass difference corresponds to the smallest mass-to-charge ratio difference (one hydrogen atom) that is possible. However, atomic masses of different atoms are not exactly integer values, for instance, hydrogen has an isotopes mass of 1.007825 Da. Therefore, it is possible to have two major peaks positioned less than 1 Da away from each other. This process begins with the peak that has the highest intensity value in the spectrum and moves toward the lower intensity ones until all peaks in the spectrum are visited.

3.2 Model Evaluations

All experiments were performed using a 10-fold cross-validation [73] framework except for the training speed assessment (Section 3.3).

A post-processing step identical to the original CFM-ID study [1] was applied to the predicted ESI-MS/MS spectra. For a predicted ESI-MS/MS spectrum, the post-processing first sorts all peaks by their intensity, then selects the most significant peaks by intensity iteratively. The selection process is stopped once the output spectrum meets one of the following criteria: (1) either the number of peaks is at least 5 and the sum of peak intensity is equal or more than 80% of the total input intensity or (2) the number of peaks equals to 30.

To enable direct comparisons between our version of CFM-ID and the original one, we purposely set the hyperparameters of our approach to be the same as the original CFM-ID unless otherwise mentioned. The model performance is then measured by several metrics between the experimentally measured ESI-MS/MS spectra and their corresponding predicted counterparts. Details of model configuration and performance metric can be found in Section 3.2.1 and Section 3.2.4.

3.2.1 Model Configuration

In our evaluations, we trained all models with the same basic model configurations and the same experimental specific tweaks. The legacy CFM-ID's performance measurements are directly from the original publications [1].

3.2.2 Model Training Configuration

For each model, we set both the maximum depth of the fragmentation graphs and the model depth to 2 for all models. The maximum number of allowed ring breaks are also limited to 2. Due to computational constraints, we only tested the neural network

variant of CFM-ID with a simple fully connected structure consists of two hidden layers each with 128 nodes. Each hidden layer node has a ReLU [52] activation function, and the output node has a linear activation function. Furthermore, the hidden layer also carries a standard dropout mechanism [68]. The model training process learns the model through a set of Adam optimizers [40], one per each M-step. All Adam optimizers have the same setting: $\beta_1 = 0.9$ and $\beta_2 = 0.999$. A naive learning rate decay method controls the optimizers' starting learning rate, beginning from 0.001 and ending at 0.00025; this value is reduced by half when training losses can't be further improved at the end of each M-step. Each sampling methods uses the following specific parameters:

- Random Sampling: The total number of random selections is set to 100.
- Random Walk Sampling: The total number of random walk count is set 100.
- Peak Difference Sampling: A maximum of 30 most different peaks will be used during each sampling step.

3.2.3 Feature Configuration

All experiments used the following chemical features unless otherwise stated where *Ion Root* and *NL Root* denotes the root atoms for ion fragment and neural loss fragment respectively.

- Break Atom Pair: Indicators for the type of atom type pairs formed by the *Ion Root* and the *NL Root*. Unlike the same feature used in original study, indicators for the ring cleavage are removed [1].
- BrokenOrigBondType: Indicators for type of cleaved bond in one of single, double, triple, aromatics, conjugated, ionic, or hydrogen bond only [1].

- Gasteiger Charges: Features for Gasteiger charges the for the *Ion Root* and the *NL Root* in the original unbroken molecule [1].

- Hydrogen Movement: Features for how hydrogen atoms switch sites [1].

- *Ion Root* and *NL Root* Connectivity Matrix Features: Connectivity Matrix Features proposed in section 2.2. The size of the connectivity matrix has a default value of 10.

3.2.4 Evaluation Metrics

Models are compared by their ability to predict ESI-MS/MS spectra based on input chemical structures. The performance of the prediction tasks was evaluated via metrics computed between a ground truth (i.e. the experimentally measured MS/MS spectrum) S_M and the predicted MS/MS spectrum S_P. In order to evaluate the performance of our methods in a more realistic manner, the predicted spectra were generated using a pruned fragmentation graph at a threshold of 0.001 [1]. Between S_M and S_P, a pair of peaks are considered as matching peaks if their mass difference is smaller than 0.01 Da and 10 parts per million (ppm). The first three metrics are unweighted measurements that do not consider peak intensity. We define set $M(S_P)$ and $M(S_M)$ for all peak masses in S_M and S_P respectively, then these metrics are described as follows:

- Precision – The percentage of matching peaks over the predicted spectrum: $|M(S_P) \cap M(S_M)| \div |M(S_P)|$ [1].

- Recall: The percentage of matching peaks over the predicted spectrum: $|M(S_P) \cap M(S_M)| \div |M(S_M)|$ [1].

- Sørensen–Dice coefficient: $2 \times |M(S_P) \cap M(S_M)| \div |M(S_P) + M(S_M)|$ [1]

[1]This measurement was incorrectly called the "Jaccard Index" in the original CFM-ID papers [1],

Although accuracy, recall, and the Dice coefficient are commonly used indicators in this case, they do not reflect whether the model can predict major peaks in the spectrum. Especially in the case of a predicted MS/MS spectrum that has multiple matching lower intensity peaks and mismatching high-intensity peaks, those three metrics can produce a misleading reading and create a false conclusion. Therefore, we also include two indicators – weighted precision [1] and weighted recall [1] – to show how well models work when taking intensity into account:

- Weighted Precision: The total percentage of matching peaks' predicted intensity over total intensity in the predicted spectrum [1] where m and h denote the mass and relative intensity of a given peak:

$$\text{Weighted Precision} = \frac{100 \times \sum_{(m,h) \in S_P} h \times g(m)}{\sum_{(m,h) \in S_P} h}$$

$$\text{where } g(m, M(S_P), M(S_M)) = \begin{cases} 1 & \text{if } m \in (M(S_P) \cap M(S_M)) \\ 0 & \text{otherwise} \end{cases}$$

- Weighted Recall: The total percentage of matching peaks' predicted intensity over total intensity in the measured spectrum [1] where m and h denote the mass and relative intensity of a given peak:

$$\text{Weighted Precision} = \frac{100 \times \sum_{(m,h) \in S_P} h \times g(m)}{\sum_{(m,h) \in S_M} h}$$

$$\text{where } g(m, M(S_P), M(S_M)) = \begin{cases} 1 & \text{if } m \in (M(S_P) \cap M(S_M)) \\ 0 & \text{otherwise} \end{cases}$$

[2].

Among all metrics including both weighted and unweighted variants, the Dice coefficient is the single most important measurement in this study since it is also used as ranking metric by CFM-ID in spectrum classification tasks.

3.3 Training Speed Evaluations

This experiment was used to determine the training speed gain when employing different sampling methods. Due to the computational feasibility concerns, this experiment was carried out on a relatively small data set with only 240 randomly selected compounds and their 10 eV MS/MS spectra. Since there is no significant difference between time cost per energy level with the same hyperparameters, testing on a single energy level is sufficient for our purposes. To ensure fairness, this experiment executes each training algorithm 5 times on the same hardware with the same configurations except the sampling methods used.

The performance of the learned models was measured by average training log-likelihood losses and average training Dice coefficient. Although the validation metrics are a better choice than training metrics, due to the size and diversity of data set, they are hardly meaningful. Figure 3.1 presents the result of this test as series of these two measurements at different time points. Ranking by the training speed, model training with random sampling method finished under 2,000s, followed by the random walk sampling method and peak difference sampling method each of which used approximately 3,700s and 10,000s – on average. In contrast, the original training approach took around 1,400,000s to reach a comparable training loss level. Judging by the two model performance metrics, either random sampling, or random walk sampling appear to match the original training approach, however, the peak difference sampling approach does manage to achieve a similar training loss while using only 1/14 time compared to the original CFM-ID training method.

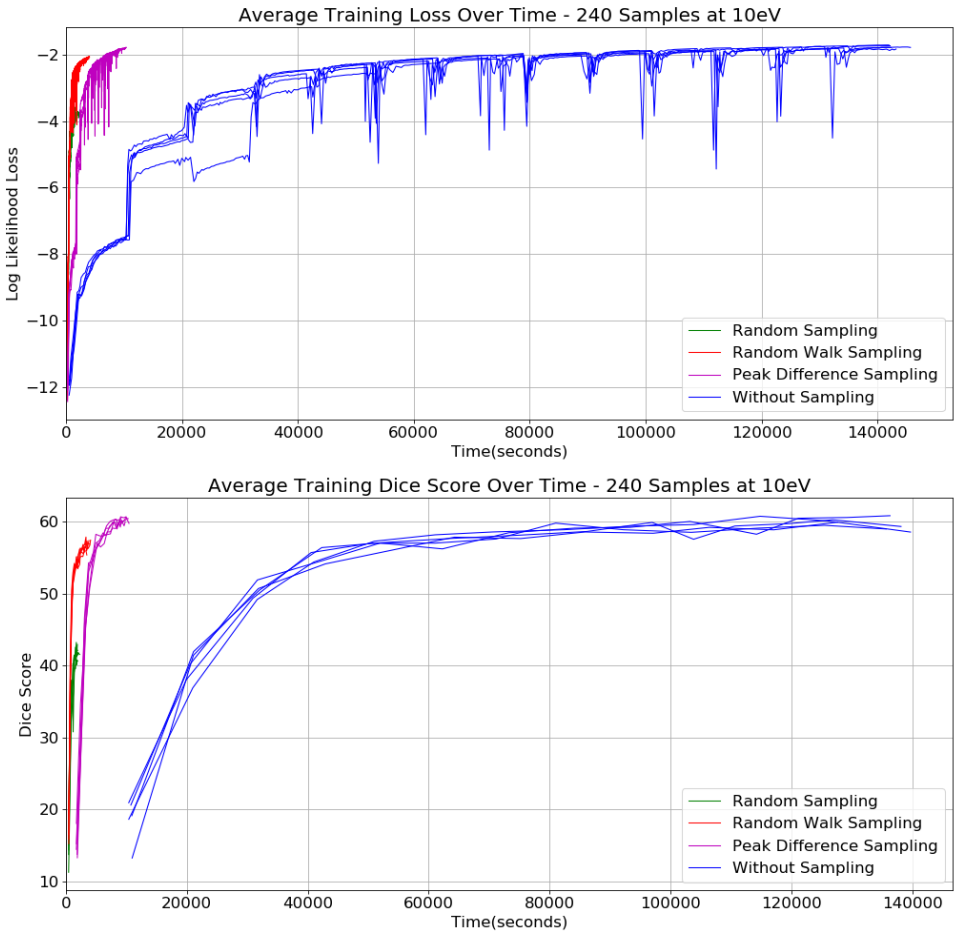


Figure 3.1: Training speed comparison between four different training methods. Since CFM-ID used an EM algorithm on log-likelihood, it performs a gradient ascent instead of more common gradient descent. The spike of the log-likelihood drop is caused by alternating between the E-Step and M-Step.

3.4 Sampling Method and Spectrum Prediction Evaluations

In this section, we present the results of more in-depth comparisons between the CFM-ID models learned by our new approach as well as the legacy CFM-ID approach. Two models from the original CFM-ID work are used as performance baselines, denoted as SE-CFM and CE-CFM. A detailed description of them can be found in the original CFM-ID publication [2].

3.4.1 Results on Model Prediction Evaluations

3.4.1.1 Sampling Methods Evaluations

The first experiment was carried out on the Metlin Metabolites 2015 ($[M+H]^+$) data set and mainly focused on determining the model performance trained via different sampling methods. Since those experiments were all carried out on the same data set as the legacy CFM-ID model, we also used those results to compare our models with their legacy counterparts.

The left side subgraph of Figure 3.2 shows the averaged performance across all three collision energy levels and the right subgraphs report the model performance at different collision energy levels. In contrast to the two baseline models from the original CFM-ID study, two out of three models learned with our proposed sampling methods achieved better performance in terms of the overall average Dice coefficient. The fastest training method, random sampling, performed worse than the baseline models while random walk sampling and peak difference sampling show significant improvement compared with the legacy models. Since the random sampling method does not perform well, it will not be discussed further. When we considered the results of each energy level separately, the models trained with the random walk sampling and peak difference sampling algorithms have similar improvements at dif-

Models	Energy	Z-Score	P-Value
RWS+CM10 (Metlin 2015)	10 eV	8.65	<1E-10
	20 eV	8.315	<1E-10
	40 eV	6.045	1.50E-09
	Overall	9.938	<1E-10
PDS+CM10 (Metlin 2015)	10 eV	10.2701	<1E-10
	20 eV	11.441	<1E-10
	40 eV	10.635	<1E-10
	Overall	13.669	<1E-10

Table 3.1: Results of two-sample z-test for comparing prediction's Dice coefficient between our models and baseline models.

ferent energy levels over the legacy or baseline models. At the 10 eV, 20 eV and 40 eV energy levels, the best performing peak difference sampling algorithm achieved an increase in Dice coefficients of 22.9 %, 26.2 % and 22.12 % respectively, followed by the random sampling method with 18.7 %, 19.8 % and 12.2 %. We had also performed two-sample z tests between the Dice score of predictions made by models and baseline. As shown in Table: 3.1, p-values suggest that spectra predicted by models trained with the random walk sampling and peak difference sampling algorithms are significantly different than spectra that are predicted by baseline models.

In the context of *in-silico* spectrum predictions, high recall performance or high-precision performance can be easily achieved separately. For a molecule, the former can be done by simply enumerating all possible peaks, while the latter can be achieved by only predicting the peak corresponding to molecular ions. The legacy models have a noticeable difference between their precision and recall performance. In contrast, models trained by random walk sampling and peak difference sampling achieved 43.19% and 30.8% improvement in terms of precision respectively, and the weighted precision increased by 12.9% and 10.9% respectively. At the same time, the recall performance is slightly worse than the legacy approaches, with a 4.6% and 1.6% decrease respectively. These performance metrics indicate that spectra predicted the

newer models not only have a larger intersection with their experimentally measured counter-part MS/MS spectra but also have a lower number of false-positive peaks.

More importantly, our models have achieved better spectrum prediction accuracy while using fewer parameters. The legacy CFM-ID models in this test had 7,171,176 parameters [1] while our models only have 376,707. Although our models are harder to learn compared to the legacy counterpart, they are faster in the inference phase due to a nearly 95% reduction in parameter counts.

3.4.2 Results on Connectivity Matrix Feature Sizes Evaluations

The second batch of experiments aimed to determine the best connectivity matrix size used for feature representation. Recall that the size of the connectivity matrix determines how many atoms of each fragment are included in the feature representation. On one hand, a larger connectivity matrix does contain more in-depth information, which may be required to produce a more accurate prediction, on the other hand, a large matrix brings problems such as possible over-fitting and a much longer training time. All models in this batch are trained using the same features and model configurations, expect the size of connectivity matrix.

Figure 3.3 shows the 10-fold cross-validation results using different connectivity matrix features at size 6, 8, 10, 16 on the Metlin Metabolites 2015 ($[M + H]^+$) data set. Unsurprisingly, the feature that has been extracted using the largest connectivity matrix manages to surpass all others in terms of the Dice coefficient and precision performance. We also see a clear recall-precision trade off-trend, the larger the matrix, the lower the recall score and the higher the precision. Moreover, as the size of the matrix increased, the Dice coefficient gains slowed down. Since a feature that uses the size 10 matrix have the almost the same Dice coefficient performance as the much larger size 16 matrix and considering that the amount of computation

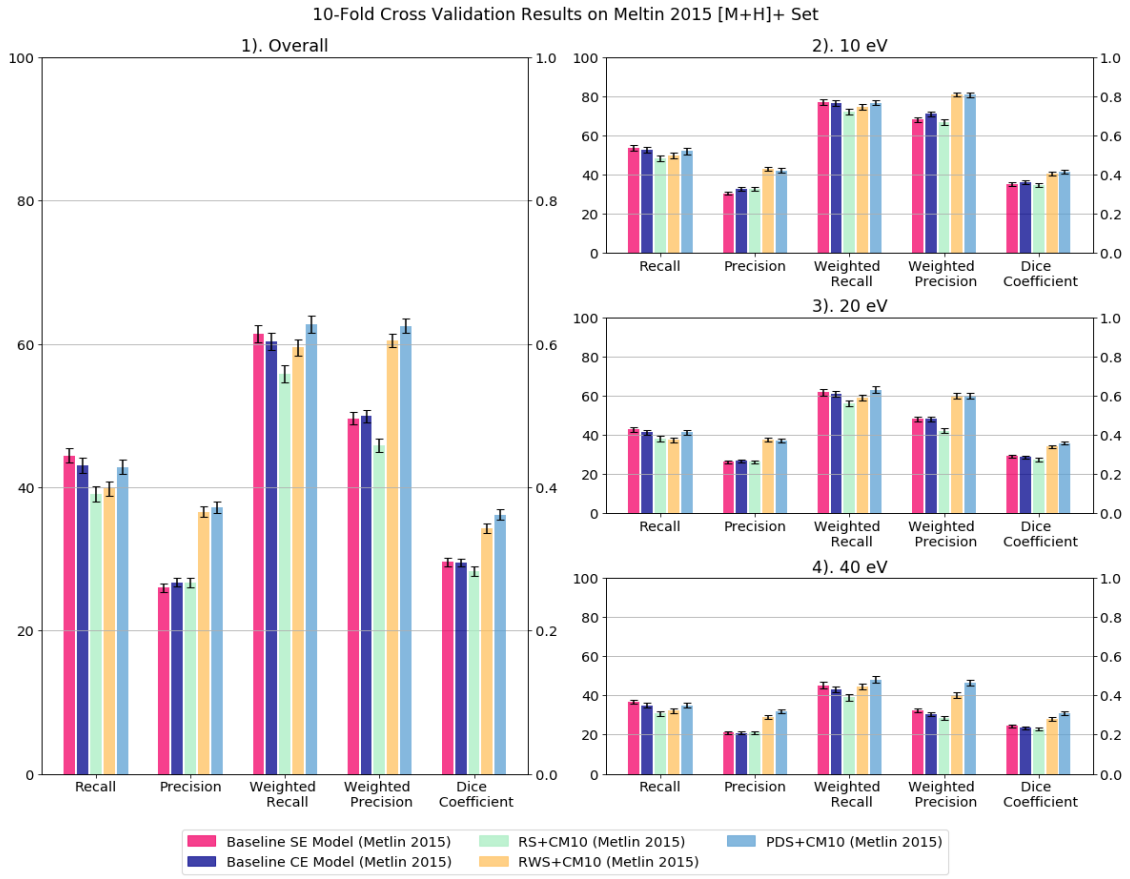


Figure 3.2: Spectrum prediction results for the Metlin metabolites 2015 $[M + H]^+$. Each bar displays mean scores for its metrics with an error bar indicates the 95% confidence interval. The plot on the left presents the overall performance of the model, and plots on the right provide the performance measures for each collision energy. Values used in this figure can be found at Table A.1.

required by each of them, we chose the size 10 matrix to be used in the rest of our evaluations.

3.4.3 Spectrum Prediction Evaluations on A Larger Dataset

The third batch of experiments evaluated our model performance with larger MS data sets. We used several positive spectra data sets, namely, Metlin Metabolites 2019 ($[M + H]^+$) : Metlin 2019^+, Metlin Metabolites 2019 Common ($[M + H]^+$): Metlin $Common^+$ as well as several negative spectral sets including, Metlin Metabolites 2019 ($[M - H]^-$): Metlin 2019^- and Metlin Metabolites 2019 ($[M - H]^-$):Metlin $Common^-$. Both positive set and negative set are much larger than their predecessor used in the original CFM-ID study, with the Metlin 2019^+ set having 4055 molecules and Metlin 2019^- set having 2040 molecules. With increased sample sizes, these data sets cover more diverse chemical structures compared to the legacy data set. We expect that on one hand, more data may improve model performance, while on the other hand, more diverse samples pose a challenge to the model's predictive ability.

All models in this batch were trained using peak difference sampling methods with standard features and model configuration mentioned in Section 3.2.1. Because the data used in this section are collected and pre-processed differently than the older data set which was used by the older CFM-ID, we used the common data sets (Metlin $Common^+$ and Metlin $Common^-$) to create our baseline models. Illustrated in Figure 3.4, that samples in common sets also have their chemical structure included in both Metlin Metabolites 2015 ($[M + H]^+$) and 2019 ($[M + H]^+$) set for positive ion mode and Metlin Metabolites 2015 ($[M - H]^-$) and 2019 ($[M - H]^-$) for negative ones. This gives us data collection close to the data set used in the original work yet has up to date measured spectra. Furthermore, under the 10-fold cross-validation framework, samples are carefully assigned to each fold such that molecules in $fold_i$

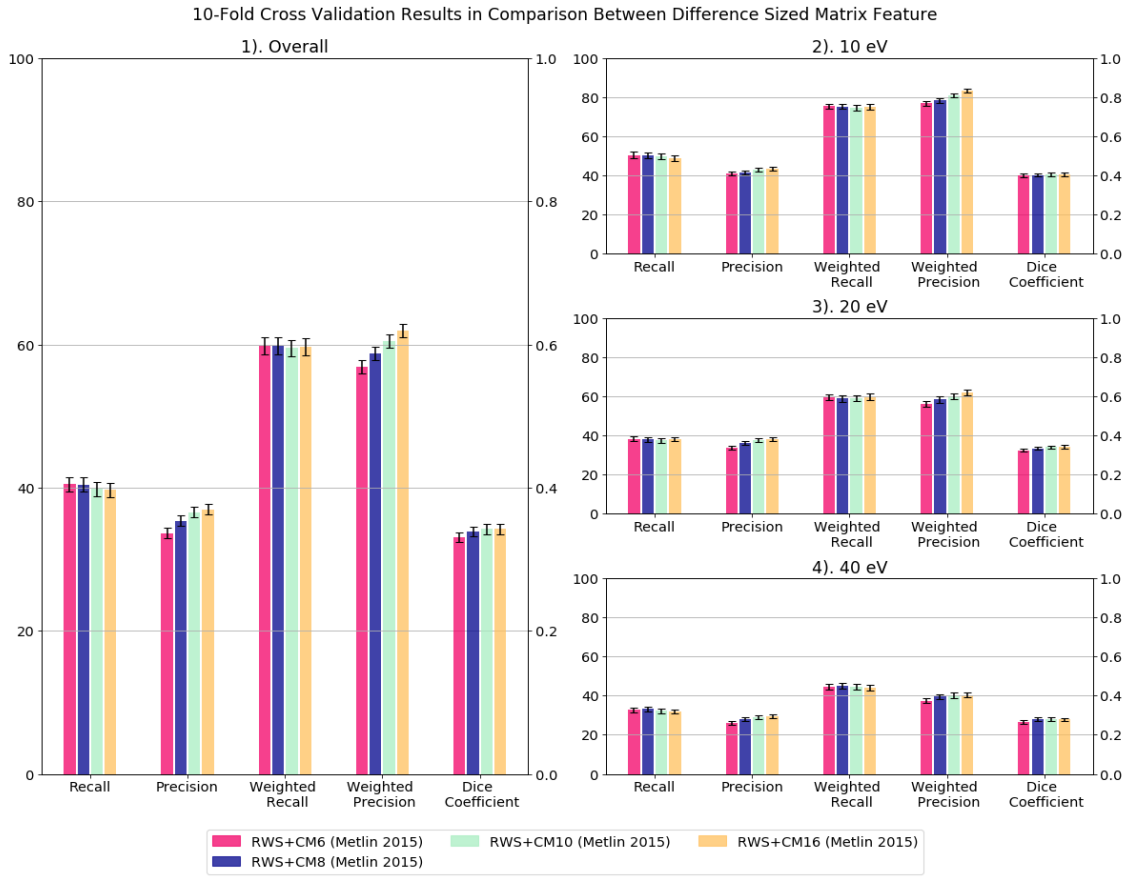


Figure 3.3: Spectrum prediction results for models using different sized connectivity matrix features. Each bar displays the mean scores for its metrics with an error bar indicates the 95% confidence interval. The plot on the left presents the overall performance of the model, and the plots on the right provide the performance measures for each collision energy. Values used in this figure can be found at table A.3.

of Common sets also in the fold$_i$ of the full sets.

Figure 3.5 and 3.6 show that the performance of the upgraded CFM-ID models trained from the full set and common set are very close for both positive and negative data-set. When we evaluated model performances via the Metlin Common$^+$ and Metlin Common$^-$, the Metlin 2019$^+$ and Metlin 2019$^-$ models managed to outperform the common set models by an average of 3.73 % and 3.77% in terms of the overall Dice coefficient. Comparing the results between different collision energies, a larger amount of performance gain can be seen at a higher energy level with 2.77%, 4.29%, and 4.47% improvements for 10 eV, 20 eV and 40 eV collision energy levels for positive ion mode spectra models and 2.0%, 4.13%, and 6.17% for negative ion mode spectra. These results suggest that with our current approach, the higher collision energy mass spectra prediction gains more benefits from a larger data set. The exact cause of these observations is not yet clear and thus requires further investigation. We believe that further improvement to the lower collision energy level spectra prediction may require a deeper fragmentation graph and a more effective way to create and explore the graph.

3.4.4 Spectra Classification Evaluations on CASMI 2016

In this section, we present the spectra classification results using CASMI 2016 [63] competition (category 3) data set. This data set consists of a total of 208 experimentally collected MS/MS spectra, 127 measured in the positive $[M+H]^+$ ion mode, and 82 in the negative $[M-H]^-$ ion mode. Since spectra in this are collected using Higher-energy collisional dissociation (HCD) not Collision-induced dissociation (CID) used by our training data, a pre-processing step was taken to determine the corresponding collision energy level for each spectrum with an equation provided by *Thermo Fisher Scientific* and mapped into one on of the corresponding CID energies of 10, 20, 40 eV. In addition, candidate structures are retrieved from CFM-ID 3.0's

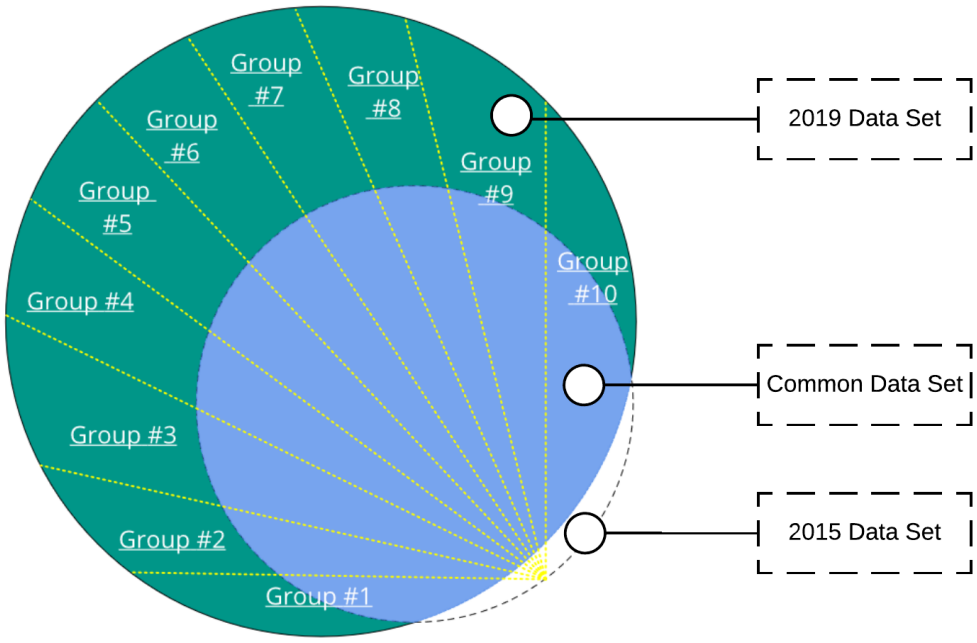


Figure 3.4: Diagram of sample distribution between the Metlin 2015 data-sets and the Metlin 2019 data-sets.

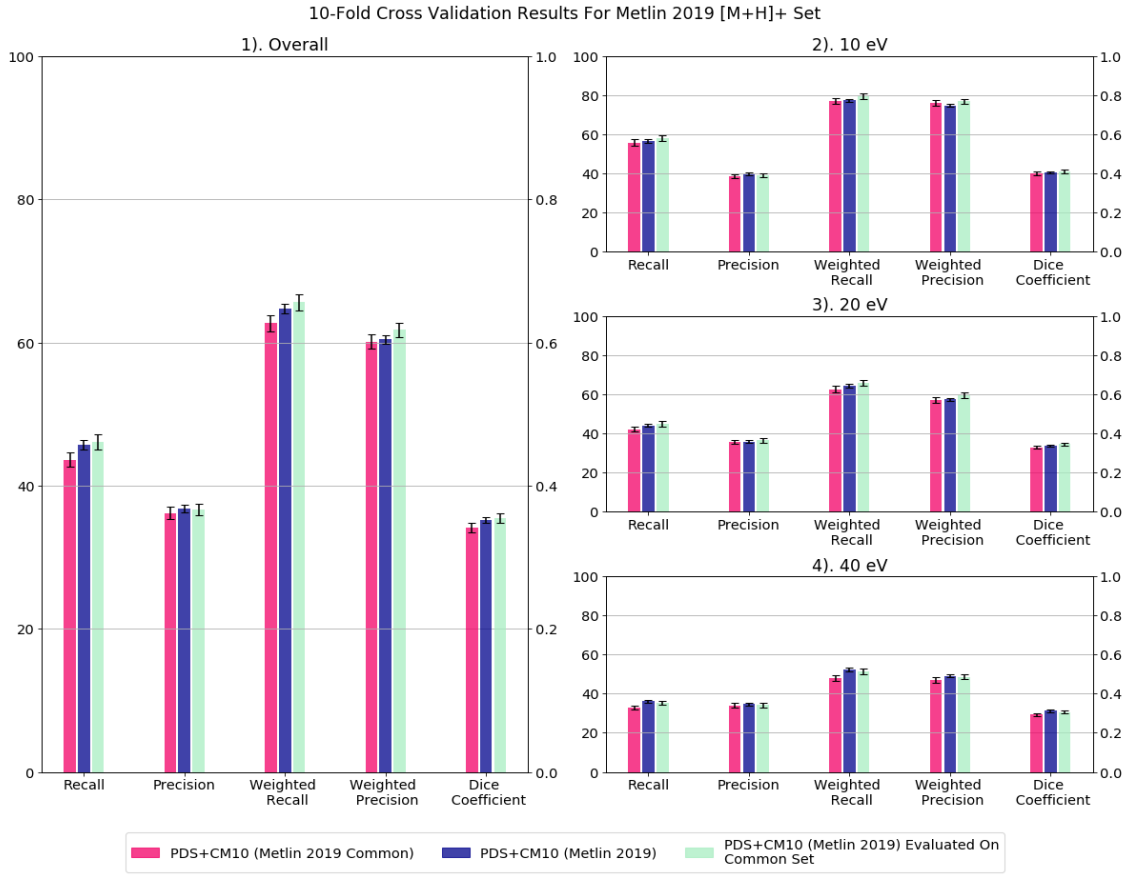


Figure 3.5: Spectrum prediction results for the Metlin metabolites 2019 $[M + H]^+$. Each bar displays mean scores for its metrics with an error bar indicates the 95% confidence interval. The plot on the left presents the overall performance of the model, and plots on the right provide the performance measures for each collision energy. Values used in this figure can be found at Table: A.4.

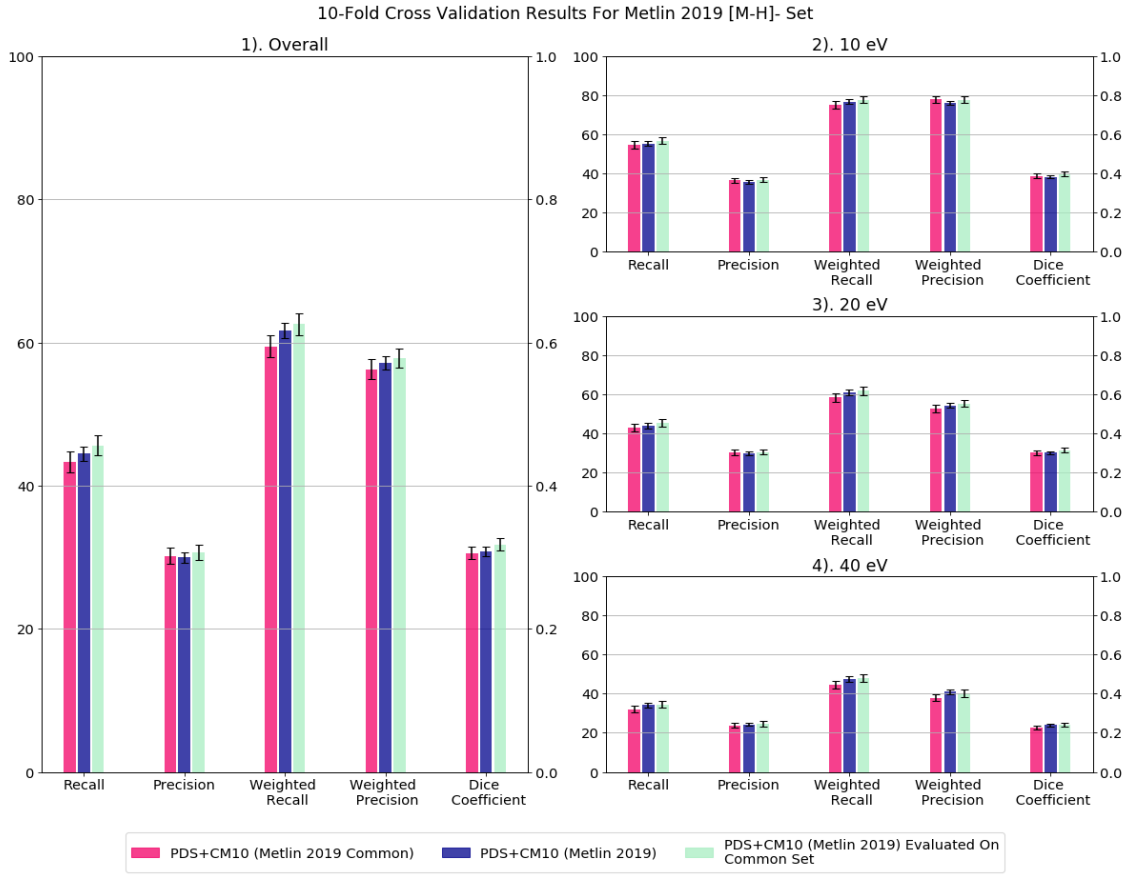


Figure 3.6: Spectrum prediction results for the Metlin metabolites 2019 $[M-H]^-$. Each bar displays mean scores for its metrics with an error bar indicates the 95% confidence interval. The plot on the left presents the overall performance of the model, and the plots on the right provide the performance measures for each collision energy. Values used in this figure can be found at Table: A.5

spectral database with a mass tolerance of 10ppm. This database consists of 160,758 compounds (without derivatives) collected from multiple sources. For details of this database please refers to the CFM-ID 3.0 publication [13].

The model for the $[M+H]^+$ ion mode used in this test was trained on all samples in the Metlin 2019 $[M+H]^+$ samples and the $[M-H]^-$ model was trained on the full Metlin 2019 $[M-H]^-$ set. Both models were trained using the same configuration mentioned in Section: 3.2.1. The scoring metric used is the modified Dot-Product function introduced by CFM-ID 2.0 [2] instead of the Dice coefficient. Compared to the Dice coefficient, the Dot-Product function scoring has less chance to output the same score between different spectra which is a desired property when each sample only has one mass spectrum. Table: 3.2 shows the classification results between the older CFM-ID 2.0 [1], the more recent CFM-ID 3.0 [13], while current model listed as CFM-ID 4.0. When using the latest version of the CFM-ID without the meta data scoring function from CFM-ID 3.0 [13], we managed to classify 147 out of 208 spectra with *in-silico* spectra only. That is only 2 fewer than CFM-ID 3.0, even though the later getting help from both experimentally collected spectra and meta data. It is also a 22.5% improvement in classification accuracy over the original CFM-ID 2.0(*in-silico* only). Furthermore, the result also surpassed the record achieved by MS-Finder [63], [81]. Once incorporated with the meta data scoring mechanism, CFM-ID 4.0 was able to correctly identify 162 MS/MS spectra out of 208 total input spectra, a significant improvement over previous models. In addition, with the help of the same experimental spectra library used in CFM-ID 3.0, our methods managed to achieve an even better classification performance, that is 165 out of 208 compounds had been correctly classified through their MS/MS spectra.

	Top1	Top3	Top10
CFM-ID 2.0 + Candidate Database	120	160	182
CFM-ID 2.0 + Candidate Database + Experimental Spectra	123	171	201
CFM-ID 3.0 + Candidate Database + Experimental Spectra + Meta Data	149	194	204
CFM-ID 4.0 + Candidate Database	147	178	203
CFM-ID 4.0 + Candidate Database + Meta Data	162	186	204
CFM-ID 4.0 + Candidate Database + Experimental Spectra + Meta Data	165	197	207
MS-Finder	146	162	174

Table 3.2: Classification results between CFM-ID 4.0, CFM-ID 3.0, CFM-ID 2.0, and MS-FINDER using CASMI 2016 challenge set(Category 3). In total 208 compound were used in this test, with 127 of them have a positiveESI-MS/MS spectrum collected in the $[M + H]^+$ ion mode , and 82 have negative ESI-MS/MS spectrum collected in the$[M - H]^-$ ion mode.

Chapter 4

Conclusion

In this dissertation, we introduced a novel tensor representation for describing chemical structures and used it to extend the capabilities of Competitive Fragmentation Modeling (CFM) in ESI-MS/MS spectral prediction tasks. We also proposed three sampling methods that can be used during model training process to significantly reduce the training time cost. The empirical results in Chapter 3 examined the *in-silico* spectrum prediction performance of these novel methods on multiple ESI-MS/MS data sets, encompassing a wide range of chemical classes, on both positive and negative mode ionization. While still imperfect, our proposed method outperformed the legacy CFM-ID model by a significant margin across all data sets while using similar training time with newly proposed sampling methods. More importantly, our model used far fewer parameters than the original CFM-ID approaches, which should also speed up the spectrum prediction calculation..

Although our approach has surpassed its predecessor, there is still room for further improvement, both in terms of accuracy and runtime. One future direction is the use of deeper fragmentation graphs and more efficient ways to explore them. A deeper fragmentation graph may be beneficial when handling more complicated chemical structures, however, it will also introduce extra computational complexity. In a similar spirit, another potential area of improvement is to use a prioritized frag-

mentation graph such that chemical bonds linked to certain functional groups will be first dissociated. Finding such functional groups is a huge challenge. Another direction is to extended CFM-ID with even better molecular modelling methods from the current trend in graph neural networks. Last but not least, is to adopt CFM-ID to handle spectrum collected using instruments other than QToF, such as more advanced orbitrap mass spectrometers.

References

[1] F. Allen, R. Greiner, and D. Wishart, "Competitive fragmentation modeling of ESI-MS/MS spectra for putative metabolite identification," en, *Metabolomics*, vol. 11, no. 1, pp. 98–110, Feb. 2015, ISSN: 1573-3882, 1573-3890. DOI: 10.1007/s11306-014-0676-4.

[2] F. Allen, A. Pon, R. Greiner, and D. Wishart, "Computational Prediction of Electron Ionization Mass Spectra to Assist in GC/MS Compound Identification," en, *Anal. Chem.*, vol. 88, no. 15, pp. 7689–7697, Aug. 2016, ISSN: 0003-2700, 1520-6882. DOI: 10.1021/acs.analchem.6b01622.

[3] C. Andrieu and C. Andrieu, "An Introduction to MCMC for Machine Learning," en, p. 39, 2003.

[4] I. Blaženović, T. Kind, J. Ji, and O. Fiehn, "Software Tools and Approaches for Compound Identification of LC-MS/MS Data in Metabolomics," en, *Metabolites*, vol. 8, no. 2, p. 31, May 2018, ISSN: 2218-1989. DOI: 10.3390/metabo8020031.

[5] I. Blaženović, T. Kind, H. Torbašinović, S. Obrenović, S. S. Mehta, H. Tsugawa, T. Wermuth, N. Schauer, M. Jahn, R. Biedendieck, D. Jahn, and O. Fiehn, "Comprehensive comparison of in silico MS/MS fragmentation tools of the CASMI contest: Database boosting is needed to achieve 93% accuracy," en, *Journal of Cheminformatics*, vol. 9, no. 1, Dec. 2017, ISSN: 1758-2946. DOI: 10.1186/s13321-017-0219-x.

[6] E. E. Bolton, Y. Wang, P. A. Thiessen, and S. H. Bryant, "PubChem: Integrated Platform of Small Molecules and Biological Activities," *Annual Reports in Computational Chemistry*, vol. 4, pp. 217–241, 2008.

[7] C. Brouard, H. Shen, K. Dührkop, F. d'Alché-Buc, S. Böcker, and J. Rousu, "Fast metabolite identification with Input Output Kernel Regression," en, *Bioinformatics*, vol. 32, no. 12, pp. i28–i36, Jun. 2016, ISSN: 1367-4803, 1460-2059. DOI: 10.1093/bioinformatics/btw246.

[8] C. Brouard, M. Szafranski, and F. d'Alch"é"-Buc, "Input output kernel regression: Supervised and semi-supervised structured output prediction with operator-valued kernels," *Journal of Machine Learning Research*, vol. 17, no. 176, pp. 1–48, 2016.

[9] K. Burgess, N. Rankin, and S. Weidt, "Metabolomics," en, in *Handbook of Pharmacogenomics and Stratified Medicine*, Elsevier, 2014, pp. 181–205, ISBN: 978-0-12-386882-4. DOI: 10.1016/B978-0-12-386882-4.00010-4.

[10] T. Cajka and O. Fiehn, "Toward Merging Untargeted and Targeted Methods in Mass Spectrometry-Based Metabolomics and Lipidomics," en, *Analytical Chemistry*, vol. 88, no. 1, pp. 524–545, Jan. 2016, ISSN: 0003-2700, 1520-6882. DOI: 10.1021/acs.analchem.5b04491.

[11] B. Curry and D. E. Rumelhart, "MSnet: A Neural Network That Classifies Mass Spectra," en, p. 32, Oct. 1990.

[12] P. de Matos, R. Alcántara, A. Dekker, M. Ennis, J. Hastings, K. Haug, I. Spiteri, S. Turner, and C. Steinbeck, "Chemical Entities of Biological Interest: An update," en, *Nucleic Acids Research*, vol. 38, no. suppl_1, pp. D249–D254, Jan. 2010, ISSN: 0305-1048, 1362-4962. DOI: 10.1093/nar/gkp886.

[13] Y. Djoumbou-Feunang, A. Pon, N. Karu, J. Zheng, C. Li, D. Arndt, M. Gautam, F. Allen, and D. S. Wishart, "CFM-ID 3.0: Significantly Improved ESI-MS/MS Prediction and Compound Identification," en, *Metabolites*, vol. 9, no. 4, p. 72, Apr. 2019, ISSN: 2218-1989. DOI: 10/gf7c7b.

[14] K. Dührkop, H. Shen, M. Meusel, J. Rousu, and S. Böcker, "Searching molecular structure databases with tandem mass spectra using CSI:FingerID," en, *Proc Natl Acad Sci USA*, vol. 112, no. 41, pp. 12 580–12 585, Oct. 2015, ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.1509788112.

[15] W. B. Dunn and D. I. Ellis, "Metabolomics: Current analytical platforms and methodologies," *Trends in Analytical Chemistry*, vol. 24, no. 4, pp. 285–294, 2005.

[16] W. B. Dunn, A. Erban, R. J. M. Weber, D. J. Creek, M. Brown, R. Breitling, T. Hankemeier, R. Goodacre, S. Neumann, J. Kopka, and M. R. Viant, "Mass appeal: Metabolite identification in mass spectrometry-focused untargeted metabolomics," en, *Metabolomics*, vol. 9, no. S1, pp. 44–66, Mar. 2013, ISSN: 1573-3882, 1573-3890. DOI: 10.1007/s11306-012-0434-4.

[17] D. Duvenaud, D. Maclaurin, J. Aguilera-Iparraguirre, R. Gómez-Bombarelli, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, "Convolutional Networks on Graphs for Learning Molecular Fingerprints," en, *arXiv:1509.09292 [cs, stat]*, Sep. 2015. arXiv: 1509.09292 [cs, stat].

[18] A. Eghbaldar, T. P. Forrest, and D. Cabrol-Bass, "Development of neural networks for identification of structural features from mass spectral data.," en, *Analytica Chimica Acta*, p. 19, 1998.

[19] E. A. Feigenbaum and B. G. Buchanan, "DENDRAL and Meta-DENDRAL: Roots of knowledge systems and expert system applications," en, *Artificial Intelligence*, vol. 59, no. 1-2, pp. 233–240, Feb. 1993, ISSN: 00043702. DOI: 10.1016/0004-3702(93)90191-D.

[20] O. Fiehn, "Metabolomics — the link between genotypes and phenotypes," en, in *Functional Genomics*, C. Town, Ed., Dordrecht: Springer Netherlands, 2002, pp. 155–171, ISBN: 978-94-010-3903-1. DOI: 10.1007/978-94-010-0448-0_11.

[21] J. Gasteiger, W. Hanebeck, and K. P. Schulz, "Prediction of mass spectra from structural information," en, *Journal of Chemical Information and Modeling*, vol. 32, no. 4, pp. 264–271, Jul. 1992, ISSN: 1549-9596. DOI: 10.1021/ci00008a001.

[22] J. B. German, B. D. Hammock, and S. M. Watkins, "Metabolomics: Building on a century of biochemistry to guide human health," en, *Metabolomics*, vol. 1, no. 1, pp. 3–9, Mar. 2005, ISSN: 1573-3882, 1573-3890. DOI: 10.1007/s11306-005-1102-8.

[23] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural Message Passing for Quantum Chemistry," en, *arXiv:1704.01212 [cs]*, Apr. 2017. arXiv: 1704.01212 [cs].

[24] J. H. Gross, *Mass Spectrometry: A Textbook*, en, 2. ed. Berlin: Springer, 2011, OCLC: 704764642, ISBN: 978-3-642-10709-2.

[25] A. Grover and J. Leskovec, "Node2vec: Scalable Feature Learning for Networks," en, *arXiv:1607.00653 [cs, stat]*, Jul. 2016. arXiv: 1607.00653 [cs, stat].

[26] C. Guijas, J. R. Montenegro-Burke, X. Domingo-Almenara, A. Palermo, B. Warth, G. Hermann, G. Koellensperger, T. Huan, W. Uritboonthai, A. E. Aisporna, D. W. Wolan, M. E. Spilker, H. P. Benton, and G. Siuzdak, "METLIN: A Technology Platform for Identifying Knowns and Unknowns," en, *Anal. Chem.*, vol. 90, no. 5, pp. 3156–3164, Mar. 2018, ISSN: 0003-2700, 1520-6882. DOI: 10.1021/acs.analchem.7b04424.

[27] M. Heinonen, H. Shen, N. Zamboni, and J. Rousu, "Metabolite identification and molecular fingerprint prediction through machine learning," en, *Bioinformatics*, vol. 28, no. 18, pp. 2333–2341, Sep. 2012, ISSN: 1367-4803, 1460-2059. DOI: 10.1093/bioinformatics/bts437.

[28] M. Heinonen, A. Rantanen, T. Mielikäinen, J. Kokkonen, J. Kiuru, R. A. Ketola, and J. Rousu, "FiD: A software for *ab initio* structural identification of product ions from tandem mass spectrometric data," en, *Rapid Commun. Mass Spectrom.*, vol. 22, no. 19, pp. 3043–3052, Oct. 2008, ISSN: 09514198, 10970231. DOI: 10/fgg7cj.

[29] A. W. Hill and R. J. Mortishire-Smith, "Automated assignment of high-resolution collisionally activated dissociation mass spectra using a systematic bond disconnection approach," en, *Rapid Commun. Mass Spectrom.*, vol. 19, no. 21, pp. 3111–3118, Nov. 2005, ISSN: 0951-4198, 1097-0231. DOI: 10/drw58p.

[30] E. de Hoffmann and V. Stroobant, *Mass Spectrometry: Principles and Applications*, en, 3rd ed. Chichester, West Sussex, England ; Hoboken, NJ: J. Wiley, 2007, ISBN: 978-0-470-03310-4.

[31] H. Horai, M. Arita, S. Kanaya, Y. Nihei, T. Ikeda, K. Suwa, Y. Ojima, K. Tanaka, S. Tanaka, K. Aoshima, Y. Oda, Y. Kakazu, M. Kusano, T. Tohge, F. Matsuda, Y. Sawada, M. Y. Hirai, H. Nakanishi, K. Ikeda, N. Akimoto, T. Maoka, H. Takahashi, T. Ara, N. Sakurai, H. Suzuki, D. Shibata, S. Neumann, T. Iida, K. Tanaka, K. Funatsu, F. Matsuura, T. Soga, R. Taguchi, K. Saito, and T. Nishioka, "MassBank: A public repository for sharing mass spectral data for life sciences.," *Journal of Mass Spectrometry*, vol. 45, no. 7, pp. 703–714, 2010.

[32] F. Hufsky, K. Scheubert, and S. Böcker, "Computational mass spectrometry for small-molecule fragmentation," en, *TrAC Trends in Analytical Chemistry*, vol. 53, pp. 41–48, Jan. 2014, ISSN: 01659936. DOI: 10.1016/j.trac.2013.09.008.

[33] J. Hummel, N. Strehmel, J. Selbig, D. Walther, and J. Kopka, "Decision tree supported substructure prediction of metabolites from GC-MS profiles," en, *Metabolomics*, vol. 6, no. 2, pp. 322–333, Jun. 2010, ISSN: 1573-3882, 1573-3890. DOI: 10.1007/s11306-010-0198-7.

[34] L. J. Kangas, T. O. Metz, G. Isaac, B. T. Schrom, B. Ginovska-Pangovska, L. Wang, L. Tan, R. R. Lewis, and J. H. Miller, "In silico identification software (ISIS): A machine learning approach to tandem mass spectral identification

of lipids," en, *Bioinformatics*, vol. 28, no. 13, pp. 1705–1713, Jul. 2012, ISSN: 1460-2059, 1367-4803. DOI: 10/f332qz.

[35] S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. Riley, "Molecular graph convolutions: Moving beyond fingerprints," en, *Journal of Computer-Aided Molecular Design*, vol. 30, no. 8, pp. 595–608, Aug. 2016, ISSN: 0920-654X, 1573-4951. DOI: 10.1007/s10822-016-9938-8.

[36] A. Kerber, M. Meringer, and C. Rücker, "CASE via MS: Ranking Structure Candidates by Mass Spectra," en, *Croat. Chem. Acta*, p. 16, 2006.

[37] T. Kind, K.-H. Liu, D. Y. Lee, B. DeFelice, J. K. Meissen, and O. Fiehn, "LipidBlast in silico tandem mass spectrometry database for lipid identification," en, *Nat Methods*, vol. 10, no. 8, pp. 755–758, Aug. 2013, ISSN: 1548-7091, 1548-7105. DOI: 10.1038/nmeth.2551.

[38] T. Kind, H. Tsugawa, T. Cajka, Y. Ma, Z. Lai, S. S. Mehta, G. Wohlgemuth, D. K. Barupal, M. R. Showalter, M. Arita, and O. Fiehn, "Identification of small molecules using accurate mass MS/MS search," en, *Mass Spectrometry Reviews*, vol. 37, no. 4, pp. 513–532, Jul. 2018, ISSN: 02777037. DOI: 10.1002/mas.21535.

[39] T. Kind, G. Wohlgemuth, D. Y. Lee, Y. Lu, M. Palazoglu, S. Shahbaz, and O. Fiehn, "FiehnLib: Mass Spectral and Retention Index Libraries for Metabolomics Based on Quadrupole and Time-of-Flight Gas Chromatography/Mass Spectrometry," en, *Anal. Chem.*, vol. 81, no. 24, pp. 10 038–10 048, Dec. 2009, ISSN: 0003-2700, 1520-6882. DOI: 10.1021/ac9019522.

[40] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *Computer Science*, 2014.

[41] C. Klawun and C. L. Wilkins, "Joint Neural Network Interpretation of Infrared and Mass Spectra," en, *J. Chem. Inf. Comput. Sci.*, vol. 36, no. 2, pp. 249–257, Jan. 1996, ISSN: 0095-2338. DOI: 10.1021/ci9501002.

[42] F. Lab, *MassBank of North America*, UC Davis, CA 95618.

[43] G. Landrum *et al.*, *RDKit: Open-Source Cheminformatics*. 2006.

[44] I. Laponogov, N. Sadawi, D. Galea, R. Mirnezami, and K. A. Veselkov, "ChemDistiller: An engine for metabolite annotation in mass spectrometry," en, *Bioinformatics*, vol. 34, no. 12, J. Wren, Ed., pp. 2096–2102, Jun. 2018, ISSN: 1367-4803, 1460-2059. DOI: 10/gc2mpp.

[45] R. K. Lindsay, B. G. Buchanan, E. A. Feigenbaum, and J. Lederberg, "DENDRAL: A case study of the first expert system for scientific hypothesis formation," en, *Artificial Intelligence*, vol. 61, no. 2, pp. 209–261, Jun. 1993, ISSN: 00043702. DOI: 10.1016/0004-3702(93)90068-M.

[46] M. Ludwig, K. Dührkop, and S. Böcker, "Bayesian networks for mass spectrometric metabolite identification via molecular fingerprints," en, *Bioinformatics*, vol. 34, no. 13, pp. i333–i340, Jul. 2018, ISSN: 1367-4803, 1460-2059. DOI: 10.1093/bioinformatics/bty245.

[47] S. Maher, F. P. M. Jjunju, and S. Taylor, "*COLLOQUIUM* : 100 years of mass spectrometry: Perspectives and future trends," en, *Rev. Mod. Phys.*, vol. 87, no. 1, pp. 113–135, Jan. 2015, ISSN: 0034-6861, 1539-0756. DOI: 10/f6xntk.

[48] F. McLafferty, "Tandem mass spectrometry," *Science*, vol. 214, no. 4518, pp. 280–287, 1981, ISSN: 0036-8075. DOI: 10.1126/science.7280693.

[49] *METLIN Mass Spectral Database (bundled with NIST MS/MS)*, Mass Spectrometry Data Center National Institute of Standards and Technology, 2017.

[50] T. Moon, "The expectation-maximization algorithm," en, *IEEE Signal Process. Mag.*, vol. 13, no. 6, pp. 47–60, 1996, ISSN: 10535888. DOI: 10/cf737w.

[51] R. Murray, D. Bender, K. Botham, P. Kennelly, V. Rodwell, and P. Weil, *Harper's Illustrated Biochemistry*, en, 28th ed. Lange Medical Books/McGraw-Hill, 2009, OCLC: 893571654, ISBN: 978-0-07-170197-6.

[52] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," en, p. 8, 2010.

[53] D. L. Pavia, Ed., *Introduction to Spectroscopy*, en, 4th ed. Belmont, CA: Brooks/Cole, Cengage Learning, 2009, ISBN: 978-0-495-11478-9.

[54] H. E. Pence and A. Williams, "ChemSpider: An Online Chemical Information Resource," en, *J. Chem. Educ.*, vol. 87, no. 11, pp. 1123–1124, Nov. 2010, ISSN: 0021-9584, 1938-1328. DOI: 10.1021/ed100697w.

[55] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online Learning of Social Representations," *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '14*, 2014.

[56] J. Qiu, Y. Dong, H. Ma, J. Li, K. Wang, and J. Tang, "Network Embedding as Matrix Factorization: Unifying DeepWalk, LINE, PTE, and node2vec," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining - WSDM '18*, Marina Del Rey, CA, USA: ACM Press, 2018, ISBN: 978-1-4503-5581-0. DOI: 10.1145/3159652.3159706.

[57] D. Ramage, A. N. Rafferty, and C. D. Manning, "Random walks for text semantic similarity," in *Proceedings of the 2009 Workshop on Graph-based Methods for Natural Language Processing - TextGraphs-4*, Suntec, Singapore: Association for Computational Linguistics, 2009, p. 23, ISBN: 978-1-932432-54-1.

[58] L. Ridder, J. J. J. van der Hooft, S. Verhoeven, R. C. H. de Vos, R. van Schaik, and J. Vervoort, "Substructure-based annotation of high-resolution multistage MS ^{n} spectral trees: Substructure-based annotation of MS ^{n} spectral trees," en, *Rapid Commun. Mass Spectrom.*, vol. 26, no. 20, pp. 2461–2471, Oct. 2012, ISSN: 09514198. DOI: 10.1002/rcm.6364.

[59] D. Rogers and M. Hahn, "Extended-Connectivity Fingerprints," en, *J. Chem. Inf. Model.*, vol. 50, no. 5, pp. 742–754, May 2010, ISSN: 1549-9596, 1549-960X. DOI: 10.1021/ci100050t.

[60] C. Ruttkies, E. L. Schymanski, S. Wolf, J. Hollender, and S. Neumann, "MetFrag relaunched: Incorporating strategies beyond in silico fragmentation," en, *Journal of Cheminformatics*, vol. 8, no. 1, Dec. 2016, ISSN: 1758-2946. DOI: 10.1186/s13321-016-0115-9.

[61] K. T. Schütt, P.-J. Kindermans, H. E. Sauceda, S. Chmiela, A. Tkatchenko, and K.-R. Müller, "SchNet: A continuous-filter convolutional neural network for modeling quantum interactions," en, *arXiv:1706.08566 [physics, stat]*, Jun. 2017. arXiv: 1706.08566 [physics, stat].

[62] E. L. Schymanski, M. Meringer, and W. Brack, "Matching Structures to Mass Spectra Using Fragmentation Patterns: Are the Results As Good As They Look?" en, *Analytical Chemistry*, vol. 81, no. 9, pp. 3608–3617, May 2009, ISSN: 0003-2700, 1520-6882. DOI: 10.1021/ac802715e.

[63] E. L. Schymanski, C. Ruttkies, M. Krauss, C. Brouard, T. Kind, K. Dührkop, F. Allen, A. Vaniya, D. Verdegem, S. Böcker, J. Rousu, H. Shen, H. Tsugawa, T. Sajed, O. Fiehn, B. Ghesquière, and S. Neumann, "Critical Assessment of Small Molecule Identification 2016: Automated methods," en, *J Cheminform*, vol. 9, no. 1, p. 22, Dec. 2017, ISSN: 1758-2946. DOI: 10/f9z6rq.

[64] H. Shen, K. Dührkop, S. Böcker, and J. Rousu, "Metabolite identification through multiple kernel learning on fragmentation trees," en, *Bioinformatics*, vol. 30, no. 12, pp. i157–i164, Jun. 2014, ISSN: 1460-2059, 1367-4803. DOI: 10.1093/bioinformatics/btu275.

[65] H. Shen, N. Zamboni, M. Heinonen, and J. Rousu, "Metabolite Identification through Machine Learning— Tackling CASMI Challenge Using FingerID," en, *Metabolites*, vol. 3, no. 2, pp. 484–505, Jun. 2013, ISSN: 2218-1989. DOI: 10/gchm59.

[66] M. Simonovsky and N. Komodakis, "GraphVAE: Towards Generation of Small Graphs Using Variational Autoencoders," en, *arXiv:1802.03480 [cs]*, Feb. 2018. arXiv: 1802.03480 [cs].

[67] C. A. Smith, G. O'maille, E. J. Want, C. Qin, S. A. Trauger, T. R. Brandon, D. E. Custodio, R. Abagyan, and G. Siuzdak, "METLIN : A metabolite mass spectral database," *Therapeutic Drug Monitoring*, vol. 27, no. 6, pp. 747–751, 2005.

[68] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," en, p. 30, 2014.

[69] S. Stein, "Mass Spectral Reference Libraries: An Ever-Expanding Resource for Chemical Identification," en, *Anal. Chem.*, vol. 84, no. 17, pp. 7274–7282, Sep. 2012, ISSN: 0003-2700, 1520-6882. DOI: 10.1021/ac301205z.

[70] S. E. Stein and D. R. Scott, "Optimization and testing of mass spectral library search algorithms for compound identification," en, *J Am Soc Mass Spectrom*, vol. 5, no. 9, pp. 859–866, Sep. 1994, ISSN: 1044-0305, 1879-1123. DOI: 10.1016/1044-0305(94)87009-8.

[71] S. Stephen, *NIST/EPA/NIH Mass Spectral Library with Search Program Data Version: NIST v14*, Mass Spectrometry Data Center National Institute of Standards and Technology, 2014.

[72] ——, *NIST/EPA/NIH Mass Spectral Library with Search Program Data Version: NIST v17*, Mass Spectrometry Data Center National Institute of Standards and Technology, 2017.

[73] M. Stone, "Cross-Validatory Choice and Assessment of Statistical Predictions," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 36, no. 2, pp. 111–133, Jan. 1974, ISSN: 00359246. DOI: 10.1111/j.2517-6161.1974.tb00994.x. [Online]. Available: http://doi.wiley.com/10.1111/j.2517-6161.1974.tb00994.x.

[74] W. Tanaka and M. Arita, "Physicochemical Prediction of Metabolite Fragmentation in Tandem Mass Spectrometry," en, *Mass Spectrometry*, vol. 7, no. 1, A0066–A0066, Jun. 2018, ISSN: 2186-5116, 2187-137X. DOI: 10.5702/massspectrometry.A0066.

[75] R. Tautenhahn, K. Cho, W. Uritboonthai, Z. Zhu, G. J. Patti, and G. Siuzdak, "An accelerated workflow for untargeted metabolomics using the METLIN database," en, *Nat Biotechnol*, vol. 30, no. 9, pp. 826–828, Sep. 2012, ISSN: 1087-0156, 1546-1696. DOI: 10.1038/nbt.2348.

[76] *The LIPID MAPS Lipidomics Gateway*.

[77] G. J. Tortora and M. T. Nielsen, *Principles of Human Anatomy*, en, 12th ed. Hoboken, NJ: John Wiley & Sons, 2012, ISBN: 978-0-470-56705-0.

[78] H. Tsugawa, "Advances in computational metabolomics and databases deepen the understanding of metabolisms," en, *Current Opinion in Biotechnology*, vol. 54, pp. 10–17, Dec. 2018, ISSN: 09581669. DOI: 10.1016/j.copbio.2018.01.008.

[79] K. Uppal, D. I. Walker, K. Liu, S. Li, Y.-M. Go, and D. P. Jones, "Computational Metabolomics: A Framework for the Million Metabolome," en, *Chemical Research in Toxicology*, vol. 29, no. 12, pp. 1956–1975, Dec. 2016, ISSN: 0893-228X, 1520-5010. DOI: 10.1021/acs.chemrestox.6b00179.

[80] A. Vaniya and O. Fiehn, "Using fragmentation trees and mass spectral trees for identifying unknown compounds in metabolomics," en, *TrAC Trends in Analytical Chemistry*, vol. 69, pp. 52–61, Jun. 2015, ISSN: 01659936. DOI: 10.1016/j.trac.2015.04.002.

[81] A. Vaniya, S. N. Samra, M. Palazoglu, H. Tsugawa, and O. Fiehn, "Using MS-FINDER for identifying 19 natural products in the CASMI 2016 contest," en, *Phytochemistry Letters*, vol. 21, pp. 306–312, Sep. 2017, ISSN: 18743900. DOI: 10/gf68zg.

[82] K. Varmuza and W. Werther, "Mass Spectral Classifiers for Supporting Systematic Structure Elucidation †," en, *J. Chem. Inf. Comput. Sci.*, vol. 36, no. 2, pp. 323–333, Jan. 1996, ISSN: 0095-2338. DOI: 10.1021/ci9501406.

[83] D. Verdegem, D. Lambrechts, P. Carmeliet, and B. Ghesquière, "Improved metabolite identification with MIDAS and MAGMa through MS/MS spectral dataset-driven parameter optimization," en, *Metabolomics*, vol. 12, no. 6, p. 98, Jun. 2016, ISSN: 1573-3882, 1573-3890. DOI: 10/ggb5zn.

[84] Y. Wang, G. Kora, B. P. Bowen, and C. Pan, "MIDAS: A Database-Searching Algorithm for Metabolite Identification in Metabolomics," en, *Analytical Chemistry*, vol. 86, no. 19, pp. 9496–9503, Oct. 2014, ISSN: 0003-2700, 1520-6882. DOI: 10.1021/ac5014783.

[85] J. N. Wei, D. Belanger, R. P. Adams, and D. Sculley, "Rapid Prediction of Electron-Ionization Mass Spectrometry using Neural Networks," en, *ACS Cent. Sci.*, vol. 5, no. 4, pp. 700–708, Apr. 2019, ISSN: 2374-7943, 2374-7951. DOI: 10.1021/acscentsci.9b00085. arXiv: 1811.08545.

[86] D. S. Wishart, "Current Progress in computational metabolomics," en, *Briefings in Bioinformatics*, vol. 8, no. 5, pp. 279–293, Jun. 2007, ISSN: 1467-5463, 1477-4054. DOI: 10.1093/bib/bbm030.

[87] D. S. Wishart, D. Tzur, C. Knox, R. Eisner, A. C. Guo, N. Young, D. Cheng, K. Jewell, D. Arndt, S. Sawhney, C. Fung, L. Nikolai, M. Lewis, M.-A. Coutouly, I. Forsythe, P. Tang, S. Shrivastava, K. Jeroncic, P. Stothard, G. Amegbey, D. Block, D. D. Hau, J. Wagner, J. Miniaci, M. Clements, M. Gebremedhin, N. Guo, Y. Zhang, G. E. Duggan, G. D. MacInnis, A. M. Weljie, R. Dowlatabadi, F. Bamforth, D. Clive, R. Greiner, L. Li, T. Marrie, B. D. Sykes, H. J. Vogel, and L. Querengesser, "HMDB: The Human Metabolome Database," en, *Nucleic Acids Research*, vol. 35, no. Database, pp. D521–D526, Jan. 2007, ISSN: 0305-1048, 1362-4962. DOI: 10.1093/nar/gkl923.

[88] D. S. Wishart, Y. D. Feunang, A. Marcu, A. C. Guo, K. Liang, R. Vázquez-Fresno, T. Sajed, D. Johnson, C. Li, N. Karu, Z. Sayeeda, E. Lo, N. Assempour, M. Berjanskii, S. Singhal, D. Arndt, Y. Liang, H. Badran, J. Grant, A. Serra-Cayuela, Y. Liu, R. Mandal, V. Neveu, A. Pon, C. Knox, M. Wilson, C. Manach, and A. Scalbert, "HMDB 4.0: The human metabolome database for 2018," en, *Nucleic Acids Research*, vol. 46, no. D1, pp. D608–D617, Jan. 2018, ISSN: 0305-1048, 1362-4962. DOI: 10.1093/nar/gkx1089.

[89] S. Wolf, S. Schmidt, M. Müller-Hannemann, and S. Neumann, "In silico fragmentation for computer assisted identification of metabolite mass spectra," en, *BMC Bioinformatics*, vol. 11, no. 1, Dec. 2010, ISSN: 1471-2105. DOI: 10.1186/1471-2105-11-148.

[90] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande, "MoleculeNet: A Benchmark for Molecular Machine Learning," en, *arXiv:1703.00564 [physics, stat]*, Mar. 2017. arXiv: 1703.00564 [physics, stat].

=totoc

Glossary

$[\mathbf{M}+\mathbf{H}]^+$ An adduct ion type formed by the interaction of a molecule with a hydrogen atom in a positive charge mode.. vii–ix, 11, 28, 30, 31, 35, 39, 57, 63, 64, 73, 75–77, 79, 81, 83, 84, 100, 103

$[\mathbf{M}-\mathbf{H}]^-$ An adduct ion type formed by the interaction of a molecule with a hydrogen atom in negative charge mode.. vii, ix, x, 11, 63, 64, 77, 79, 82–84, 101, 104, 106

Da Dalton (Da) or unified atomic mass unit is a unit used to measure atomic mass, defined as 1/12 of the mass of Carbon-12 atom in the ground state. 63–65, 68

EI-MS Electron Ionization Mass Spectrometry (EI-MS) is type of mass spectrometry uses Electron Ionization method as ionization source. 18, 26

ESI Electrospray Ionization (ESI). 12

ESI-MS Electrospray Ionization Mass Spectrometry (ESI-MS). 6

ESI-MS/MS Electrospray Ionization Mass Spectrometry/Mass Spectrometry (ESI-MS/MS) is a type of mass spectrometry which uses both Electrospray Ionization and a two-spectrometry tandom setup. vii, 5, 7, 14, 15, 17, 18, 22, 26–28, 63, 64, 66, 68, 84

eV Electronvolt (eV) is a Non-SI unit of energy. 63, 64, 71, 74, 79

mass-to-charge ratio A unit used in mass spectrometry, for a given ion, its $m/z = \frac{\text{Atomic mass}}{\text{Number of Charges}}$. 2, 9, 10, 13, 14, 27, 32, 33, 65

MS/MS Mass SpectrometryMass Spectrometry (MS/MS). 17, 18, 22, 62, 63, 68, 69, 71, 75

ppm Parts Per Million (ppm) is a unit defined as one in a million, 1 ppm is one millionth.. 68

QToF Quadrupole Time-of-Flight (QToF) is a type of mass analyzer arrangement commonly used in tandem mass spectromerty.. 63, 86

ToF Time-of-Flight (ToF) is a type of mass analyzer used in mass spectromerty. 12–14

Appendix A

Additonal Tables

Models	Energies	Recall ± 95% CI	Precision ± 95% CI	Weighted Recall ± 95% CI	Weighted Precision ± 95% CI	Dice ± 95% CI
Baseline SE Model	10 eV	53.65 ± 1.45	30.41 ± 0.83	77.16 ± 1.36	68.24 ± 1.23	0.35 ± 0.01
	20 eV	42.85 ± 1.25	26.38 ± 0.81	61.84 ± 1.58	48.21 ± 1.27	0.29 ± 0.01
	40 eV	36.80 ± 1.19	21.15 ± 0.68	45.26 ± 1.56	32.33 ± 1.03	0.24 ± 0.01
	Overall	44.44 ± 1.02	25.98 ± 0.58	61.42 ± 1.20	49.59 ± 0.89	0.30 ± 0.01
Baseline SE Model	10 eV	52.68 ± 1.47	32.65 ± 0.87	76.64 ± 1.38	70.99 ± 1.23	0.36 ± 0.01
	20 eV	41.45 ± 1.25	26.74 ± 0.83	61.09 ± 1.58	48.31 ± 1.30	0.29 ± 0.01
	40 eV	35.10 ± 1.19	20.77 ± 0.69	43.20 ± 1.56	30.48 ± 0.98	0.24 ± 0.01
	Overall	43.08 ± 1.03	26.72 ± 0.60	60.31 ± 1.21	49.93 ± 0.90	0.29 ± 0.01
RS+CM10	10 eV	48.12 ± 1.50	32.60 ± 0.91	72.30 ± 1.49	66.84 ± 1.32	0.35 ± 0.01
	20 eV	38.20 ± 1.26	26.27 ± 0.86	56.22 ± 1.66	42.27 ± 1.34	0.27 ± 0.01
	40 eV	30.91 ± 1.18	21.01 ± 0.86	39.04 ± 1.59	28.52 ± 1.21	0.23 ± 0.01
	Overall	39.08 ± 1.05	26.63 ± 0.67	55.85 ± 1.21	45.87 ± 0.95	0.28 ± 0.01
RWS+CM10	10 eV	49.59 ± 1.52	43.03 ± 0.99	74.65 ± 1.43	81.03 ± 1.15	0.41 ± 0.01
	20 eV	37.61 ± 1.19	37.52 ± 1.01	59.24 ± 1.57	60.05 ± 1.48	0.34 ± 0.01
	40 eV	32.24 ± 1.15	29.16 ± 1.05	44.63 ± 1.57	40.34 ± 1.41	0.28 ± 0.01
	Overall	39.81 ± 0.98	36.57 ± 0.75	59.51 ± 1.16	60.47 ± 0.94	0.34 ± 0.01
PDS+CM10	10 eV	51.97 ± 1.49	42.20 ± 1.03	76.78 ± 1.37	80.86 ± 1.18	0.42 ± 0.01
	20 eV	41.47 ± 1.20	37.41 ± 1.01	63.24 ± 1.52	60.31 ± 1.42	0.36 ± 0.01
	40 eV	35.01 ± 1.15	32.01 ± 1.06	48.23 ± 1.56	46.40 ± 1.43	0.31 ± 0.01
	Overall	42.82 ± 0.98	37.20 ± 0.78	62.75 ± 1.15	62.53 ± 0.97	0.36 ± 0.01

Table A.1: 10-Fold Cross Validation Results on Meltin 2015 $[M + H]^+$ Set

Models	Energies	Recall ± 95% CI	Precision ± 95% CI	Weighted Recall ± 95% CI	Weighted Precision ± 95% CI	Dice ± 95% CI
Baseline SE Model	10 eV	30.78 ± 1.33	38.66 ± 1.27	75.25 ± 1.65	75.88 ± 1.46	0.30 ± 0.01
	20 eV	17.79 ± 1.29	23.53 ± 1.47	36.24 ± 2.28	34.02 ± 2.10	0.17 ± 0.01
	40 eV	21.65 ± 1.30	27.65 ± 1.30	39.16 ± 2.04	39.85 ± 1.74	0.20 ± 0.01
	Overall	23.41 ± 0.97	29.94 ± 1.00	50.22 ± 1.42	49.92 ± 1.25	0.22 ± 0.01
PDS+CM10	10 eV	30.16 ± 1.36	40.80 ± 1.41	73.55 ± 1.74	79.55 ± 1.50	0.30 ± 0.01
	20 eV	27.20 ± 1.32	36.43 ± 1.50	57.48 ± 2.02	59.07 ± 1.93	0.26 ± 0.01
	40 eV	22.81 ± 1.31	27.24 ± 1.39	40.79 ± 2.03	40.92 ± 1.90	0.20 ± 0.01
	Overall	26.72 ± 1.02	34.82 ± 1.18	57.27 ± 1.45	59.84 ± 1.35	0.26 ± 0.01

Table A.2: 10-Fold Cross Validation Results on Meltin 2015 $[M - H]^-$ Set

Models	Energies	Recall ± 95% CI	Precision ± 95% CI	Weighted Recall ± 95% CI	Weighted Precision ± 95% CI	Dice ± 95% CI
RWS+CM6	10 eV	50.49 ± 1.50	41.14 ± 1.00	75.39 ± 1.42	76.97 ± 1.21	0.40 ± 0.01
	20 eV	38.26 ± 1.19	33.92 ± 0.94	59.53 ± 1.58	56.19 ± 1.42	0.33 ± 0.01
	40 eV	32.75 ± 1.16	25.91 ± 0.91	44.53 ± 1.58	37.43 ± 1.31	0.27 ± 0.01
	Overall	40.50 ± 0.99	33.66 ± 0.70	59.81 ± 1.18	56.86 ± 0.93	0.33 ± 0.01
RWS+CM8	10 eV	50.15 ± 1.51	41.70 ± 0.97	75.37 ± 1.41	78.36 ± 1.18	0.40 ± 0.01
	20 eV	37.96 ± 1.20	36.19 ± 0.99	58.97 ± 1.58	58.42 ± 1.48	0.34 ± 0.01
	40 eV	33.18 ± 1.17	28.12 ± 0.99	45.21 ± 1.59	39.52 ± 1.36	0.28 ± 0.01
	Overall	40.43 ± 0.99	35.34 ± 0.73	59.85 ± 1.18	58.77 ± 0.96	0.34 ± 0.01
RWS+CM10	10 eV	49.59 ± 1.52	43.03 ± 0.99	74.65 ± 1.43	81.03 ± 1.15	0.41 ± 0.01
	20 eV	37.61 ± 1.19	37.52 ± 1.01	59.24 ± 1.57	60.05 ± 1.48	0.34 ± 0.01
	40 eV	32.24 ± 1.15	29.16 ± 1.05	44.63 ± 1.57	40.34 ± 1.41	0.28 ± 0.01
	Overall	39.81 ± 0.98	36.57 ± 0.75	59.51 ± 1.16	60.47 ± 0.94	0.34 ± 0.01
RWS+CM16	10 eV	48.88 ± 1.50	43.46 ± 1.00	75.08 ± 1.40	83.36 ± 1.14	0.40 ± 0.01
	20 eV	38.19 ± 1.20	38.07 ± 0.99	59.86 ± 1.56	62.19 ± 1.45	0.34 ± 0.01
	40 eV	31.96 ± 1.16	29.41 ± 1.05	44.10 ± 1.58	40.44 ± 1.41	0.28 ± 0.01
	Overall	39.67 ± 0.99	36.98 ± 0.74	59.68 ± 1.17	61.99 ± 0.94	0.34 ± 0.01

Table A.3: 10-Fold Cross Validation Results in Comparison Between Difference Sized Matrix Feature.

Models	Energies	Recall ± 95% CI	Precision ± 95% CI	Weighted Recall ± 95% CI	Weighted Precision ± 95% CI	Dice ± 95% CI
PDS+CM10 (Metlin 2019 Common)	10 eV	55.83 ± 1.54	38.72 ± 1.05	77.21 ± 1.38	76.14 ± 1.26	0.40 ± 0.01
	20 eV	42.23 ± 1.30	35.61 ± 1.07	62.77 ± 1.55	57.18 ± 1.44	0.33 ± 0.01
	40 eV	32.86 ± 1.07	34.12 ± 1.15	48.07 ± 1.53	47.01 ± 1.40	0.29 ± 0.01
	Overall	43.64 ± 1.03	36.15 ± 0.87	62.68 ± 1.15	60.11 ± 1.01	0.34 ± 0.01
PDS+CM10 (Metlin 2019)	10 eV	56.67 ± 0.92	39.75 ± 0.64	77.43 ± 0.83	74.79 ± 0.75	0.41 ± 0.01
	20 eV	44.25 ± 0.80	35.96 ± 0.67	64.31 ± 0.93	57.44 ± 0.84	0.34 ± 0.01
	40 eV	36.09 ± 0.69	34.83 ± 0.72	52.47 ± 0.94	49.15 ± 0.83	0.31 ± 0.01
	Overall	45.67 ± 0.64	36.85 ± 0.54	64.74 ± 0.71	60.46 ± 0.60	0.35 ± 0.00
PDS+CM10 (Metlin 2019) Evaluated On Common Set	10 eV	58.08 ± 1.52	39.17 ± 1.03	79.46 ± 1.30	76.87 ± 1.19	0.41 ± 0.01
	20 eV	44.96 ± 1.32	36.55 ± 1.06	65.89 ± 1.50	59.74 ± 1.37	0.35 ± 0.01
	40 eV	35.40 ± 1.09	34.21 ± 1.12	51.52 ± 1.53	48.70 ± 1.33	0.31 ± 0.01
	Overall	46.15 ± 1.04	36.64 ± 0.84	65.62 ± 1.14	61.77 ± 0.97	0.35 ± 0.01

Table A.4: 10-Fold Cross Validation Results For Metlin 2019 $[M + H]^+$ Set.

Models	Energies	Recall ± 95% CI	Precision ± 95% CI	Weighted Recall ± 95% CI	Weighted Precision ± 95% CI	Dice ± 95% CI
PDS+CM10 (Metlin 2019 Common)	10 eV	54.81 ± 1.91	36.46 ± 1.37	75.17 ± 1.89	77.93 ± 1.65	0.39 ± 0.01
	20 eV	42.98 ± 1.83	30.28 ± 1.42	58.59 ± 2.16	52.85 ± 2.05	0.30 ± 0.01
	40 eV	32.15 ± 1.65	23.79 ± 1.28	44.64 ± 2.13	37.99 ± 1.90	0.23 ± 0.01
	Overall	43.32 ± 1.40	30.18 ± 1.09	59.47 ± 1.57	56.26 ± 1.41	0.31 ± 0.01
PDS+CM10 (Metlin 2019)	10 eV	55.30 ± 1.29	35.72 ± 0.92	76.72 ± 1.22	76.07 ± 1.13	0.38 ± 0.01
	20 eV	44.02 ± 1.24	29.84 ± 0.91	61.03 ± 1.44	54.44 ± 1.31	0.30 ± 0.01
	40 eV	34.16 ± 1.14	24.30 ± 0.85	47.41 ± 1.43	40.98 ± 1.28	0.24 ± 0.01
	Overall	44.49 ± 0.98	29.95 ± 0.71	61.72 ± 1.07	57.16 ± 0.94	0.31 ± 0.01
PDS+CM10 (Metlin 2019) Evaluated On Common Set	10 eV	56.85 ± 1.89	36.81 ± 1.39	77.64 ± 1.76	77.77 ± 1.59	0.40 ± 0.01
	20 eV	45.54 ± 1.84	30.57 ± 1.34	61.98 ± 2.11	55.47 ± 1.92	0.32 ± 0.01
	40 eV	34.47 ± 1.65	24.65 ± 1.25	48.07 ± 2.09	40.25 ± 1.87	0.24 ± 0.01
	Overall	45.62 ± 1.41	30.68 ± 1.06	62.56 ± 1.53	57.83 ± 1.37	0.32 ± 0.01

Table A.5: 10-Fold Cross Validation Results For Metlin 2019 $[M-H]^-$ Set.

Appendix B

Additional Figures

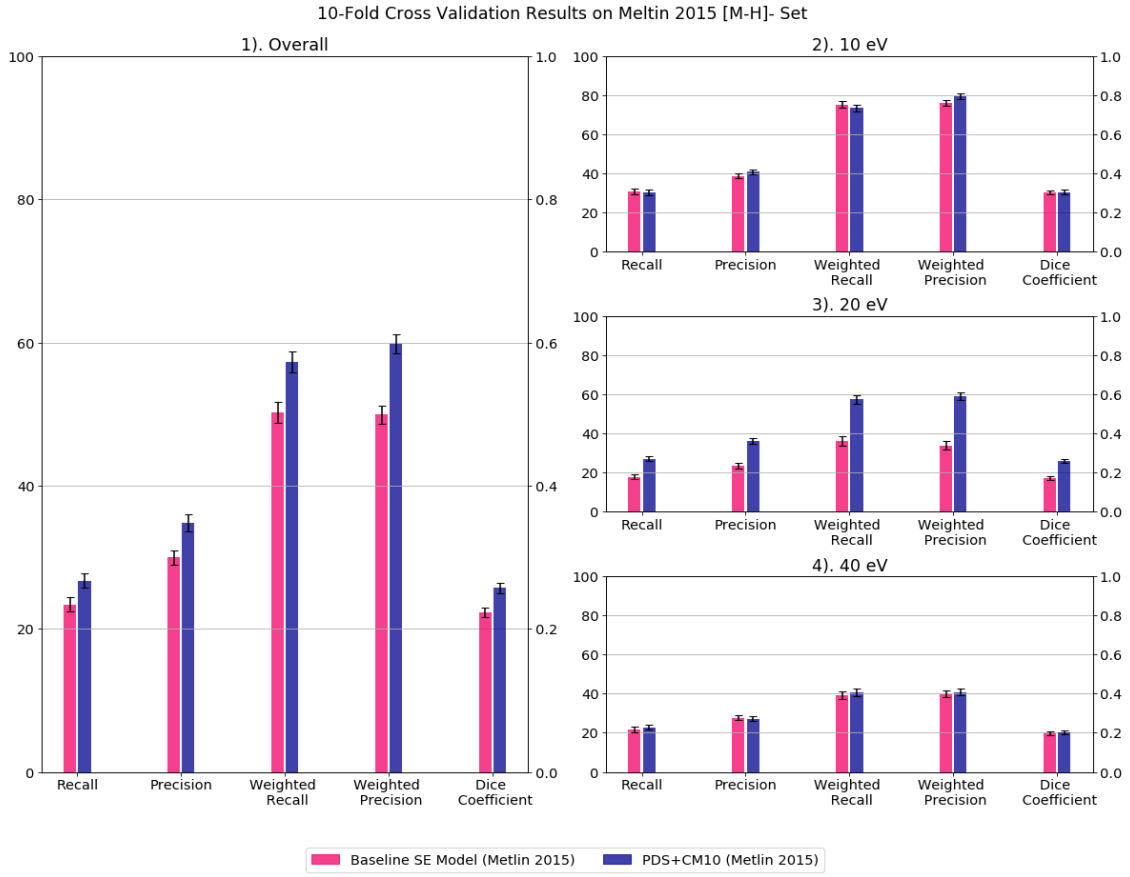


Figure B.1: Spectrum prediction results for the Metlin metabolites 2015 $[M-H]^-$.Each bar displays mean scores for its metrics with an error bar indicates the 95% confidence interval The plot on the left presents the overall performance of the model, and plots on the right provide the performance measures for each collision energy.