analytical chemistry

pubs.acs.org/ac Article

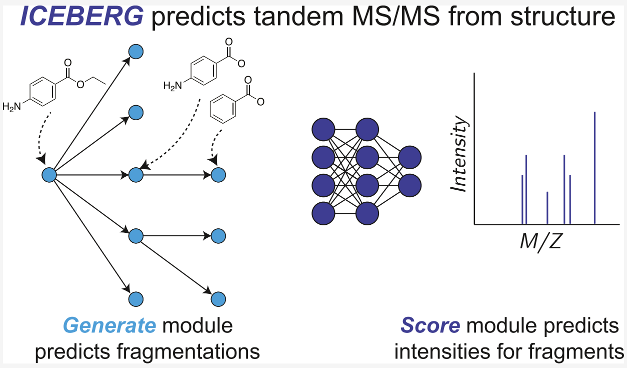
Generating Molecular Fragmentation Graphs with Autoregressive Neural Networks

Samuel Goldman, Janet Li, and Connor W. Coley*

Cite This: Anal. Chem. 2024, 96, 3419−3428

Read Online

ACCESS | Metrics & More | Article Recommendations | SI Supporting Information


ICEBERG predicts tandem MS/MS from structure
H_2N
Intensity
M/Z
Generate module predicts fragmentations
Score module predicts intensities for fragments

ABSTRACT: The accurate prediction of tandem mass spectra from molecular structures has the potential to unlock new metabolomic discoveries by augmenting the community's libraries of experimental reference standards. Cheminformatic spectrum prediction strategies use a "bond-breaking" framework to iteratively simulate mass spectrum fragmentations, but these methods are (a) slow due to the need to exhaustively and combinatorially break molecules and (b) inaccurate as they often rely upon heuristics to predict the intensity of each resulting fragment; neural network alternatives mitigate computational cost but are black-box and not inherently more accurate. We introduce a physically grounded neural approach that learns to predict each breakage event and score the most relevant subset of molecular fragments quickly and accurately. We evaluate our model by predicting spectra from both public and private standard libraries, demonstrating that our hybrid approach offers state-of-the-art prediction accuracy, improved metabolite identification from a database of candidates, and higher interpretability when compared to previous breakage methods and black-box neural networks. The grounding of our approach in physical fragmentation events shows especially great promise for elucidating natural product molecules with more complex scaffolds.

1. INTRODUCTION

Identifying unknown molecules in complex metabolomic or environmental samples is of critical importance to biologists,[1] forensic scientists,[2] and ecologists alike.[3] Tandem mass spectrometry, MS/MS, is the standard analytical chemistry method for analyzing such samples, favored for its speed and sensitivity.[4] In brief, MS/MS metabolomics experiments isolate, ionize, and fragment small molecules, resulting in a characteristic spectrum for each where peaks correspond to molecular subfragments (Figure 1A). Importantly, these experiments are high throughput, leading to thousands of detected spectra per single experiment for complex samples such as human serum.

The most straightforward way to identify an unknown molecule from its fragmentation spectrum is to compare the spectrum to a library of known standards.[5] However, spectral libraries only contain on the order of 10^4 compounds—a drop in the bucket compared to the vast size of biologically relevant chemical space, oft-cited as large as 10^{60}.[6] Of the many tandem spectra deposited into a large community library, 87% still cannot be annotated.[5] The accurate prediction of mass spectra from molecular structures would enable these libraries to be augmented with hypothetical compounds and significantly advance the utility of mass spectrometry for structural elucidation. This paradigm of comparing unknown spectra to putative spectra is well established in the adjacent field of proteomics due to the ease of predicting protein fragmentations.[7]

Because tandem mass spectrometry experiments physically break covalent bonds in a process known as "collision-induced-dissociation" (CID) to create fragments, simulating such fragmentation events computationally is a natural strategy for prediction. Tools from the past decade including MetFrag,[8] MAGMa,[9] and CFM-ID[10,11] use fragmentation rules (based on removing atoms or bonds) and local scoring methods to (a) enumerate molecular fragmentation trees and (b) estimate the intensity at each node in the tree with a mix of heuristic rules and statistical learning (Figure 1B).

However, these combinatorial methods are computationally demanding and often make inaccurate predictions by overestimating the possible fragments (Figure 1B, bottom). We recently found CFM-ID to be far less accurate than black-box neural networks,[12] an observation separately confirmed by Murphy et al.[13] Furthermore, current learned fragmentation models are not easily adapted or scaled to new data sets;

Received: October 15, 2023
Revised: January 8, 2024
Accepted: January 29, 2024
Published: February 13, 2024

https://doi.org/10.1021/acs.analchem.3c04654
Anal. Chem. 2024, 96, 3419−3428

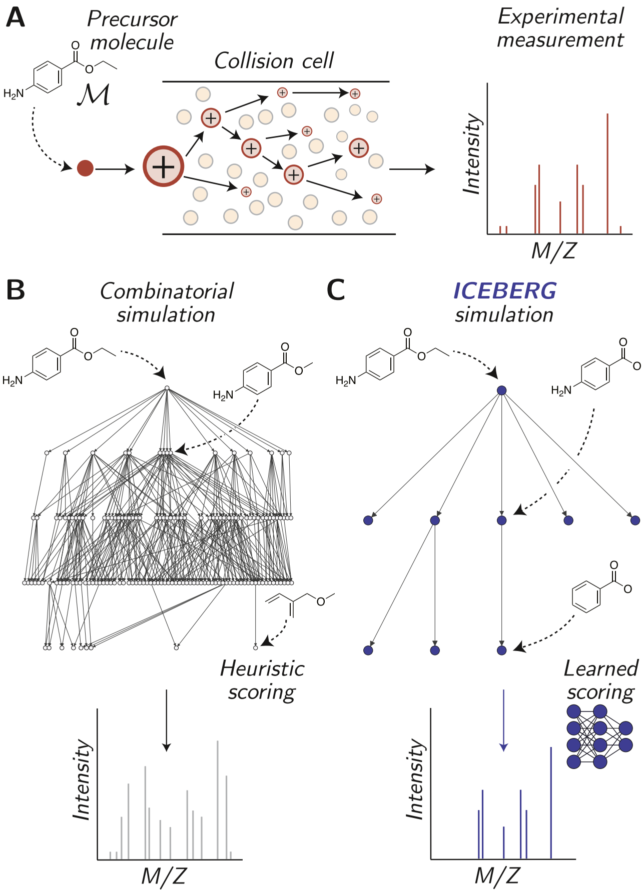


Figure 1. ICEBERG enables the prediction of tandem mass spectra by efficiently navigating the space of possible fragmentation events. (A) Example experimental mass spectrum. An input molecule, benzocaine, is depicted entering a mass spectrometer collision cell and fragmenting. The observation of the resulting charged fragments results in a characteristic spectrum. (B) Combinatorial mass spectrum simulation. The root molecule, benzocaine, is iteratively fragmented by removing atoms or breaking bonds, resulting in a large fragmentation tree. Heuristic rules score nodes in the tree to predict intensities. (C) ICEBERG spectrum simulation. ICEBERG learns to generate only the most relevant substructures. After generating fragments, a neural network module scores the resulting fragments to predict intensities.

Murphy et al. estimate that the leading fragmentation approach, CFM-ID,[10] would require approximately three months on a 64-core machine to train on a dataset containing 300,000 spectra.

Alternative strategies that utilize black-box neural networks to predict MS/MS spectra have been attempted. They encode an input molecule (e.g., as a fingerprint, graph, or 3D structure) and predict either a 1D binned representation of the spectrum,[14−17] or a set of output formulas corresponding to peaks in the spectrum.[12,13,18] While we have demonstrated that predicting chemical formulas provides a fast, accurate, and interpretable alternative to binned representation approaches,[12] the improved accuracy surprisingly did not directly translate to better database retrieval for complex natural product molecules contained within the Global Natural Products Social (GNPS) database.[19] We hypothesized that combining the flexibility of neural networks to learn from experimental MS/MS data in reference libraries with the structural bias of combinatorial fragmentation approaches could lead to increased prediction performance on complex natural product molecules.

Herein, we introduce a hybrid strategy for simulating molecular fragmentation graphs using neural networks, Inferring Collision-induced-dissociation by Estimating Breakage Events and Reconstructing their Graphs (ICEBERG). ICEBERG is a two-part model that simulates probable breakage events (Generate) and scores the resulting fragments using a Transformer architecture (Score) (Figure 1C; details in Figure 2). Our core computational contribution is to leverage previous exhaustive cheminformatics methods for the same task, specifically MAGMa,[9] to build a training data set, from which our model learns to make fast estimates prioritizing only likely bond breakages. In doing so, we lift MAGMa and previous bond-breaking approaches into a neural network space with demonstrable benefits in performance.

We evaluate ICEBERG on two data sets: NPLIB1(GNPS data[19] as used to train the CANOPUS model[20]) and NIST20,[21] which test the model's ability to predict both complex natural products and small organic standard molecules, respectively. We find that ICEBERG increases cosine similarity of predicted spectra by 10% (0.63 vs 0.57) compared to a recent state-of-the-art method on NPLIB1 data. When used to identify molecules in retrospective retrieval studies, ICEBERG leads to a 46% relative improvement (29% vs 20%) in top 1 retrieval accuracy on a challenging natural product data set compared to the next best model tested. ICEBERG is fully open-sourced with pretrained weights alongside other existing prediction baseline methods available on GitHub at https://github.com/samgoldman97/ms-pred.

2. METHODS

2.1. Data Sets. We train our models on the two data sets, NIST20[21] as generated by the National Institute of Standards and NPLIB1 extracted from the GNPS database[19] and prepared previously by Dührkop et al.[20] and Goldman et al.[22] For each spectrum in the data set, we first merge all scans at various collision energies, combine peaks that are within 10^{-4} m/z tolerance from each other, renormalize the resulting spectrum by dividing by the maximum observed intensity, and take the square-root of each intensity. We subset the resulting spectrum to keep the top 50 peaks with an intensity above 0.003. This normalization process is identical to our previous study[12] and emphasizes (a) removing peaks that are likely noise and (b) combining various collision energies. We refer the reader to our previous study[12] for exact details on data set extraction.

To further normalize the data set, for each spectrum, we subtract the mass of the adduct ion from each resulting MS2 peak. Concretely, the precursor molecule is ionized with an adduct ion, for instance, H^+. In this case, the mass of each peak in the spectrum is shifted by the mass of H^+ before proceeding further. In doing so, we normalize against different ionizations. While adduct switching is possible, this assumption allows us to make predictions in a more constrained model output space (i.e., models do not need to predict all combinations of potential adduct masses at each predicted fragment) and allows our models to share output representations across different input adduct types (i.e., potentially learning how to predict rarer fragments more accurately). Furthermore, as described below, our models are capable of predicting various hydrogen-shifted peak intensities that can serve to regularize and correct wrongly annotated adduct switching patterns in a small

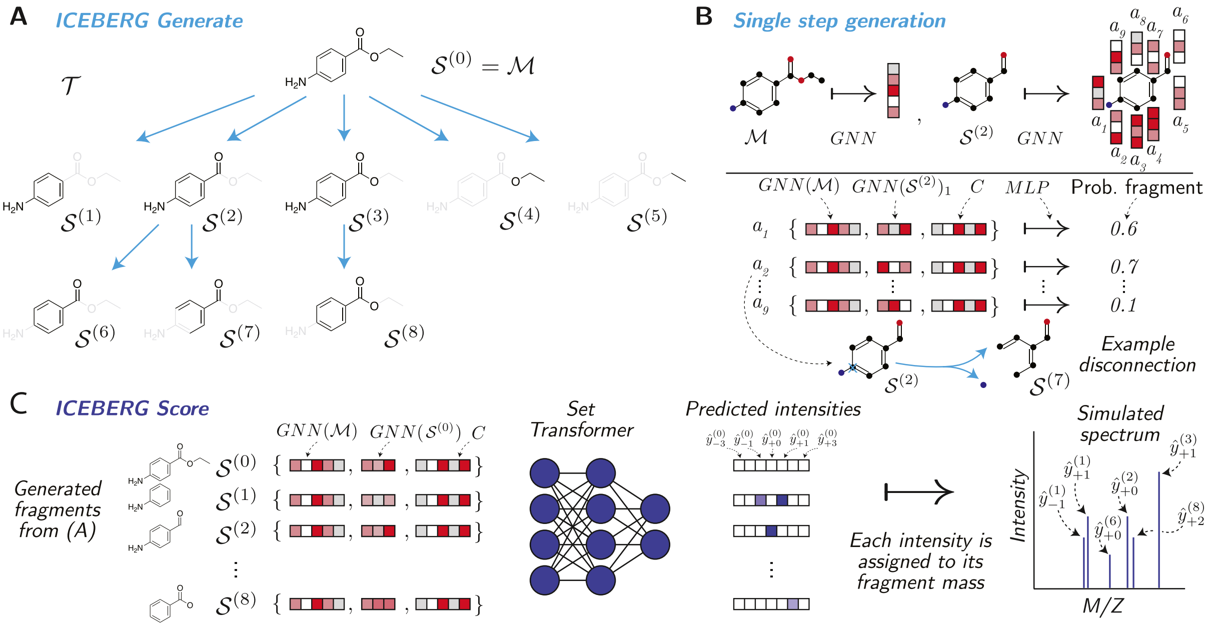


Figure 2. Overview of ICEBERG. (A) Target fragmentation directed acyclic graph (DAG) for an example molecule $\mathcal{M}$, benzocaine. Fragments are colored in black with missing substructures in gray. (B) Example illustration for the generative process at a single step in the DAG generation predicting subfragments of $\mathcal{S}^{(2)}$. The root molecule $\mathcal{M}$, fragment of interest $\mathcal{S}^{(2)}$, and context vector C are encoded and used to predict fragment probabilities at each atom of the fragment of interest. A sample disconnection is shown at atom a_2, resulting in fragment $\mathcal{S}^{(7)}$. (C) ICEBERG Score module. Fragments generated from (A) are encoded alongside the root molecule. A Set Transformer module predicts intensities for each fragment, allowing mass changes corresponding to the loss or gain of hydrogen atoms, resulting in the final predicted mass spectrum.

number of cases (e.g., $[M + H]^+ \rightarrow [M]^+$). We defer more complete incorporation of adduct switching into modeling as a potential direction for future work. We make the simplifying assumption that all peaks are singly charged and use mass and m/z interchangeably. Ultimately, each spectrum $\mathcal{Y}$ can be considered a set of mass, intensity tuples, $\mathcal{Y} = \{(m_0, y_0), (m_1, y_1), \ldots (m_{|\mathcal{Y}|}, y_{|\mathcal{Y}|})\}$.

2.2. Canonical DAG Construction.

We build a custom reimplementation of the MAGMa algorithm[9] to help create explanatory directed acyclic graphs (DAGs) for each normalized and adduct-shifted spectrum.

Given an input molecule,$\mathcal{M}$, MAGMa iteratively breaks each molecule by removing atoms. Each time an atom is removed, multiple fragments may form, from which we keep all fragments of >2 heavy (non-hydrogen) atoms. To prevent the combinatorial explosion of DAG nodes, we use a Weisfeiler-Lehman isomorphism test[23] to generate a unique hash ID of each generated fragment and reject new fragments with hash IDs already observed. When conducting this test, to remain insensitive to how this fragment originated, we hash only the atom identities and bonds in the fragment graph, not the number of hydrogen atoms. For instance, consider an ethane fragment in which the terminal carbon was originally double-bonded to a single neighboring atom in the precursor molecule compared to an ethane fragment in which the terminal carbon was single-bonded to two adjacent atoms in the original precursor—our approach applies the same hash ID to both fragments. The chemical formula and hydrogen status for the fragment are randomly selected from the fragments that require a minimal number of atom removals. Each fragment corresponds to multiple potential m/z observations due to the allowance for hydrogen shifts equal to the number of broken bonds.

After creating the fragmentation graph for the molecule,$\mathcal{M}$, a subset of the fragments are selected to explain each peak in $\mathcal{Y}$, using mass differences of under 20 parts-per-million as the primary filter and the minimal MAGMa heuristic score as a secondary filter. We include nodes along all paths back to the root molecule for each selected fragment. To prune the DAG to select only the most likely paths to each fragment, we design a greedy heuristic. Starting from the lowest level of the DAG, we iteratively select the parent nodes for inclusion in the final DAG that "cover" the highest number of peak-explaining nodes. Finally, the "neutral loss" fragments are added into the DAG, as they provide useful training signals for ICEBERG Generate to learn when to stop fragmenting each molecule. Using these heuristics, it is likely that we select incorrect fragment annotations for a subset of peaks in our training data set. Our intuition is that a deep learning model trained on these data will still be able to make accurate and generalizable predictions to aid in small molecule structure elucidation, even if the level of abstraction is not always true or exactly matched to how an expert chemist may annotate certain MS2 peaks; empirical results demonstrate that this is the case.

2.3. Model Details.

We introduce ICEBERG to predict spectra from input molecules. At a high level, our model is split into two components, where the first model generates molecular substructure (fragment) candidates and the second model predicts intensities and hydrogen rearrangements for each generated fragment. In doing so, we simplify the complexity of both models: the first model outputs a set of likely substructures without needing to handle rearrangements, and the second model predicts potential rearrangement peak shifts and intensities for the constrained set of potential fragments. We describe the methodology for both models separately below.

2.4. DAG Generation Prediction. We train a model to recursively select the atoms around which to break bonds. The model begins by making predictions for the input molecule. We chemiformatically generate the resulting substructures and iteratively apply our model to each. To train this model, we utilize the "ground truth DAG" as described above and train ICEBERG Generate to reconstruct the DAG from an input molecule and input adduct type. We define the root fragment in the DAG as the input molecule $\mathcal{S}^{(0)}$. We can generally define the predicted fragmentation probability for the bonds around the jth atom node in the ith generated fragment in the DAG, $\mathcal{S}^{(i)}$

$$p(\mathcal{F}[\mathcal{S}_j^{(i)}]|\mathcal{S}^{(i)}, \mathcal{M}, C) = g_\theta^{\text{Generate}}(\mathcal{M}, \mathcal{S}^{(i)}, C)_j \quad (1)$$

As can be seen in this equation, predictions are mode atom-wise (i.e., at every single atom for each fragment in the DAG). To make this atom-wise prediction, we encode information about the root molecule, fragment molecule, their difference, their respective chemical formulas, the adduct, and the number of bonds that were broken between the root molecule and fragment. To embed the root molecule, we utilize a gated graph neural network,[24] GNN($\mathcal{M}$), where either average or weighted summations are used to pool embeddings across atoms (specified by a hyperparameter). We utilize the same network to learn representations of the fragment, GNN($\mathcal{S}^{(i)}$) and define GNN$(\mathcal{S}^{(i)})_j$ as the graph neural network-derived embedding of fragment i at the jth atom prior to the pooling operation. For all graph neural networks, a one-hot encoding of the adduct type is also added as atom-wise features alongside the bond types and atom types. We define the chemical formula f for each DAG fragment and specify an encoding, Enc, using the Fourier feature scheme defined in ref 12. We encode the root and ith node of the fragmentation DAG as Enc(f_0) and Enc(f_i), respectively. Lastly, we define one hot vector for the number of bonds broken, b.

All of the encodings described above are concatenated together, and a shallow multilayer perceptron (MLP) ending with a sigmoid function is utilized to predict binary probabilities of fragmentation at each atom.

$$\begin{aligned} &p(\mathcal{F}[\mathcal{S}_j^{(i)}]|\mathcal{S}^{(i)}, \mathcal{M}, C) \\ &= \text{MLP}([\text{GNN}(\mathcal{M}), \text{GNN}(\mathcal{M}) - \text{GNN}(\mathcal{S}^{(i)}), \\ &\text{GNN}(\mathcal{S}^{(i)})_j, \text{Onehot}(b), \text{Enc}(f_i), \text{Enc}(f_0 - f_i)]) \end{aligned} \quad (2)$$

The model is trained to maximize the probability of generating the DAG by minimizing the binary cross entropy loss over each atom for every fragment in an observed spectrum.

2.5. DAG Intensity Prediction. The trained Generate module is used to generate DAGs for each input molecule in the training set by iteratively fragmenting the input molecule with the probability of each fragment computed autoregressively. Once this fragment DAG is created, the ICEBERG Score module learns to predict intensities at each fragment. Because we deferred rearrangement predictions within ICEBERG Generate, we designed ICEBERG Score to simultaneously predict intensities at each generated fragment mass and at a range of masses representing potential hydrogen additions or removals.

We define the node indices for an ordering from each fragment $\mathcal{S}^{(i)}$ back to the root node through its highest likelihood path $\pi[i]$, where $\pi[i, j]$ defines the jth node on this factorization path.

$$p(\mathcal{S}^{(i)}|\mathcal{M},C) = p(\mathcal{S}^{(i)}|\mathcal{S}^{(\pi[i,1])}, \mathcal{M}, C) \prod_{j=1}^{|\pi[i]|} p(\mathcal{S}^{(\pi[i,j])}|\mathcal{S}^{(\pi[i,j+1])}, \mathcal{M}, C) \quad (3)$$

At each step, we maintain only the top 100 most likely fragments in the DAG as a practical consideration until reaching the maximum possible fragmentation depth. To further reduce complexity in the inference step, we maintain the highest-scoring isomers from the DAG. This resulting set of fragments is featurized and passed to a Set Transformer module to generate output values at each fragment. Following the notation from the generative model, we featurize each individual fragment with a shallow MLP to generate hidden representations, h_i

$$\begin{aligned} h_i = \text{MLP}([&\text{GNN}(\mathcal{M}), \\ &\text{GNN}(\mathcal{M}) - \text{GNN}(\mathcal{S}^{(i)}), \\ &\text{GNN}(\mathcal{S}^{(i)}), \\ \text{Onehot}(b), \text{Enc}(f_i&), \text{Enc}(f_0 - f_i)]) \end{aligned} \quad (4)$$

These are subsequently jointly embedded with a Transformer module and used to predict unnormalized intensity weights at each possible hydrogen shift δ alongside an attention weight α to determine how heavily each prediction should be weighted for its specified hydrogen shift. To compute the attention weight, we take a softmax over all prediction indices that fall into the same intensity bin (0.1 resolution), $M(i, \delta)$

$$\hat{y}_\delta^{(i)} = \text{MLP}_{\text{inten}}(\text{Transformer}(h_0, h_1, h_2, ..., h_{|\mathcal{T}|})_i)_\delta \quad (5)$$

$$\begin{aligned} \alpha_\delta^{(i)} = \ &\text{Softmax}_{k \in M(i,\delta)} \\ &(\text{MLP}_{\text{attn}}(\text{Transformer}(h_0, h_1, h_2, ..., h_{|\mathcal{T}|})_k))_{i,\delta} \end{aligned} \quad (6)$$

The final intensity prediction for the bin at mass m is then a weighted sum over all predictions that fall within this mass bin followed by a sigmoid activation function

$$\hat{y}_m = \sigma(\sum_i \sum_\delta \alpha_\delta^{(i)} \hat{y}_\delta^{(i)} \mathcal{I}[M(i, \delta) = m]) \quad (7)$$

The model is trained to maximize the cosine similarity between the predicted spectrum and the ground truth spectrum. While ICEBERG Generate is trained to generate the annotated molecular DAG without any binning assumptions, ICEBERG Score is trained to directly maximize similarity to a vectorized, binned version of the original spectrum to utilize the same loss function as our baseline methods as described below.

2.6. Model Training. All models are implemented and trained using Pytorch Lightning,[25] the Adam optimizer,[26] and the DGL library.[27] Ray[28] is used to complete hyperparameter optimizations over all models and baselines. Models are trained on a single RTX A5000 NVIDIA GPU (CUDA Version 11.6) in under 3 h for each module. A complete list of hyperparameters and their definition can be found in the Supporting Information.

2.7. Baselines. A key component of this work is to extend our robust comparison to previous and contemporary methods.[12] To conduct this benchmarking and specifically emphasize methodological differences rather than data differences, we port and modify code from a number of previous methods including NEIMS,[14] NEIMS with a graph neural network[15] 3DMolMS,[17] SCARF,[12] a modified version of GRAFF-MS we refer to as FixedVocab,[13] MassFormer,[16] and CFM-ID[11] into a single GitHub repository.

Following our previous approach,[12] we emphasize conditioning on the same experimental settings of adduct type across methods for fair comparison, excluding collision energies and instrument types as extensions for future work. We rigorously hyperparameter optimize each method for our data regime and train each model on data splits with the exception of CFM-ID for which retraining is not feasible.[12,13,16]

All model predictions are transformed into binned representations for fair evaluation at a bin resolution of 0.1 from mass 0 to 1500 Da. Further details are included in the Supporting Information.

2.8. Spectral Similarity Evaluations. Both ICEBERG and the presented baseline models are trained to maximize the spectral similarity between each predicted spectrum and the true spectrum as this is well aligned with the retrieval settings and has been used in previous studies. While there are many different variants of cosine similarity calculations,[29−31] we specifically compute a vectorized cosine similarity for any predicted spectrum, $\mathcal{Y}$, and true spectrum, $\hat{\mathcal{Y}}$. We denote each vectorized spectrum and predicted spectrum as $\boldsymbol{s}$ and $\hat{\boldsymbol{s}}$ respectively, in which all intensities are binned into ranges of 0.1 Da from 0 to 1500 Da (a max aggregation function is used to resolve any peaks within the same bin). The vectorized cosine similarity is then defined as

$$\mathrm{sim}(\boldsymbol{s}, \hat{\boldsymbol{s}}) = \frac{\boldsymbol{s}^{\mathrm{T}}\hat{\boldsymbol{s}}}{\|\boldsymbol{s}\|_2 \|\hat{\boldsymbol{s}}\|_2} = \frac{\sum_{i=1}^{|\boldsymbol{s}|} s_i \hat{s}_i}{\sqrt{\sum_{j=1}^{|\boldsymbol{s}|} s_j^2}\sqrt{\sum_{j=1}^{|\boldsymbol{s}|} \hat{s}_j^2}} \tag{8}$$

While this distance metric forces the conversion of exact mass fragment predictions into binned values for ICEBERG predictions, it ensures that we avoid unfairly penalizing the binned spectrum models with which we compare our model. Such baseline models are incapable of predicting exact mass spectra outputs and can only be trained on this variant of the cosine similarity.[14−17]

3. RESULTS

3.1. ICEBERG Is Trained as a Two-Stage Generative and Scoring Model. *3.1.1. Learning to Generate Likely Substructures.* ICEBERG simulates a mass spectrum by generating the substructure fragments from an initial molecule that are most likely to be generated by collision-induced dissociation and subsequently measured in the mass spectrometer. We define an input molecule $\mathcal{M}$ (benzocaine example shown in Figure 2A) and its observed spectrum $\mathcal{Y}$, which is a set of intensities at various mass-to-charge values (m/z), termed peaks. Each peak represents one or more observed molecular fragment.

A core question then is how to generate the set of potential fragments. These fragments can be sampled from the many possible substructure options, $\mathcal{S}^{(i)} \in (N^{(i)}, E^{(i)}) \subseteq \mathcal{M}$, where the set of nodes and edges in substructures are subsets of the atoms and bonds in the original molecule, $\mathcal{M} \in (N, E)$. Most often, this sampling is accomplished by iteratively and exhaustively removing edges or atoms from the molecular graph, creating a fragmentation graph $\mathcal{T} \in (\mathcal{S}, \mathcal{E})$, where all the nodes in this graph are themselves substructures of the original molecule $\mathcal{S} = \{\mathcal{S}^{(0)}, \mathcal{S}^{(1)}, \ldots \mathcal{S}^{(|\mathcal{T}|)}\}$[8−10] (Figure 1b). However, such a combinatorial approach leads to thousands of molecular fragments, making this procedure slow and complicating the second step of estimating intensity values for all of the enumerated fragments.

We eschew combinatorial generation and instead leverage a graph neural network to parametrize breakage events of the molecule, defining the Generate module of ICEBERG (Figure 2A,B). ICEBERG Generate predicts the fragmentation graph iteratively, beginning with just the root of the graph $\mathcal{S}^{(0)} = \mathcal{M}$, borrowing ideas from autoregressive tree generation.[32,33] At each step in iterative expansion, the model $g_\theta^{\mathrm{Generate}}$ assigns a probability of fragmentation to each atom j in the current substructure fragment $\mathcal{S}^{(i)}$, $p(F[\mathcal{S}_j^{(i)}])$. Learned atom embeddings are concatenated alongside embeddings of the root molecule and a context vector C containing metadata, such as the ionization adduct type, to make this prediction. An illustrative example can be seen for the fragment $\mathcal{S}^{(2)}$ in Figure 2B. Atom a_2 has the highest predicted probability, so this atom is then removed from the graph, leading to the subsequent child node $\mathcal{S}^{(7)}$ (Figure 2B). Importantly, the number of child fragments is determined by how many disjoint molecular graphs form upon removal of the jth node from the molecular graph; in this example, fragments $\mathcal{S}^{(1)}$ and $\mathcal{S}^{(4)}$ originate from the same fragmentation event of $\mathcal{S}^{(0)}$ (Figure 2A).

In this way, ICEBERG predicts breakages at the level of each atom, following the convention of MAGMa[9] rather than each bond as is the convention with CFM-ID.[10] We strategically used this abstraction to ensure that all fragmentation events lead to changes in heavy-atom composition. We acknowledge that this formulation does not currently allow for the prediction of skeletal rearrangements and recombinations, which might further improve the model's ability to explain fragmentation spectra. We refer the reader to Model details for a full description of the model $g_\theta^{\mathrm{Generate}}(\mathcal{M}, \mathcal{S}^{(i)}, C)_j$, graph neural network architectures, and context vector inputs.

While this defines a neural network for generation, we must also specify an algorithm for how to train this network. Spectral library data sets contain only molecule and spectrum pairs, but not the directed acyclic graph (DAG) $\mathcal{T}$ of the molecule's substructures that generated the spectrum. We infer an explanatory substructure identity of each peak for model training by leveraging previous combinatorial enumeration methods, specifically MAGMa.[9] For each training molecule and spectrum pair, $(\mathcal{M}, \mathcal{Y})$, we modify MAGMa to enumerate all substructures of $\mathcal{M}$ up to a depth of 3 sequential fragmentation events. We filter enumerated structures to include only those with m/z values appearing in the final spectrum, thereby defining a data set suitable for training ICEBERG Generate (see Canonical DAG Construction). As a result, each paired example, $(\mathcal{M}, \mathcal{Y})$, in the training data set is labeled with an estimated fragmentation DAG. Generate learns

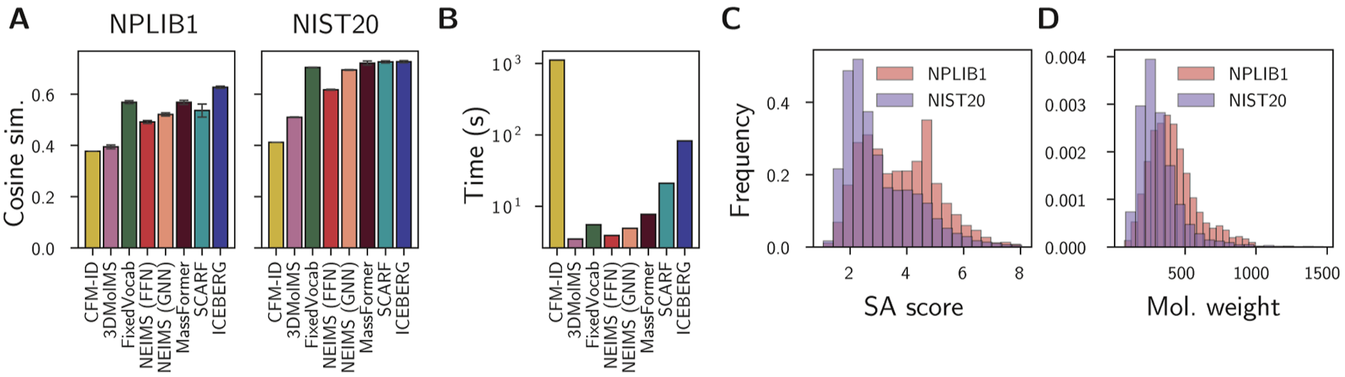


Figure 3. ICEBERG predictions are highly accurate. (A) Cosine similarities to true spectra on NPLIB1 (left) and NIST20 respectively (right) for CFM-ID,[10] 3DMolMS,[17] FixedVocab,[13] NEIMS (FFN),[14] NEIMS (GNN),[15] MassFormer,[16] SCARF,[12] and ICEBERG. Error bars are computed as 1.96 times the standard error of the mean across three random seeds on a single test set split. (B) Time required to predict spectra for 100 molecules randomly sampled from NIST20 on a single CPU, including the time to load models into memory. (C,D) Comparison of NPLIB1 and NIST20 molecules in terms of synthetic accessibility (SA) score[38] and molecular weight (Mol. weight).

from these DAGs to generate only the most relevant and probable substructures for a molecule of interest (see Model Details).

3.1.2. Predicting Substructure Intensities. After generating a set of potential substructure fragments, we employed a second module, ICEBERG Score, to predict their intensities (Figure 2C). Importantly, this design decision enables our models to consider two important physical phenomena: (i) neutral losses and (ii) mass shifts due to hydrogen rearrangements and isotope effects.

Because we elect to fragment molecules at the level of atoms (see Model Details), multiple substructures can result from a single fragmentation event. In physical experiments, not all of these substructure fragments will be observed; when fragmentation events occur in the collision cell, one fragment often retains the charge of the parent, while the other is uncharged and therefore undetected, termed a "neutral loss". By deferring the prediction of intensities to a second module, ICEBERG Generate does not need to predict or track whether structures are ionized, greatly reducing the complexity of the fragmentation DAG.

In addition to the occurrence of neutral losses, molecules often undergo complex rearrangements in the collision cell, leading to bond order promotions or reductions (e.g., spurious formation of double bonds when a single bond breaks to maintain valence), the most classic of which is the McLafferty rearrangement.[34,35] While other approaches attempt to model and estimate where these rearrangements occur using hand-crafted rules,[10] we instead adopt the framework of Ridder et al.[9] to consider hydrogen tolerances. That is, for each generated molecular substructure $\mathcal{S}^{(i)}$ we consider the possibility that this fragment is observed not only at its mass, but also at masses shifted by discrete hydrogen masses, ±δH. This design choice also simplifies ICEBERG Generate by deferring the specification of hydrogen counts to the second model. In addition to accounting for a mass shift of 1 hydrogen, such flexibility also allows the model to predict the common M + 1 isotopes for carbon- and nitrogen-containing compounds.

Mathematically, we define a neural network, g_θ^{Score} that predicts multiple intensities for each fragment $\hat{y}_\delta^{(i)}$ corresponding to different hydrogen shifts, δ

$$\hat{y}_\delta^{(i)} = g_\theta^{\text{Score}}(\mathcal{M}, \mathcal{S}^{(i)}, \mathcal{T}, C)_\delta \tag{9}$$

In practice, we predict up to 13 intensities at each fragment (i.e., {+0H, ±1H, ..., ±6H}). For each individual subfragment, the tolerance is further restricted to the number of bonds broken, most often less than 6. We then take the masses of all fragments, perturb them by the corresponding hydrogen or isotope shifts, and aggregate them into a set of unique m/z peaks by summing the intensities of the perturbed fragments with the same m/z value.

To consider all fragments simultaneously in a permutation-invariant manner, g_θ^{Score} is parametrized as a Set Transformer network.[36,37] We train this second module to maximize the cosine similarity between the ground truth spectrum and the predicted spectrum after converting the set of substructures and intensities to m/z peaks.

At test time, we generate the top 100 most likely fragments from ICEBERG Generate and predict intensities for these fragments and their possible hydrogen shifts using the ICEBERG Score. We find that this tree size allows our model to consider sufficiently many potential fragments while maintaining a speed advantage over previous fragmentation approaches.

3.2. ICEBERG Enables Highly Accurate Spectrum Prediction. We evaluate ICEBERG on its ability to accurately simulate positive ion mode mass spectra for both natural product-like molecules and smaller organic molecules under 1500 Da. Using the data cleaning pipeline from,[12] we compile a public natural products data set NPLIB1 with 10,709 spectra (8,553 unique structures)[19,20,22] as well as a gold standard chemical library NIST20 with 35,129 spectra (24,403 unique structures).[21] We note that NPLIB1 was previously named "CANOPUS", renamed here to disambiguate the data from the tool CANOPUS.[20] Both data sets are split into structurally disjoint 90%/10% train/test splits, with 10% of the training data reserved for model validation (see Data Sets). We compare ICEBERG against an expansive suite of contemporary and competitive methods;[10,12−17] all methods excluding CFM-ID are reimplemented, hyperparameter optimized, and trained on equivalent data splits, further described in the Supporting Information.

To measure performance, we calculate the average cosine similarity between each predicted spectrum and the true spectrum, as cosine similarity is widely used to cluster mass

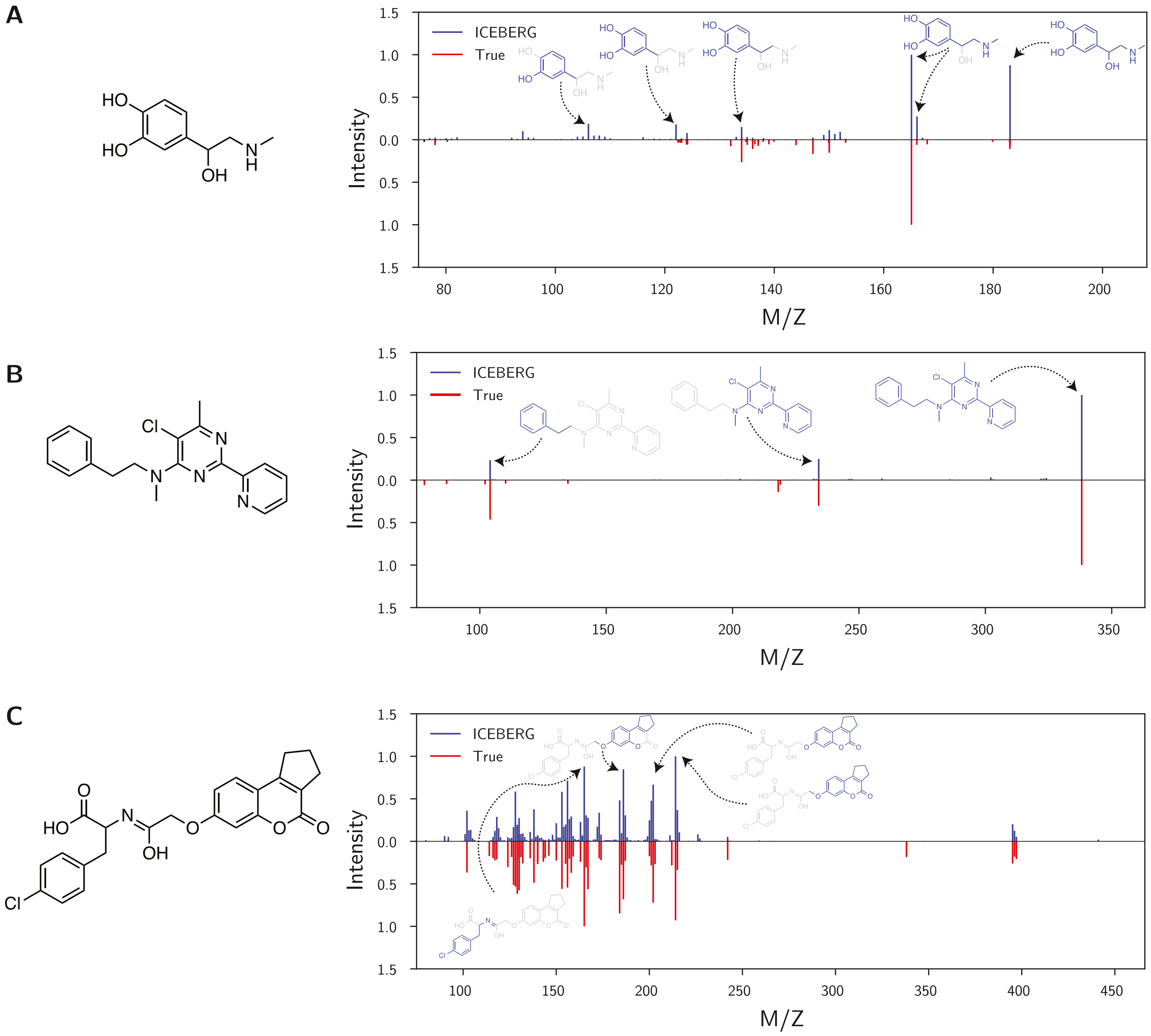


Figure 4. Examples of predicted spectra from ICEBERG. Predictions are shown as generated by ICEBERG trained on NPLIB1 for select test set examples GNPS:CCMSLIB00003137969 (A), MoNA:001659 (B), and GNPS:CCMSLIB00000080524 (C). The input molecular structures are shown (left); fragmentation spectra are plotted (right) with predictions (top, blue) and ground truth spectra (bottom, red). Molecular fragments are shown inset. Spectra are plotted with m/z shifted by the mass of the precursor adduct. All examples shown were not included in the model training set.

spectra in molecular networking.[39] We find that ICEBERG outperforms the next method MassFormer on the natural product-focused data set (Figure 3A and Table S3). ICEBERG achieves an average cosine similarity of 0.627, compared to MassFormer and FixedVocab which each achieved a cosine similarity of 0.568—a 10% improvement.

Surprisingly, however, this boost in performance extends only marginally to the gold standard data set, NIST20. ICEBERG, while still outperforming binned spectrum prediction approaches (i.e., NEIMS[14]) on this data set, is nearly equivalent to SCARF (0.727 v. 0.726).[12] Still, our model performs substantially better than CFM-ID and uses only a fraction of the computational resources (Figure 3B). Unlike previous physically inspired models, because ICEBERG only samples the most relevant fragments from chemical space, it requires just over 1 CPU second per spectrum.

We hypothesize that the discrepancy in performance improvement between NPLIB1 and NIST20 may be partially explained by differences in the chemical spaces they cover. Many molecules within NPLIB1 are natural products with more complicated chemical scaffolds. To characterize this, we analyzed the distributions for both the synthetic accessibility (SA) score[38,40] (Figure 3C) and molecular weight (Figure 3D), both proxies for molecular complexity. In concordance with our hypothesis, we find that SA scores and molecular weight are substantially higher on NPLIB1 than NIST20: NPLIB1 has an average SA score of 3.75, compared to 3.01 for NIST20; the data sets have average molecular weights of 413 and 317 Da, respectively.

3.3. Model Explanations of Observed Peaks Are Consistent with Chemistry Intuition. In addition to accurate predictions, a key benefit of simulating fragmentation events is that predictions can be interpreted, even for highly complex molecules. Each predicted peak from ICEBERG is directly attributed to a fragment of the predicted molecule.

By inspecting certain patterns and examples, we found expected broken bonds. Weaker bonds such as carbon−oxygen and carbon−nitrogen bonds tend to be more reliably breakable, compared to carbon−carbon bonds and more complex ring breakages (Figure 4A). A second strength of using fragmentation-based models can be seen in Figure 4B, where, despite the heteroaromatic ring structures, our model is still able to correctly predict peak intensities by predicting a small number of carbon−nitrogen breakages.

Further alignment can be seen within the intensity prediction module. Because ICEBERG predicts multiple intensities for each substructure corresponding to hydrogen shifts, 2 peaks can be present when a single bond breaks. In the fragmentation example of Figure 4A, the most intense peak is estimated at the mass shift of −1H from the original fragment, indicating that ICEBERG correctly recognizes the hydroxyl group will likely leave as neutral H_2O and result in a hydrogen rearrangement. We include additional randomly selected prediction examples in the Supporting Information.

3.4. Fragmentation Simulations Lead to Improved Structural Elucidation. We next demonstrate that ICEBERG improves the structural elucidation of unknown molecules using reference libraries of model-predicted spectra. We design a retrospective evaluation using our labeled data to resemble the prospective task of spectral lookup within libraries. For each test spectrum, we extract up to 49 "decoy" isomers from PubChem[41] with the highest Tanimoto similarity to the true molecular structure. The consideration of up to 50 isomers mimics the realistic elucidation setting, as an unknown spectrum can yield clues regarding certain properties of its source molecule (e.g., computed using MIST,[22] CSI:FingerID,[42] or molecular networking[39]), which narrows the chemical space of possible molecules to a smaller, more relevant set. We predict the fragmentation spectrum for each isomer, and for each model, we rank these possible matches by their spectral similarity to the spectrum of interest and compute how often the true molecule is found within the top *k* ranked isomers for different values of *k*.

We find that ICEBERG improves upon the next best model by a margin of 9% accuracy (a nearly 46% relative improvement) in top 1 retrieval accuracy for the NPLIB1 data set (Figure 5A and Table S5). Previous models with high spectrum prediction accuracies have struggled on this task due to their poor ability to differentiate structurally similar isomers.[12] Our structure-based model appears to excel in retrieval and may have out-of-domain robustness that is beneficial to this task.

While the effect for top 1 retrieval accuracy on the NIST20 data set is not pronounced (i.e., tied with MassFormer), ICEBERG outperforms the next best model by an absolute margin of over 5% (a 7.5% relative improvement) at top 10 accuracy (Figure 5B and Table S4). These results underscore the real-world utility of ICEBERG to identify unknown molecules of interest.

To additionally evaluate the robustness of our model on the retrieval task in the real world, we further examine the accuracy of the highest performing methods to identify the correct molecule on the positive mode spectra from the recent Critical Assessment of Small Molecule Identification 2022 (CASMI22) competition.[43] Because CASMI22 is a natural product identification challenge, we test models trained on NPLIB1 and test the four models with the highest top 1 retrieval accuracy on NPLIB1: ICEBERG, MassFormer, NEIMS (FFN), and FixedVocab. We find, once again, that ICEBERG outperforms the other models tested on the retrieval task with 12.9% accuracy compared to the next best NEIMS (FFN) method, which achieves a top 1 accuracy of 8.6%. Absolute performance is low, but this is expected because (1) we utilize a challenging PubChem[44] retrieval library and (2) the retrieval accuracy for all entrants was relatively low; the entries submitted by the state-of-the-art forward prediction model, CFM-ID,[11] only correctly predicted 2D structures for 26 of the 303 entries (8.6%). Full results and comparison interpretations are discussed in the Supporting Information.

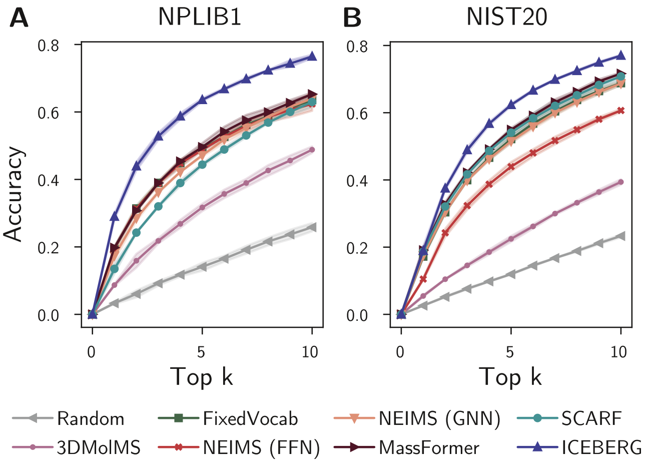


Figure 5. ICEBERG enables improved spectrum retrieval over other methods on both NPLIB1 (A) and NIST20 (B) compared to other spectrum prediction models. Top *k* retrieval accuracy is computed by ranking a list of 49 additional candidates by putative cosine similarity as predicted by the model and determining the fraction of times that the true molecule is within the first *k* entries. A region around each line is shaded using an upper and lower bound of 1.96 times the standard error of the mean across three training seeds.

3.5. Challenging, Nonrandom Data Splits Better Explain Retrieval Performance. The strong performance on the retrieval task, particularly for increasing values of *k* on NIST20, suggests that ICEBERG is able to generalize well to decoys not appearing in the training set and to account for how structural changes should affect fragmentation patterns. While encouraging, we observed only minor increases in cosine similarity accuracy when predicting spectra using NIST20 (Figure 3 and Table S2).

To try to explain this apparent discrepancy, we reevaluate prediction accuracy on a more challenging data set split. We retrain all models on the NIST20 utilizing a Murcko scaffold split of the data[45] with smaller scaffold clusters (i.e., more unique compounds) placed in the test set. This split demonstrates that molecules in the test set will be more distant and less similar to the training set, probing the ability of each model to generalize in a more stringent setting than our previous random split.

In the strict scaffold split evaluation, the improved accuracy of the ICEBERG over existing models is more apparent (Table 1). We find that ICEBERG outperforms MassFormer and

Table 1. Comparing the Accuracy of Spectrum Prediction on NIST20 using Random (Easier) or Scaffold (Harder) Split.

NIST20	Cosine sim.	
	Random split	Scaffold split
CFM-ID	0.412	0.411
3DMolMS	0.510	0.466
FixedVocab	0.704	0.658
NEIMS (FFN)	0.617	0.546
NEIMS (GNN)	0.694	0.643
MassFormer	0.721	0.682
SCARF	0.726	0.669
ICEBERG	**0.727**	**0.699**

SCARF by 0.017 and 0.03−2% and 4% improvements, respectively. These results suggest that, particularly for standard libraries with more homogeneous molecules, more challenging scaffold split evaluations may yield performance metrics that better correlate with performance on the structural elucidation problem (retrieval).

4. DISCUSSION

We have proposed a physically grounded mass spectrum prediction strategy that we term ICEBERG. From a computational perspective, this integration of neural networks into fragmentation prediction is enabled by (a) bootstrapping MAGMa to construct fragmentation trees on which our model is trained, (b) posing the tree generation step as a sequential prediction over atoms, and (c) predicting multiple intensities at each generated fragment with a second module to account for hydrogen rearrangements and isotopic peaks. By learning to generate fragmentation events, ICEBERG is able to accurately predict mass spectra, yielding especially strong improvements for natural product molecules under evaluation settings of both spectrum prediction and retrieval.

ICEBERG establishes new state-of-the-art performance for these tasks, yet there are some caveats that we wish to highlight. First, while we learn to generate molecular substructures to explain each peak, there are no guarantees that they are the correct physical explanations given the number of potential equivalent-mass atom and bond rearrangements that could occur and our decision to train ICEBERG Score to maximize a vectorized cosine similarity. Second, while we achieve increased accuracy, this comes at a higher computational cost of roughly 1 CPU second per molecule, nearly an order of magnitude more than other neural approaches like SCARF.[12] Future work will consider more explicitly how to synergize fragment- and formula-prediction approaches to achieve higher accuracy and speed. In addition to model architecture modifications, we anticipate model accuracy improvements from modeling other covariates such as collision energy, adduct switching, instrument type, and even jointly modeling MS/MS with other analytical chemistry measurements such as FTIR.[46]

The discovery of unknown metabolites and molecules is rapidly expanding our knowledge of potential medical targets,[47] the effects of environmental toxins,[48] and the diversity of biosynthetically accessible chemical space.[49] We envision exciting possibilities to apply our new model to expand the discovery of novel chemical matter from complex mixtures.

■ ASSOCIATED CONTENT

Data Availability Statement

All codes to replicate experiments, train new models, and load pretrained models are available at https://github.com/samgoldman97/ms-pred. Pretrained models are available for download or as workflows through the GNPS2 platform[19] with an up-to-date link provided within the README file of our released GitHub code.

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.analchem.3c04654.

Extended description of results, including exact values in tables, additional evaluation of the CASMI22 data set, additional spectrum prediction examples, further elaboration on the baselines utilized, and hyperparameters tested (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Connor W. Coley − *Department of Chemical Engineering and Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, United States;* orcid.org/0000-0002-8271-8723; Email: ccoley@mit.edu

Authors

Samuel Goldman − *Computational and Systems Biology, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, United States;* orcid.org/0000-0002-3928-6873

Janet Li − *Harvard College, Harvard University, Cambridge, Massachusetts 02138, United States*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.analchem.3c04654

Author Contributions

S.G. and J.L. jointly wrote the software. S.G. conducted experiments. S.G. and C.W.C. conceptualized the project and wrote the manuscript. C.W.C. supervised the work.

Funding

This work was supported by the Machine Learning for Pharmaceutical Discovery and Synthesis consortium. S.G. was additionally supported by the MIT-Takeda Fellowship program.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

We thank John Bradshaw, Priyanka Raghavan, David Graff, Fanwang Meng, other members of the Coley Research Group, and Michael Murphy for helpful discussions, as well as Lucas Janson for feedback on earlier iterations of this idea. We thank Mingxun Wang for feedback and helpful suggestions regarding both the method and manuscript.

■ REFERENCES

(1) Wishart, D. S. *Physiol. Rev.* **2019**, *99*, 1819−1875.

(2) Szeremeta, M.; Pietrowska, K.; Niemcunowicz-Janica, A.; Kretowski, A.; Ciborowski, M. *Int. J. Mol. Sci.* **2021**, *22*, 3010.

(3) Bundy, J. G.; Davey, M. P.; Viant, M. R. *Metabolomics* **2009**, *5*, 3−21.

(4) Neumann, S.; Böcker, S. *Anal. Bioanal. Chem.* **2010**, *398*, 2779−2788.
(5) Bittremieux, W.; Wang, M.; Dorrestein, P. C. *Metabolomics* **2022**, *18*, 94.
(6) Kirkpatrick, P.; Ellis, C. *Nature* **2004**, *432*, 823.
(7) Frewen, B. E.; Merrihew, G. E.; Wu, C. C.; Noble, W. S.; MacCoss, M. J. *Anal. Chem.* **2006**, *78*, 5678−5684.
(8) Wolf, S.; Schmidt, S.; Müller-Hannemann, M.; Neumann, S. *BMC Bioinf.* **2010**, *11*, 148.
(9) Ridder, L.; van der Hooft, J. J.; Verhoeven, S. *Mass Spectrom.* **2014**, *3*, S0033.
(10) Allen, F.; Greiner, R.; Wishart, D. *Metabolomics* **2015**, *11*, 98−110.
(11) Wang, F.; Liigand, J.; Tian, S.; Arndt, D.; Greiner, R.; Wishart, D. S. *Anal. Chem.* **2021**, *93*, 11692−11700.
(12) Goldman, S.; Bradshaw, J.; Xin, J.; Coley, C. W. Prefix-Tree Decoding for Predicting Mass Spectra from Molecules. *Advances in Neural Information Processing Systems*; Curran Associates, Inc., 2023; Vol. *37*.
(13) Murphy, M.; Jegelka, S.; Fraenkel, E.; Kind, T.; Healey, D.; Butler, T. Efficiently predicting high resolution mass spectra with graph neural networks. *Proceedings of the 40th International Conference on Machine Learning*, 2023.
(14) Wei, J. N.; Belanger, D.; Adams, R. P.; Sculley, D. *ACS Cent. Sci.* **2019**, *5*, 700−708.
(15) Zhu, H.; Liu, L.; Hassoun, S. *arXiv* **2020**, arXiv:2010.04661.
(16) Young, A.; Wang, B.; Röst, H. *arXiv* **2021**, arXiv:2111.04824.
(17) Hong, Y.; Li, S.; Welch, C. J.; Tichy, S.; Ye, Y.; Tang, H. *Bioinformatics* **2023**, *39*, btad354.
(18) Zhu, R. L.; Jonas, E. *Anal. Chem.* **2023**, *95*, 2653−2663.
(19) Wang, M.; Carver, J. J.; Phelan, V. V.; Sanchez, L. M.; Garg, N.; Peng, Y.; Nguyen, D. D.; Watrous, J.; Kapono, C. A.; Luzzatto-Knaan, T.; et al. *Nat. Biotechnol.* **2016**, *34*, 828−837.
(20) Dührkop, K.; Nothias, L.-F.; Fleischauer, M.; Reher, R.; Ludwig, M.; Hoffmann, M. A.; Petras, D.; Gerwick, W. H.; Rousu, J.; Dorrestein, P. C.; et al. *Nat. Biotechnol.* **2021**, *39*, 462−471.
(21) NIST. *Tandem Mass Spectral Library*, 2020.
(22) Goldman, S.; Wohlwend, J.; Stražar, M.; Haroush, G.; Xavier, R. J.; Coley, C. W. *Nat. Mach. Intell.* **2023**, *5*, 965−979.
(23) Weisfeiler, B.; Leman, A. *nti Series* **1968**, *2*, 12−16.
(24) Li, Y.; Tarlow, D.; Brockschmidt, M.; Zemel, R. Gated Graph Sequence Neural Networks. *International Conference on Learning Representations*, 2016.
(25) Falcon, W. *The PyTorch Lightning Team*; PyTorch Lightning, 2019.
(26) Kingma, D. P.; Ba, J. *arXiv* **2015**, arXiv:1412.6980.
(27) Wang, M.; Zheng, D.; Ye, Z.; Gan, Q.; Li, M.; Song, X.; Zhou, J.; Ma, C.; Yu, L.; Gai, Y.; Xiao, T.; He, T.; Karypis, G.; Li, J.; Zhang, Z. *arXiv* **2019**, arXiv:1909.01315.
(28) Liaw, R.; Liang, E.; Nishihara, R.; Moritz, P.; Gonzalez, J. E.; Stoica, I. *arXiv* **2018**, arXiv:1807.05118.
(29) Demuth, W.; Karlovits, M.; Varmuza, K. *Anal. Chim. Acta* **2004**, *516*, 75−85.
(30) Huber, F.; Ridder, L.; Verhoeven, S.; Spaaks, J. H.; Diblen, F.; Rogers, S.; van der Hooft, J. J. J. *PLoS Comput. Biol.* **2021**, *17*, No. e1008724.
(31) Li, Y.; Kind, T.; Folz, J.; Vaniya, A.; Mehta, S. S.; Fiehn, O. *Nat. Methods* **2021**, *18*, 1524−1531.
(32) Bradshaw, J.; Paige, B.; Kusner, M. J.; Segler, M.; Hernández-Lobato, J. M. Barking up the right tree: an approach to search over molecule synthesis DAGs. *Advances in Neural Information Processing Systems*; Curran Associates, Inc., 2020; Vol. *33*, pp 6852−6866.
(33) Gao, W.; Mercado, R.; Coley, C. W. Amortized Tree Generation for Bottom-up Synthesis Planning and Synthesizable Molecular Design. *International Conference on Learning Representations*, 2021.
(34) Demarque, D. P.; Crotti, A. E.; Vessecchi, R.; Lopes, J. L.; Lopes, N. P. *Nat. Prod. Rep.* **2016**, *33*, 432−455.
(35) McLafferty, F. W. *Anal. Chem.* **1959**, *31*, 82−87.
(36) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *Advances in Neural Information Processing Systems*; Curran Associates, Inc., 2017; Vol. *30*, pp 5998−6008.
(37) Lee, J.; Lee, Y.; Kim, J.; Kosiorek, A.; Choi, S.; Teh, Y. W. Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks. *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp 3744−3753.
(38) Ertl, P.; Schuffenhauer, A. *J. Cheminf.* **2009**, *1*, 8.
(39) Nothias, L.-F.; Petras, D.; Schmid, R.; Dührkop, K.; Rainer, J.; Sarvepalli, A.; Protsyuk, I.; Ernst, M.; Tsugawa, H.; Fleischauer, M.; Aicheler, F.; Aksenov, A. A.; Alka, O.; Allard, P.-M.; Barsch, A.; Cachet, X.; Caraballo-Rodriguez, A. M.; Da Silva, R. R.; Dang, T.; Garg, N.; Gauglitz, J. M.; Gurevich, A.; Isaac, G.; Jarmusch, A. K.; Kameník, Z.; Kang, K. B.; Kessler, N.; Koester, I.; Korf, A.; Le Gouellec, A.; Ludwig, M.; Martin H, C.; McCall, L.-I.; McSayles, J.; Meyer, S. W.; Mohimani, H.; Morsy, M.; Moyne, O.; Neumann, S.; Neuweger, H.; Nguyen, N. H.; Nothias-Esposito, M.; Paolini, J.; Phelan, V. V.; Pluskal, T.; Quinn, R. A.; Rogers, S.; Shrestha, B.; Tripathi, A.; van der Hooft, J. J. J.; Vargas, F.; Weldon, K. C.; Witting, M.; Yang, H.; Zhang, Z.; Zubeil, F.; Kohlbacher, O.; Böcker, S.; Alexandrov, T.; Bandeira, N.; Wang, M.; Dorrestein, P. C. *Nat. Methods* **2020**, *17*, 905−908.
(40) Huang, K.; Fu, T.; Gao, W.; Zhao, Y.; Roohani, Y.; Leskovec, J.; Coley, C. W.; Xiao, C.; Sun, J.; Zitnik, M. *Nat. Chem. Biol.* **2022**, *18*, 1033−1036.
(41) Kim, S.; Thiessen, P. A.; Bolton, E. E.; Chen, J.; Fu, G.; Gindulyte, A.; Han, L.; He, J.; He, S.; Shoemaker, B. A.; et al. *Nucleic Acids Res.* **2016**, *44*, D1202−D1213.
(42) Dührkop, K.; Shen, H.; Meusel, M.; Rousu, J.; Böcker, S. *Proc. Natl. Acad. Sci. U.S.A.* **2015**, *112*, 12580−12585.
(43) CASMI. *Critical Assessment of Small Molecule Identification*, 2022.
(44) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Thiessen, P. A.; Yu, B.; et al. *Nucleic Acids Res.* **2019**, *47*, D1102−D1109.
(45) Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M.; et al. *J. Chem. Inf. Model.* **2019**, *59*, 3370−3388.
(46) Fine, J. A.; Rajasekar, A. A.; Jethava, K. P.; Chopra, G. *Chem. Sci.* **2020**, *11*, 4618−4630.
(47) Quinn, R. A.; Melnik, A. V.; Vrbanac, A.; Fu, T.; Patras, K. A.; Christy, M. P.; Bodai, Z.; Belda-Ferre, P.; Tripathi, A.; Chung, L. K.; Downes, M.; Welch, R. D.; Quinn, M.; Humphrey, G.; Panitchpakdi, M.; Weldon, K. C.; Aksenov, A.; da Silva, R.; Avila-Pacheco, J.; Clish, C.; Bae, S.; Mallick, H.; Franzosa, E. A.; Lloyd-Price, J.; Bussell, R.; Thron, T.; Nelson, A. T.; Wang, M.; Leszczynski, E.; Vargas, F.; Gauglitz, J. M.; Meehan, M. J.; Gentry, E.; Arthur, T. D.; Komor, A. C.; Poulsen, O.; Boland, B. S.; Chang, J. T.; Sandborn, W. J.; Lim, M.; Garg, N.; Lumeng, J. C.; Xavier, R. J.; Kazmierczak, B. I.; Jain, R.; Egan, M.; Rhee, K. E.; Ferguson, D.; Raffatellu, M.; Vlamakis, H.; Haddad, G. G.; Siegel, D.; Huttenhower, C.; Mazmanian, S. K.; Evans, R. M.; Nizet, V.; Knight, R.; Dorrestein, P. C. *Nature* **2020**, *579*, 123−129.
(48) Tian, Z.; Zhao, H.; Peter, K. T.; Gonzalez, M.; Wetzel, J.; Wu, C.; Hu, X.; Prat, J.; Mudrock, E.; Hettinger, R.; Cortina, A. E.; Biswas, R. G.; Kock, F. V. C.; Soong, R.; Jenne, A.; Du, B.; Hou, F.; He, H.; Lundeen, R.; Gilbreath, A.; Sutton, R.; Scholz, N. L.; Davis, J. W.; Dodd, M. C.; Simpson, A.; McIntyre, J. K.; Kolodziej, E. P. *Science* **2021**, *371*, 185−189.
(49) Doroghazi, J. R.; Albright, J. C.; Goering, A. W.; Ju, K.-S.; Haines, R. R.; Tchalukov, K. A.; Labeda, D. P.; Kelleher, N. L.; Metcalf, W. W. *Nat. Chem. Biol.* **2014**, *10*, 963−968.