Neural Graph Matching Improves Retrieval Augmented Generation in Molecular Machine Learning

Runzhong Wang [* 1]   **Rui-Xi Wang** [* 1]   **Mrunali Manjrekar** [1]   **Connor W. Coley** [1]

Abstract

Molecular machine learning has gained popularity with the advancements of geometric deep learning. In parallel, retrieval-augmented generation has become a principled approach commonly used with language models. However, the optimal integration of retrieval augmentation into molecular machine learning remains unclear. Graph neural networks stand to benefit from clever matching to understand the structural alignment of retrieved molecules to a query molecule. Neural graph matching offers a compelling solution by explicitly modeling node and edge affinities between two structural graphs while employing a noise-robust, end-to-end neural network to learn affinity metrics. We apply this approach to mass spectrum simulation and introduce MARASON, a novel model that incorporates neural graph matching to enhance a fragmentation-based neural network. Experimental results highlight the effectiveness of our design, with MARASON achieving 28% top-1 accuracy, a substantial improvement over the non-retrieval state-of-the-art accuracy of 19%. Moreover, MARASON outperforms both naive retrieval-augmented generation methods and traditional graph matching approaches. Code is publicly available at https://github.com/coleygroup/ms-pred.

1. Introduction

Enhancing neural networks with task-relevant factual knowledge has shown great promise in advancing knowledge-intensive applications, a technique widely recognized as retrieval-augmented generation (RAG) (Lewis et al., 2020). In scientific domains, where the demand for accurate and reliable models is prominent, retrieval-augmented generation has achieved significant success. Recent progress in small-molecule research further highlights the potential of retrieval-augmented generation, which is also the main focus of this paper.[1] This approach has been demonstrated to enhance the accuracy and robustness of various molecular machine learning applications, including structure-based drug design (Zhang et al., 2024; Huang et al., 2024), fragment-based drug discovery (Lee et al., 2024), and monomer design for advanced materials (Buehler, 2024).

We assume a database exists with pairs of molecular structures and their properties of interest. For a new structure of interest, references in the database can be straightforwardly retrieved with cheminformatic tools such as molecular fingerprints (Morgan, 1965) and Tanimoto similarity (Bajusz et al., 2015). Domain experts benefit from such features in databases by searching for pairs of structures to correlate differences in structures with differences in properties, even leading to entire subfields like matched molecular pair analysis (Griffen et al., 2011). Therefore, we expect that augmenting a structure-property relationship model with structurally similar molecules and their associated properties should improve the accuracy of predictions.

However, designing an effective neural network architecture for retrieval augmentation presents significant challenges. Simple approaches, such as direct concatenation of reference information to the input of the neural network, often yield minimal or no improvements (see Table 3). We hypothesize that this is due to models' inabilities to adequately address differences between the retrieved references and the target molecule. We propose that employing atom-level or fragment-level matching mechanisms in an end-to-end network would allow for reference information to be used more effectively.

Graph matching, which addresses node-level correspondences across multiple graphs, naturally emerges as a suitable approach to this challenge. It explicitly incorporates structural matching by formulating both node-wise and edge-wise graph affinity scores into the quadratic assign-

---

[*]Equal contribution [1]Massachusetts Institute of Technology, Cambridge, MA, United States. Correspondence to: Connor W. Coley <ccoley@mit.edu>.

[1]"Small molecules" are defined in this work by a mass upper limit of 1500 Da, enforced in all experiments.

matched fragments by traditional graph matching
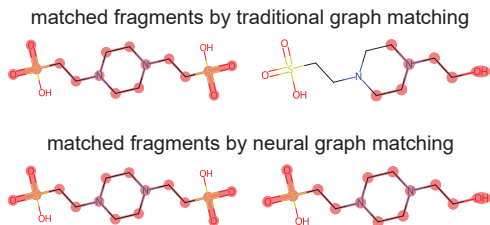
matched fragments by neural graph matching

Figure 1. Comparison of graph matching results between a traditional algorithm (Cho et al., 2010) and the neural graph matching module in this paper. Neural graph matching is more robust especially when the matched structures are not identical. More visualization results can be found in Fig. 6, 7 in Appendix.

ment problem (Lawler, 1963). In molecular tasks, graph matching recapitulates what domain experts might do when comparing two molecules. In practice, traditional graph matching methods typically define the graph affinity using pre-established metrics, such as Gaussian kernels based on Euclidean distance. Such a predefined affinity metric has a significant limitation: its limited expressivity makes traditional graph matching methods vulnerable to random noise and not robust to possible ambiguities in retrieved structures (Wang et al., 2022). To address this robustness challenge, a new class of neural graph matching methods has emerged, which learns the affinity metric and the solver module in an end-to-end manner (Zanfir & Sminchisescu, 2018; Wang et al., 2020; Li et al., 2019). As a result, neural graph matching provides a compelling design choice for retrieval augmentation, as shown in Fig. 1. Chemists are able to draw analogies between molecules beyond simple maximum common substructure analysis and consider, for example, the relationship between non-equivalent but functionally similar (i.e., isosteric) functional groups. By integrating this chemistry-inspired view of graph matching with the expressivity and adaptability of neural networks, neural graph matching represents an effective framework for aligning target structures with, and thus learning from, their retrieved counterparts.

To this end, we implement and validate our design strategy in a prominent application of molecular machine learning: the neural simulation of spectra from tandem mass spectrometry (MS/MS). MS/MS is an analytical chemistry technique that generates profiles from unknown molecules which are used as diagnostic signatures in structure elucidation workflows. This approach has wide-ranging applications in chemistry and biology, including biomarker discovery (Dang et al., 2009), metabolomics (Quinn et al., 2020), and environmental science (Tian et al., 2021), among others. An MS/MS simulator is designed to take molecular structures as input and predict their mass spectra, i.e., a set of mass-over-charge (m/z) values and peak intensities. Simulating these spectra can accelerate the structural elucidation pipeline, illuminating the spaces of so-called

"metabolite dark matter" (Bittremieux et al., 2022).

Our contributions are:

1) We present an effective design strategy for retrieval-augmented generation in molecular machine learning with neural graph matching. Graph matching techniques incorporate both node-wise and edge-wise affinities explicitly, providing a principled way of assigning node-to-node or fragment-to-fragment alignment between the reference structure and the target structure[2] that emulates how domain experts leverage retrieved references. By further introducing learning-based affinity metrics and differentiable matching layers, we improve the overall efficacy of matching and the resulting accuracy in applications.

2) We present matching-aware retrieval augmented spectrometry oracle with neural networks (MARASON), an implementation of neural graph matching-based retrieval-augmented generator for mass spectra, as shown in Fig. 2. MARASON is built upon the ICEBERG (Goldman et al., 2024) model and incorporates design principles of neural graph matching (Wang et al., 2022; Yu et al., 2020). In MARASON, we retrieve structures with known reference spectra from the training dataset (NIST, 2020). We exploit ICEBERG's fragmentation model to annotate reference spectrum peaks with fragments and form a fragmentation directed acyclic graph (DAG) with peak annotations for both the target and reference structures. We then match reference fragments to target fragments to inform the second-stage model's intensity prediction. We incorporate nested graph neural networks (GNNs) (Zhang & Li, 2021; Scarselli et al., 2008) to encode (a) structural information of each fragment and (b) hierarchical information of the fragmentation DAG into node embeddings, building a neural graph matching network for MS/MS simulation.

3) Our experimental evaluation on standard benchmarks demonstrates state-of-the-art accuracy on the mass spectrum simulation task, outperforming both RAG and non-RAG baselines and thus validating the effectiveness of our design strategy. Specifically, on the NIST (2020) dataset, we improve the top-1 retrieval accuracy from 19% to 28%. Ablations demonstrate that MARASON outperforms a naive RAG baseline (by concatenating the retrieved spectrum to the model input) as well as other matching methods using traditional graph matching solver (Cho et al., 2010) or a simple linear matching solver (Kuhn, 1955), emphasizing the value of its differentiable, end-to-end neural graph matching module. Given the broad applicability of mass spectrometry in biological and chemical campaigns, the improved accuracy of our model lends great potential for accelerating molecular discovery by expanding standard mass spectrum

---

[2]In this paper, we refer to all structures and spectra retrieved from the database as "reference", and the structures and spectra of interest as "target".
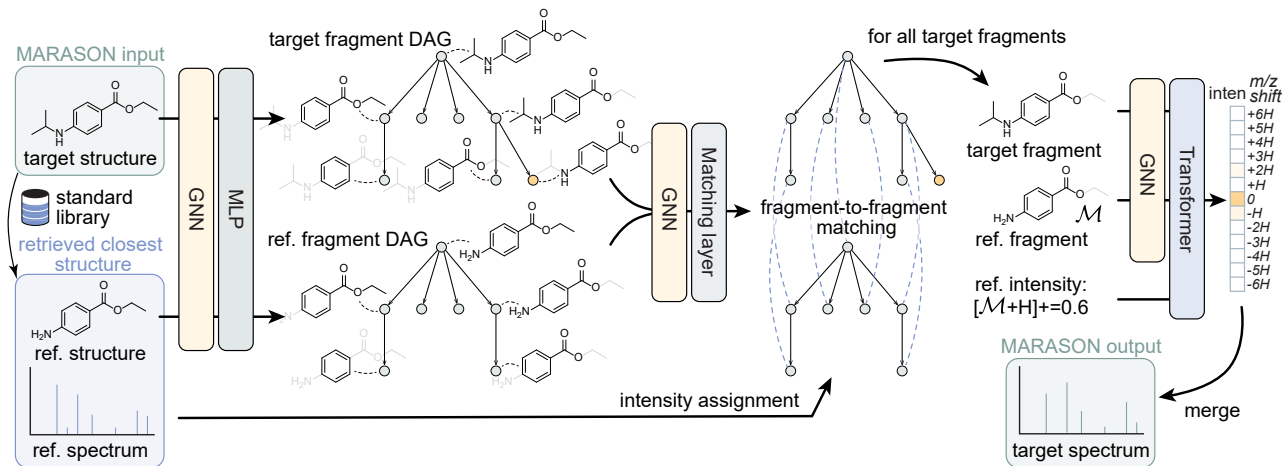
Figure 2. Overview of MARASON: a retrieval-augmented mass spectrum simulator with neural graph matching. We retrieve reference structures and spectra from a reference library (more specifically, the training dataset) based on Tanimoto similarity (Bajusz et al., 2015) to the target structure. Both target and reference structures are fragmented, resulting in two similar fragmentation DAGs, as similar structures are expected to have similar fragmentation patterns in chemistry (Shahneh et al., 2024). The neural graph matching module further finds the node-level matching between fragmentation DAGs, whereby each node represents a fragment. The aligned target and reference fragment, together with reference intensity, are further concatenated to predict the final spectrum.We use three identical GNN modules with separate weights to learn embeddings: one for the target fragments, one for the reference fragments, and one shared by both target and reference fragments to capture matching information. After computing the embeddings with these GNN modules, we use the shared GNN's outputs as input to a matching module, which produces a matching matrix. This matrix is then used to align reference fragments and spectral peaks with the target fragments. The resulting matched fragment pairs, along with their similarity scores, are fed into a transformer module to predict the final relative intensities of the spectral peaks corresponding to each target fragment.

libraries with simulated spectra for novel annotation, from a current size of ~27K unique compounds in NIST (2020) to all 111M compounds in PubChem and beyond.

2. Related Work

Retrieval Augmented Generation in AI for Science. Retrieval augmented generation (RAG) is a technique that integrates relevant information retrieved from external databases into a model's training and inference processes. While originally developed to aid Large Language Model (LLM)-based AI agents, their applicability has readily extended to scientific tasks, spanning LLMs for material discovery (Buehler, 2024), drug design (Pal et al., 2023), and organic synthesis (M. Bran et al., 2024). Utilizing RAG in various forms has also shown ground-breaking advances in scientific discovery beyond LLM agents; one of the most prominent examples is AlphaFold model (Jumper et al., 2021), whose sequence alignment module is effectively a RAG module. RAG is also found helpful in molecular machine learning tasks such as molecular generation (Lee et al., 2024), drug design (Zhang et al., 2024), and protein function prediction (Shaw et al., 2024). Despite success in certain molecular learning tasks, there remains the opportunity for a principled RAG design strategy tailored to small molecules that can perform robust and informative structural matching.

Neural Graph Matching. Graph matching is a combina-

torial optimization problem that matches the nodes of multiple graphs by maximizing the edge-wise and node-wise affinities. Neural graph matching was developed to tackle the computational challenges of the NP-hard quadratic assignment problem (Lawler, 1963) and the vulnerability of predefined affinity metrics (Rolínek et al., 2020; Nowak et al., 2018; Guo et al., 2023). Among all design choices, a family of linear-matching methodologies (Wang et al., 2020; Yu et al., 2020; Sarlin et al., 2020) incorporates graph neural networks (Scarselli et al., 2008) to embed edge affinities into node embeddings that are then used in a differentiable node-matching layer using, e.g., Sinkhorn & Rangarajan (1964) or a simpler Softmax for assignment. This paper explores and validates the applicability of neural graph matching in the context of retrieval-augmented mass spectra generation in molecular machine learning.

3. Methods

MARASON is a mass spectrum simulator built on the ICE-BERG model (Goldman et al., 2024) with reference retrieval augmentation and neural graph matching. In this section, unless otherwise specified, we use capitalized bold letters for matrices, lowercase bold letters for vectors, and non-bold letters for scalars. $\mathcal{M}, \mathcal{F}, \mathcal{G}$ indicate a molecular graph, fragment graph, and fragmentation DAG (from ICEBERG-Generate), respectively. All reference-related variables are

annotated with a superscript "r".

3.1. Preliminary: ICEBERG MS/MS Simulator

Mass spectrometry (MS) is a powerful analytical chemistry method for the discovery of unknown compounds and natural products (Wang et al., 2016). Compounds analyzed with tandem MS undergo an ionization and fragmentation process, after which charged fragments are detected at their m/z values with an intensity proportional to their relative abundance and cross section. MARASON is built on the ICEBERG model which reported state-of-the-art accuracy. As described by Goldman et al. (2024), ICEBERG is a two-stage model:

ICEBERG-Generate learns to mimic collision-induced dissociation by predicting the most likely fragments through recursive removal of atoms and bonds from the molecular graph. It results in an autoregressive model to handle multiple breakages, whose output is a directed acyclic graph (DAG) where the root node is the input molecule and fragments are the children. The hierarchical information of fragments from ICEBERG-Generate is used in MARASON for graph matching.

ICEBERG-Score takes the fragmentation DAG and outputs the intensities of peaks corresponding to each fragment. As shown in Fig. 2, the output layer also accounts for up to ± 6 hydrogen shifts in mass to support chemical rearrangements. ICEBERG-Score is composed of 1) a GNN module that takes in the molecular graphs and context parameters and outputs the corresponding hidden representation of fragments and root molecules; and 2) a set transformer (Lee et al., 2019) that predicts intensity values for each fragment. Predicted intensity values are merged into a mass spectrum, where the m/z values are straightforwardly calculated from the structures of fragments.

MARASON aims to improve ICEBERG by integrating retrieval augmentation with neural graph matching into ICEBERG-Score.

3.2. Retrieval Augmentation Processing

With the chemical intuition that structurally similar molecules are expected to have similar spectra (Shahneh et al., 2024), we retrieve molecules with the highest structural similarity to our target molecule. This subsection presents our implementation of MS/MS retrieval and processing of retrieved data for best deep-learning efficacy.

3.2.1. RETRIEVAL OF STRUCTURES AND SPECTRA

Let our retrieval database be any database with annotated structure-spectrum pairs; in our experiments, we use the training dataset to mitigate concerns about data leakage. We retrieve the most similar reference molecule as quantified by Tanimoto similarity (Bajusz et al., 2015) operating on Morgan fingerprints (Morgan, 1965). We exclude the entries that have nonmatching adduct types or instrument types recorded to mitigate the confounding effects of different experimental conditions.

We then identify up to three reference spectra with the most similar collision energy to the target (query) structure $\mathcal{M}$. Collision energies influence the extent of fragmentation and shift the spectrum patterns towards lower m/z ranges with higher energies; there are often multiple spectra with different collision energies for a unique molecule in NIST (2020). The result of retrieval is therefore a single reference structure $\mathcal{M}^r$, $\mathrm{Tanimoto}(\mathcal{M}, \mathcal{M}^r)$, and up to three reference spectra and their collision energy values, which are fed into MARASON. As the collision energy value is continuous, our aim of including three reference energies is to learn an appropriate spectrum embedding for the target collision energy value by interpolating from three closest energies.

3.2.2. PEAK-FRAGMENT ASSIGNMENT AND LEARNING

Reference spectra only provide m/z values and peak intensities. This information is enriched by annotating peaks with ICEBERG-predicted fragments of the reference structure. For each fragment $\mathcal{F}_j^r$, we assign all peaks that fall within the 20 ppm (parts-per-million) range of at most 13 mass values: $\{\mathcal{F}_j^r - 6\delta, \mathcal{F}_j^r - 5\delta, ..., \mathcal{F}_j^r, ..., \mathcal{F}_j^r + 6\delta\}$, where δ is the mass of a hydrogen atom. Such a process leads to a 13-dimensional vector for each $\mathcal{F}_j^r$ indicating its intensities in the reference spectrum. The assigned 13-dimension intensity is concatenated with values of the reference and target collision energies. All reference intensities at the same collision energy are processed by a set transformer, followed by an average pooling layer that merges intensity embeddings per fragment from three collision energies. The obtained reference intensity embeddings, denoted as $\mathbf{T}^r$, are defined such that the pooled embedding for fragment j is represented as $\mathbf{t}_j^r$.

3.3. Fragmentation DAG Graph Matching

We define a fragmentation DAG graph matching problem to recapitulate domain experts' practice of comparing similar fragments from the reference and the target, alongside their hierarchy with respect to the original structures. In fragmentation DAGs from the ICEBERG-Generate model, fragments are viewed as nodes, and fragmentation paths are viewed as edges. Since each fragment is a molecular graph, a fragmentation DAG is a graph of graphs, i.e., a meta-graph. Following training steps in Goldman et al. (2024), a pretrained ICEBERG-Generate model predicts two fragmentation DAGs: one for the reference molecular graph and one for the target. We then approach the graph matching task using fixed affinity metrics with traditional

graph matching solvers as well as learnable affinity metrics with neural graph matching.

3.3.1. TRADITIONAL SOLVERS WITH FIXED METRICS

Defining the affinity metric is the first step of matching fragments between two fragmentation DAGs. Since each fragment is a molecular graph, we resort to Tanimoto similarity (Bajusz et al., 2015) and construct the following pairwise affinity matrix $\mathbf{M}$:

$$m_{i,j} \leftarrow \text{Tanimoto}(\mathcal{F}_i, \mathcal{F}_j^r), \tag{1}$$

where $\mathcal{F}_i$ is fragment i of the target structure and $\mathcal{F}_j^r$ is fragment j of the reference structure. With the Tanimoto-based fragment-level affinity scores, we can formulate a linear assignment problem:

$$\max_{\mathbf{X}} \ \text{tr}(\mathbf{M}^\top \mathbf{X}),$$
$$s.t. \quad \mathbf{X} \in \{0,1\}^{n \times n^r}, \mathbf{X}\mathbf{1}_{n^r} \leq \mathbf{1}_n, \mathbf{X}^\top \mathbf{1}_n \leq \mathbf{1}_{n^r}, \tag{2}$$

where $\mathbf{X}$ is the matching matrix, n and n^r are the number of fragments from the target DAG and the reference DAG, respectively, $\mathbf{1}_n$ means a column vector of n 1s. $x_{i,j} = 1$ means $\mathcal{F}_i$ is matched to $\mathcal{F}_j^r$ and $x_{i,j} = 0$ otherwise. Eq. (2) can be solved by the Hungarian algorithm (Kuhn, 1955), whereby it incorporates fragment-level graph affinities but ignores the hierarchical information in DAGs.

Graph matching further extends Eq. (2), by explicitly modeling edge affinities in the DAG, resulting in a quadratic assignment problem (Lawler, 1963):

$$\max_{\mathbf{X}} \ \text{vec}(\mathbf{X})^\top \ \mathbf{K} \ \text{vec}(\mathbf{X}), \tag{3}$$

where the constraints are the same as Eq. (2), $\mathbf{K} \in \mathbb{R}^{nn^r \times nn^r}$ is the affinity matrix, and $\text{vec}(\cdot)$ means column-wise vectorization. The diagonal elements of $\mathbf{K}$ are taken to be $\mathbf{M}$ as the fragment-level affinities, while the off-diagonal elements are constructed by the inner product of edges of DAGs. Solving Eq. (3) is NP-hard, but there are approximate solvers available (Cho et al., 2010; Leordeanu et al., 2009) through a Python interface (Wang et al., 2024).

After solving Eq. (2) or Eq. (3), we establish the matching between fragments and retrieved intensities using the matching result $\mathbf{X}$. As shown in Fig. 1 and later ablation studies (Table 3), these methodologies yield inferior results to neural graph matching, indicating the importance of flexible, learnable affinity metrics.

3.3.2. NEURAL MATCHING WITH LEARNED METRICS

The biggest drawback of traditional graph matching methods is that their fixed affinity metrics do not incorporate the flexibility required to handle noisy real-world data (Wang et al., 2020; Cho et al., 2013). To overcome this challenge

in the context of RAG for MS/MS simulation, we propose a neural graph matching method that leverages message-passing networks on the fragmentation DAG. Specifically, we develop a nested GNN (Zhang & Li, 2021) that learns fragment-level embedding and DAG hierarchical embedding by two GNNs, followed by a differentiable matching layer.

Fragment-level embedding learning. For each fragment $\mathcal{F}_i$ from the target molecule $\mathcal{M}$, since both $\mathcal{F}_i$ and $\mathcal{M}$ are molecular graphs, we learn graph-level embeddings with a shared GNN_{frag}. We build embeddings at the fragment level for $\mathcal{F}_i$ using an MLP to project the concatenation of $\text{GNN}_{\text{frag}}(\mathcal{M})$, $\text{GNN}_{\text{frag}}(\mathcal{F}_i)$, and $(\text{GNN}_{\text{frag}}(\mathcal{M}) - \text{GNN}_{\text{frag}}(\mathcal{F}_i))$, together with an encoded number of broken bonds (from ICEBERG-Generate), encoded chemical formula of $\mathcal{F}_i$ and the chemical formula difference from $\mathcal{M}$ to $\mathcal{F}_i$. These differences in formula and GNN_{frag} embeddings represent the chemical concept of "neutral losses". The same neural networks are applied to the target fragments and the reference fragments, yielding $\mathbf{H}$ and $\mathbf{H}^r$ respectively as the fragment-level embeddings learned from GNNs.

DAG hierarchical embedding learning. For each fragmentation DAG, we construct two graphs $\mathcal{G}$ and $\mathcal{G}^{-1}$, where $\mathcal{G}$ is the top-down fragmentation DAG in which $e_{ij} \in \mathcal{G}$ if and only if fragment i is a parent of fragment j. $\mathcal{G}^{-1}$ is the bottom-up DAG where all edges are reversed. With $\mathbf{H}$ (or $\mathbf{H}^r$) extracted by aforementioned fragment-level GNNs, it is updated as

$$\bar{\mathbf{H}} \leftarrow \mathbf{H} + \text{GNN}_{\text{fwd}}(\mathbf{H}, \mathcal{G}) + \text{GNN}_{\text{rev}}(\mathbf{H}, \mathcal{G}^{-1}), \tag{4}$$

so that the DAG structures are embedded into fragment-level embeddings. We apply the same GNNs for both target DAG and reference DAG.

Similarity and differentiable matching. We then calculate the similarity matrix $\bar{\mathbf{M}}$ of a given reference target pair as

$$\bar{m}_{i,j} \leftarrow \text{cosine}(\bar{\mathbf{h}}_i, \bar{\mathbf{h}}_j^r), \tag{5}$$

where $\bar{\mathbf{h}}_i$ is the embedding of fragment i of the target, and $\bar{\mathbf{h}}_j^r$ is the embedding of fragment j of the reference. We can then calculate the matching matrix $\bar{\mathbf{X}}$ as follows

$$\bar{\mathbf{X}} \leftarrow \text{matching}(\bar{\mathbf{M}}). \tag{6}$$

In neural graph matching research, $\text{matching}(\cdot)$ could either be the Sinkhorn (Sinkhorn & Rangarajan, 1964) or Softmax algorithms; Sinkhorn is preferred in smaller-sized graphs to enforce stronger matching constraints (Wang et al., 2020), whereas Softmax is used in larger-sized graphs for its efficiency (Sarlin et al., 2020). We implement Softmax-based matching in this paper after a careful ablation study, as empirical experiments with $\sim$100 fragments generated

by ICEBERG-Generate suggest this number does not pose a serious concern of constraint violation. It is worth noting that $\bar{\mathbf{X}}$ is a continuous matrix so that the neural graph matching pipeline is differentiable, and that $\bar{\mathbf{X}}$ is further integrated into the intensity prediction module so that it will receive gradients during training and learn the affinity metric end-to-end.

3.4. Intensity Prediction

Given the target fragment embeddings $\mathbf{H}$ and the reference fragment embeddings $\mathbf{H}^r$, together with peak intensities $\mathbf{T}^r$ assigned to reference fragments from Sec. 3.2.2, as well as the matching matrix $\bar{\mathbf{X}}$ bridging target and reference fragments from Sec. 3.3.2, we can then create triplets of (target fragments, reference fragments, reference intensities) to serve as input to the intensity prediction layers. To reiterate our motivation for this retrieval-augmented generation framework: similar structures tend to have similar fragmentation patterns in chemistry, and similar fragments tend to have similar response factors that relate their abundance to the observed intensity (Shahneh et al., 2024). Therefore, the intensity of reference fragments should offer hints for the intensity of the target fragments.

When predicting the intensities, we also consider the matching score of target fragment i as s_i: $s_i \leftarrow \sum_{j=1}^{n^r} \bar{x}_{i,j} \bar{m}_{i,j}$, which is further aggregated with $\mathbf{H}$, $\mathbf{H}^r$, $\mathbf{T}^r$, and the Tanimoto similarity of the original molecules for intensity prediction. The input is the concatenation of

$$\left[\mathbf{H}, \bar{\mathbf{X}}\mathbf{H}^r, \bar{\mathbf{X}}\mathbf{T}^r, \mathbf{s}, \text{Tanimoto}(\mathcal{M}, \mathcal{M}^r) \right], \qquad (7)$$

which is then processed by a set transformer (Lee et al., 2019) where embeddings associated with each fragment are treated as one element in the set. Finally, the intensities are computed through an attention layer and an MLP.

4. Experiments

We implement MARASON with PyTorch (Paszke et al., 2017), based on the official implementation of ICE-BERG (Goldman et al., 2024) and the graph matching toolkit Pygmtools (Wang et al., 2024). All experiments are conducted on a workstation with AMD 3995WX CPU, $4\times$NVIDIA A5000 GPU, and 512GB RAM. Our experiments are designed to answer the following questions: (1) Does MARASON, a representative implementation of neural graph matching-based RAG in molecular machine learning, improve the performance of mass spectrum prediction compared to retrieval-free ICEBERG and other state-of-the-art models? (2) How accurate is MARASON on an authentic assessment of identifying unknown compounds to support its application to real-world chemical and biological campaigns? (3) Does MARASON generate a reasonable matching pattern that aligns with the fragmentation pro-
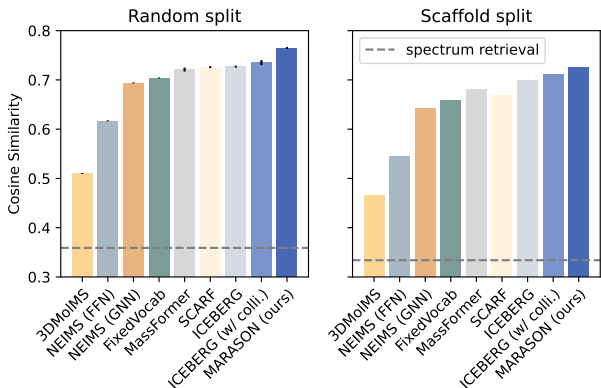

Figure 3. Spectral similarity between experimental spectra and predictions from various MS/MS simulators on NIST (2020) dataset. Results of all baselines are from the benchmark developed by Goldman et al. (2024). Incorporating collision energies with ICE-BERG slightly improves the accuracy, while MARASON further improves the accuracy by a substantial margin. Error bars are reported for the random split with 3 random seeds. The scaffold split is more challenging and results in lower cosine similarity for all methods. It is worth noting, from the dashed lines in the plots, that the retrieved spectra are informative, but the cosine similarities of the retrieved spectra are lower than the predicted spectra produced by all baseline models, demonstrating that RAG on MS/MS is a non-trivial task.

cess? We conduct experiments and answer these research questions quantitatively and qualitatively.

4.1. Experimental Setup

4.1.1. DATASET AND BASELINES

We trained our models on the NIST (2020) dataset with 530,640 high-energy collision-induced dissociation (HCD) spectra and 25,541 unique molecular structures. The number of spectra is larger than unique structures because spectra are collected at different collision energies with different adducts. The dataset is split into structurally disjoint 80%-10%-10% train-validate-test subsets. Following Goldman et al. (2024), we evaluate on two different splits: (1) a random split that splits different InChI keys and (2) a Murcko scaffold split that clusters different molecular scaffolds that require more generalization to out-of-distribution structures. It is worth noting our dataset is preprocessed differently from Goldman et al. (2024) as we include collision energies as input that are crucial to retrieving the most informative spectra, whereas the original dataset pre-average spectra when multiple collision energies are used. We also include negative adduct types in training to exploit the full NIST (2020). Despite this difference, the testing split is kept consistent with the ICEBERG paper to maintain comparative evaluation; we also include an intermediate ablation (retrieval-free ICEBERG with collision energy) to fairly position the improvement of RAG. Since collision energies

Table 1. Retrieval accuracy (mean ± 95% confidence interval on 3 random seeds) on NIST (2020) dataset with random (InChI key) split on positive adduct types. MARASON surpasses all baselines in terms of retrieval accuracy, which is an authentic assessment of MS/MS simulators' performance in real-world applications where they are used to distinguish the true structure from a list of candidates. Results of all baselines are from the benchmark developed by Goldman et al. (2024). We also outperform the recently developed method FraGNNet (Young et al., 2024b), which is reported separately in Table 6 in the Appendix as it only allows prediction for [M+H]+ adducts.

Accuracy @ Top-k	1	2	3	4	5	8	10
3DMolMS (Hong et al., 2023)	0.055±0.003	0.105±0.000	0.146±0.005	0.185±0.007	0.225±0.009	0.332±0.005	0.394±0.008
FixedVocab (Murphy et al., 2023)	0.172±0.004	0.304±0.004	0.399±0.002	0.466±0.007	0.522±0.012	0.638±0.009	0.688±0.006
NEIMS (FFN) (Wei et al., 2019)	0.105±0.003	0.243±0.012	0.324±0.013	0.387±0.011	0.440±0.014	0.549±0.010	0.607±0.005
NEIMS (GNN) (Zhu et al., 2020)	0.175±0.005	0.305±0.003	0.398±0.002	0.462±0.004	0.515±0.005	0.632±0.007	0.687±0.005
MassFormer (Young et al., 2024a)	0.191±0.008	0.328±0.006	0.422±0.004	0.491±0.002	0.550±0.005	0.662±0.005	0.716±0.003
SCARF (Goldman et al., 2023)	0.187±0.008	0.321±0.006	0.417±0.007	0.486±0.008	0.541±0.009	0.652±0.008	0.708±0.009
ICEBERG (Goldman et al., 2024)	0.189±0.012	0.375±0.005	0.489±0.007	0.567±0.005	0.623±0.004	0.725±0.003	0.770±0.002
ICEBERG (w/ collion energy)	0.202±0.009	0.399±0.008	0.513±0.008	0.585±0.008	0.639±0.010	0.749±0.006	0.793±0.007
MARASON (ours)	**0.278±0.002**	**0.455±0.004**	**0.562±0.009**	**0.636±0.006**	**0.685±0.004**	**0.784±0.002**	**0.827±0.004**

are found crucial to RAG and the NPLIB1 dataset used in Goldman et al. (2024) does not have collision energy labels for most of the spectra, this dataset is not included for comparison.

We include all peer methods reported in the benchmark developed by Goldman et al. (2024). Traditionally, MS/MS simulators have relied on combinatorial enumeration of bond breakages, which mimics the physical process, but they are computationally demanding and often inaccurate (Allen et al., 2015; Ridder et al., 2014). Deep learning-based simulators significantly reduce the inference time with competitive accuracy, while the physical constraint is either fully lifted (Young et al., 2024a; Wei et al., 2019) or relaxed to subformulae (Murphy et al., 2023; Goldman et al., 2023). ICEBERG combines both ideas by learning the fragmentation DAG. Besides them, we also consider a "spectrum retrieval" baseline by simply taking the retrieved spectrum as the prediction.

4.1.2. QUANTITATIVE EVALUATION METRICS

Spectrum similarity. We evaluate model performance via how similar the predicted mass spectra are to experimental spectra. Specifically, since we consider molecules under 1,500 Da, we create a 15,000-dimensional vector valued between 0 to 1 that encodes the spectrum, where each element means the peak intensity that falls within the mass bin of 0.1 Da, to balance mass resolution and the vector dimension sparsity. All method outputs are transformed into this 15,000-dim vector so that it accommodates both methods that predict binned spectra (Wei et al., 2019; Young et al., 2024a) and methods that predict intensities with known exact mass (Murphy et al., 2023; Goldman et al., 2023; 2024), including ours. In this paper, we use cosine similarity to compare spectra, which is the de facto standard metric used in real-world case studies (Li et al., 2021).

Retrieval accuracy. In structural elucidation campaigns, MS/MS simulators can predict pseudo-spectrum labels for
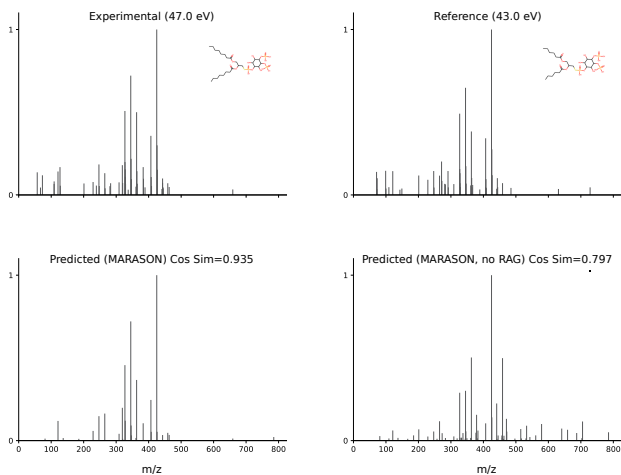

Figure 4. Qualitative comparison of experimental spectrum (i.e., ground-truth), reference spectrum, spectrum predicted by MARASON, and spectrum predicted by MARASON (no RAG). Cosine similarities between the experimental spectrum and predicted spectra are annotated. The retrieved reference structure, as shown in the plot, is structurally similar to the target structure, whereby the reference spectrum offers important intensity information. Our graph-matching-based RAG strategy associates reference peaks with the target, despite the peaks having different m/z values between the reference and the target.

candidate structures and compare them to the experimental spectrum. Candidate structures can then be ranked based on spectrum similarity to identify the most likely structure. This retrieval setting evaluates how this model might be deployed in a real-world structural elucidation campaign. Following Goldman et al. (2024), for each molecule from the testing dataset, we obtain up to 49 isomeric decoys with the highest Tanimoto similarities to the true target structure, i.e., the decoys most likely to be mistaken under Tanimoto measurement. We run MARASON and all comparative methods on 50 structures per test spectrum (49 decoys + 1 true structure) and rank all of them by cosine similarity. We evaluate the retrieval accuracy at top-k.

Table 2. Retrieval accuracy (mean with 99.9% confidence intervals upon bootstrapping, 20,000 resamples) with known chemical formula on the MassSpecGym dataset (Bushuiev et al., 2024). We consider MassSpecGym as a more challenging setting than NIST (2020) as it incorporates fewer annotated spectra and emphasizes generalization to different molecular scaffolds. MARASON surpasses all baselines in terms of retrieval accuracy. Baseline performances are quoted from Bushuiev et al. (2024).

Accuracy @ Top-k	1	5	20
NEIMS (FFN) (Wei et al., 2019)	0.0762 (0.0677-0.0854)	0.2270 (0.2132-0.2412)	0.4412 (0.4251-0.4575)
NEIMS (GNN) (Zhu et al., 2020)	0.0363 (0.0305-0.0429)	0.1355 (0.1246-0.1468)	0.3377 (0.3226-0.3537)
FraGNNet (Young et al., 2024b)	0.3193 (0.3040-0.3350)	0.6320 (0.6164-0.6476)	0.8270 (0.8145-0.8393)
MARASON (ours)	**0.3403 (0.3286-0.3520)**	**0.6404 (0.6277-0.6519)**	**0.8539 (0.8448-0.8624)**

Table 3. Ablation Study of MARASON design choices. RAG strategies include no RAG, concatenating one reference spectrum to the model input, traditional matching methods discussed in Sec. 3.3.1 including Hungarian (Kuhn, 1955) and RRWM (Cho et al., 2010), and neural graph matching (NGM) presented in Sec. 3.3.2 where the matching layer could be either Sinkhorn (Sinkhorn & Rangarajan, 1964) or Softmax. We compare cosine similarity on random split with seed = 1.

Base model	RAG strategy	Match layer	Cosine sim.
MARASON (shared GNN)	No RAG	-	0.739
	Concat	-	0.737 (-0.3%)
	Hungarian	-	0.746 (+0.9%)
	RRWM	-	0.742 (+0.4%)
	NGM	Sinkhorn	0.749 (+1.4%)
	NGM	Softmax	0.753 (+1.9%)
MARASON (not shared GNN)	NGM	Sinkhorn	0.753 (+1.9%)
	NGM	Softmax	**0.757 (+2.4%)**

4.2. Results and Discussions

4.2.1. SPECTRAL SIMILARITY AND VISUALIZATION

We evaluate the prediction power of MARASON by comparing the spectral cosine similarity of the predicted spectrum of different baseline models on the NIST (2020) dataset. The results are summarized in Fig. 3. MARASON outperforms all baselines in both types of splits, and the success on scaffold split excludes the possibility that MARASON needs highly similar reference structures for accurate predictions. It improves the cosine similarity of the ICEBERG baseline by a relative 5.2% on random split and a relative 3.7% on scaffold split. Since the original ICEBERG model does not consider collision energy, we also compare a collision energy-aware version of ICEBERG. This study demonstrates that neural graph matching-based RAG improves the performance of MS/MS simulation, a representative task in molecular machine learning. An example of predicted spectra of MARASON is provided in Fig. 4.

4.2.2. RETRIEVAL FROM PUBCHEM CANDIDATES

The evaluation of the real-world applicability of MS/MS simulators is summarized in Table 1. In the retrieval benchmark, MARASON improves the top-1 retrieval accuracy

upon ICEBERG from 18.7% to 27.8%, a marked increase in the state-of-the-art performance for this task by a relative margin of 48% over the next best method. As an ablation study, ICEBERG (with collision) has a top-1 accuracy of 20.2%, indicating that the majority of improvement is directly attributable to our use of RAG. MARASON consistently outperforms all baselines from top-1 to top-10 retrieval accuracies. Table 1 only covers the random split; retrieval experiment results on scaffold split are in Table 5 in the Appendix.

4.2.3. RETRIEVAL FROM THE MASSSPECGYM DATASET

We further retrain MARASON on the recently developed open-source dataset, MassSpecGym (Bushuiev et al., 2024), where we achieve state-of-the-art retrieval accuracy, as shown in Table 2. Since not all spectra in MassSpecGym have collision energy label, but MARASON requires labeled collision energies, we use a NIST-pretrained model to create pseudo labels for those unannotated spectra, i.e., transferring the collision energy knowledge from NIST to MassSpecGym. The model is retrained from scratch on MassSpecGym data to ensure no data leakage from NIST. MARASON improves the top-1 retrieval accuracy upon the current state-of-the-art, FraGNNet, from 31.93% to 34.03%, a marked increase by a relative margin of 6% over the next best method. Random seed is fixed as 1 for MARASON.

4.2.4. ABLATION STUDY

We conduct an ablation study to compare matching algorithms and GNN designs on the NIST (2020) dataset under a random split, as shown in Table 3. MARASON includes modest engineering changes to ICEBERG for more efficient training and inference on GPUs, which accounts for the slight improvement over ICEBERG (with collision). The naive RAG strategy that concatenates the reference spectrum as an extra 15,000-dimensional input leads to a *negative* impact on the cosine similarity. Traditional graph matching methods discussed in Sec. 3.3.1, by comparison, yield a minor benefit of fragment-level matching in RAG. With the comparatively more expressive and end-to-end learnable neural graph matching described in Sec. 3.3.2, the cosine similarity is further improved. A possible explanation for
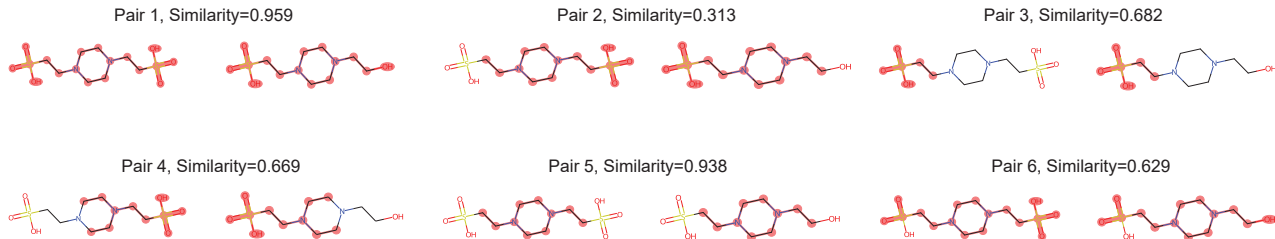
Figure 5. Visualization of matched fragment pairs and their Softmax-similarity scores discovered by the neural graph matching module in MARASON. Structures on the left are from the target fragmentation DAG and structures on the right are from the reference. Fragments are highlighted from the original structure. Our matching module learns to match not only exact structures (pair 2, 3, 4, 5), but also correlated structures with slight modifications (pair 1, 6). Only 6 pairs from two fragmentation DAGs are shown due to space limits, the full list of matched pairs can be found in Fig. 6 in the Appendix together with a visualization of traditional graph matching in Fig. 7.

Table 4. The performance of MARASON with and without RAG on target and reference pairs grouped by Tanimoto similarity. We report cosine similarity on the random split with random seed = 1. There is no testing sample with a Tanimoto similarity between 0 and 0.1.

Tanimoto Similarity	(0, 0.1]	(0.1, 0.2]	(0.2, 0.3]	(0.3, 0.4]	(0.4, 0.5]	(0.5, 0.6]	(0.6, 0.7]	(0.7, 0.8]	(0.8, 0.9]	(0.9, 1.0]
MARASON	N/A	0.550	0.614	0.690	0.741	0.789	0.815	0.808	0.805	0.824
MARASON (No RAG)	N/A	0.566	0.611	0.682	0.727	0.768	0.791	0.780	0.759	0.784

the superiority of Softmax over Sinkhorn is that Softmax is sufficient for the many-to-one aggregation path in Eq. (7) and provides better gradients because it takes fewer iterations. It is also shown in Sarlin et al. (2020) that Softmax outperforms as the matching layer for larger-sized graphs. Finally, we discover that separating the GNN that learns embeddings for matching and the GNN that learns embeddings for intensity prediction results in a higher spectrum similarity, compared to using GNNs with shared weights for both purposes. The reason could be that MARASON needs separate modules for learning intensity-related information (e.g., molecular fragment cross sections) and information about the fragmentation DAGs themselves.

4.2.5. QUALITATIVE STUDY OF MATCHING PATTERNS

We visualize matching pairs of fragments by assigning each target fragment to the reference fragment with the highest Softmax score. An example of six matched fragment pairs is shown in Fig. 5. Fragment pairs also show similar fragmentation patterns (missing the same functional group, C-N bond breakages, etc.). This example illustrates how MARASON has learned to match fragments that are generated through similar mechanisms in the fragmentation process in order to model the relationship between each matched peak and fragment pair in the reference and target spectrum.

4.2.6. WHAT IF THERE IS NO REFERENCE STRUCTURE?

To understand the relationship between RAG's improvement and the availability of a similar-enough reference structure, for all testing data in NIST (2020), we group structural pairs

based on their Tanimoto similarities and compare performances of MARASON with or without RAG (same model configurations as in Table 3). As shown in Table 4, RAG starts to bring a significant performance gain when the Tanimoto similarity is larger than 0.3. When the best reference structure has a Tanimoto similarity between 0.1 and 0.3, RAG does not bring a significant improvement, while MARASON still performs robustly with such irrelevant references. The slight performance drop between Tanimoto similarity (0.1, 0.2] also suggests a simple trick to further improve MARASON's accuracy: use standard MARASON when the retrieved structure has a Tanimoto similarity > 0.2 and use the non-RAG version otherwise.

5. Conclusion

This paper presents a retrieval-augmented generation framework for molecules by integrating neural graph matching into an existing end-to-end geometric deep learning framework. On the molecular machine learning task of mass spectrum simulation, we implement MARASON to match the fragmentation DAGs of the target and reference structures, pair up peaks and fragments in the reference and target structures, and use that alignment information to generate improved spectrum predictions. MARASON establishes new state-of-the-art performance in terms of the quality of simulated spectra and retrieval accuracy in downstream applications. Future improvements and adaptations could see this neural graph matching strategy applied to a broader range of retrieval-augmented structure-property prediction tasks across the field of molecular machine learning.

Acknowledgements

Impact Statement

This paper presents work that aims to advance both machine learning and mass spectroscopy. As a methodology-focused paper, this paper itself does not pose any significant societal impact that is required to be highlighted here. The authors are also aware that mass spectroscopy could be used in human-centric research campaigns such as metabolomics, and it is important to ensure that the reference database is not biased towards ethnicity, sex, and age, or at least being acknowledged about the potential bias. On the other hand, efforts to develop better mass spectroscopy models, including this paper, will potentially lower the cost and expedite those discovery campaigns. If such models are used properly for the underrepresented cohort, it will finally mitigate the bias and unfairness in human-centric studies.

References

Allen, F., Greiner, R., and Wishart, D. Competitive fragmentation modeling of esi-ms/ms spectra for putative metabolite identification. *Metabolomics*, 11:98–110, 2015.

Bajusz, D., Rácz, A., and Héberger, K. Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations? *Journal of cheminformatics*, 7:1–13, 2015.

Bittremieux, W., Wang, M., and Dorrestein, P. C. The critical role that spectral libraries play in capturing the metabolomics community knowledge. *Metabolomics*, 18 (12):94, 2022.

Buehler, M. J. Generative retrieval-augmented ontologic graph and multiagent strategies for interpretive large language model-based materials design. *ACS Engineering Au*, 4(2):241–277, 2024.

Bushuiev, R., Bushuiev, A., de Jonge, N. F., Young, A., Kretschmer, F., Samusevich, R., Heirman, J., Wang, F., Zhang, L., Dührkop, K., et al. Massspecgym: A benchmark for the discovery and identification of molecules. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

Cho, M., Lee, J., and Lee, K. M. Reweighted random walks for graph matching. In *European Conference on Computer Vision*, pp. 492–505, 2010.

Cho, M., Alahari, K., and Ponce, J. Learning graphs to match. In *International Conference on Computer Vision*, pp. 25–32, 2013.

Dang, L., White, D. W., Gross, S., Bennett, B. D., Bittinger, M. A., Driggers, E. M., Fantin, V. R., Jang, H. G., Jin, S., Keenan, M. C., et al. Cancer-associated idh1 mutations produce 2-hydroxyglutarate. *Nature*, 462(7274):739–744, 2009.

Goldman, S., Bradshaw, J., Xin, J., and Coley, C. Prefix-tree decoding for predicting mass spectra from molecules. *Advances in Neural Information Processing Systems*, 36: 48548–48572, 2023.

Goldman, S., Li, J., and Coley, C. W. Generating molecular fragmentation graphs with autoregressive neural networks. *Analytical Chemistry*, 96(8):3419–3428, 2024.

Griffen, E., Leach, A. G., Robb, G. R., and Warner, D. J. Matched molecular pairs as a medicinal chemistry tool: miniperspective. *Journal of medicinal chemistry*, 54(22): 7739–7750, 2011.

Guo, Z., Wang, R., Jiang, S., Yang, X., and Yan, J. Deep learning of partial graph matching via differentiable top-k. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6272–6281, 2023.

Hong, Y., Li, S., Welch, C. J., Tichy, S., Ye, Y., and Tang, H. 3dmolms: prediction of tandem mass spectra from 3d molecular conformations. *Bioinformatics*, 39(6):btad354, 2023.

Huang, Z., Yang, L., Zhou, X., Qin, C., Yu, Y., Zheng, X., Zhou, Z., Zhang, W., Wang, Y., and Yang, W. Interaction-based retrieval-augmented diffusion models for protein-specific 3d molecule generation. In *International Conferencee on Machine Learning*, 2024.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Kuhn, H. W. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

Lawler, E. L. The quadratic assignment problem. *Management Science*, 9(4):586–599, 1963.

Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., and Teh, Y. W. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International conference on machine learning*, pp. 3744–3753. PMLR, 2019.

Lee, S., Kreis, K., Veccham, S. P., Liu, M., Reidenbach, D., Paliwal, S., Vahdat, A., and Nie, W. Molecule generation

with fragment retrieval augmentation. *arXiv preprint arXiv:2411.12078*, 2024.

Leordeanu, M., Hebert, M., and Sukthankar, R. An integer projected fixed point method for graph matching and map inference. In *Neural Information Processing Systems*, pp. 1114–1122, 2009.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

Li, Y., Gu, C., Dullien, T., Vinyals, O., and Kohli, P. Graph matching networks for learning the similarity of graph structured objects. In *International Conference on Machine Learning*, pp. 3835–3845, 2019.

Li, Y., Kind, T., Folz, J., Vaniya, A., Mehta, S. S., and Fiehn, O. Spectral entropy outperforms ms/ms dot product similarity for small-molecule compound identification. *Nature Methods*, 18(12):1524–1531, 2021.

M. Bran, A., Cox, S., Schilter, O., Baldassari, C., White, A. D., and Schwaller, P. Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 6(5):525–535, May 2024. ISSN 2522-5839. doi: 10.1038/s42256-024-00832-8. URL https://doi.org/10.1038/s42256-024-00832-8.

Morgan, H. L. The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service. *Journal of chemical documentation*, 5(2):107–113, 1965.

Murphy, M., Jegelka, S., Fraenkel, E., Kind, T., Healey, D., and Butler, T. Efficiently predicting high resolution mass spectra with graph neural networks. In *International Conference on Machine Learning*, pp. 25549–25562. PMLR, 2023.

NIST. NIST standard reference database. National Institute of Standards and Technology, 2020. URL https://www.nist.gov/srd.

Nowak, A., Villar, S., Bandeira, A., and Bruna, J. Revised note on learning quadratic assignment with graph neural networks. In *Data Science Workshop*, 2018.

Pal, S., Bhattacharya, M., Islam, M. A., and Chakraborty, C. Chatgpt or llm in next-generation drug discovery and development: Pharmaceutical and biotechnology companies can make use of the artificial intelligence-based device for a faster way of drug discovery and development, Dec 2023. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC10720782/.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017.

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., et al. Global chemical effects of the microbiome include new bile-acid conjugations. *Nature*, 579(7797):123–129, 2020.

Ridder, L., van der Hooft, J. J. J., and Verhoeven, S. Automatic compound annotation from mass spectrometry data using magma. *Mass Spectrometry*, 3(Special_Issue_2): S0033–S0033, 2014. doi: 10.5702/massspectrometry.S0033.

Rolínek, M., Swoboda, P., Zietlow, D., Paulus, A., Musil, V., and Martius, G. Deep graph matching via blackbox differentiation of combinatorial solvers. In *European Conference on Computer Vision*, pp. 407–424, 2020.

Sarlin, P.-E., DeTone, D., Malisiewicz, T., and Rabinovich, A. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4938–4947, 2020.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. The graph neural network model. *Transactions on Neural Networks*, 20(1):61–80, 2008.

Shahneh, M. R. Z., Strobel, M., Vitale, G. A., Geibel, C., Abiead, Y. E., Garg, N., Wagner, B., Forchhammer, K., Aron, A., Phelan, V. V., et al. Modifinder: Tandem mass spectral alignment enables structural modification site localization. *Journal of the American Society for Mass Spectrometry*, 2024.

Shaw, P., Gurram, B., Belanger, D., Gane, A., Bileschi, M. L., Colwell, L. J., Toutanova, K., and Parikh, A. P. Protex: A retrieval-augmented approach for protein function prediction. *bioRxiv*, pp. 2024–05, 2024.

Sinkhorn, R. and Rangarajan, A. A relationship between arbitrary positive matrices and doubly stochastic matrices. *Ann. Math. Statistics*, 35(2):876–879, 1964.

Tian, Z., Zhao, H., Peter, K. T., Gonzalez, M., Wetzel, J., Wu, C., Hu, X., Prat, J., Mudrock, E., Hettinger, R., et al. A ubiquitous tire rubber–derived chemical induces acute mortality in coho salmon. *Science*, 371(6525):185–189, 2021.

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. Sharing and community curation of mass spectrometry data with global natural products social molecular networking. *Nature biotechnology*, 34(8):828–837, 2016.

Wang, R., Yan, J., and Yang, X. Combinatorial learning of robust deep graph matching: an embedding based approach. *Transactions on Pattern Analysis Machine Intelligence*, 2020.

Wang, R., Yan, J., and Yang, X. Neural graph matching network: Learning lawler's quadratic assignment problem with extension to hypergraph and multiple-graph matching. *Transactions on Pattern Analysis Machine Intelligence*, 44(9):5261–5279, 2022.

Wang, R., Guo, Z., Pan, W., Ma, J., Zhang, Y., Yang, N., Liu, Q., Wei, L., Zhang, H., Liu, C., et al. Pygmtools: A python graph matching toolkit. *Journal of Machine Learning Research*, 25:1–7, 2024.

Wei, J. N., Belanger, D., Adams, R. P., and Sculley, D. Rapid prediction of electron–ionization mass spectrometry using neural networks. *ACS central science*, 5(4):700–708, 2019.

Young, A., Röst, H., and Wang, B. Tandem mass spectrum prediction for small molecules using graph transformers. *Nature Machine Intelligence*, 6(4):404–416, 2024a.

Young, A., Wang, F., Wishart, D., Wang, B., Röst, H., and Greiner, R. Fragnnet: A deep probabilistic model for mass spectrum prediction. *arXiv preprint arXiv:2404.02360*, 2024b.

Yu, T., Wang, R., Yan, J., and Li, B. Learning deep graph matching with channel-independent embedding and hungarian attention. In *International Conference on Learning Representations*, 2020.

Zanfir, A. and Sminchisescu, C. Deep learning of graph matching. In *Computer Vision and Pattern Recognition*, pp. 2684–2693, 2018.

Zhang, M. and Li, P. Nested graph neural networks. *Advances in Neural Information Processing Systems*, 34:15734–15747, 2021.

Zhang, P.-D., Peng, X., Han, R., Chen, T., and Ma, J. Rag2mol: Structure-based drug design based on retrieval augmented generation. *bioRxiv*, pp. 2024–10, 2024.

Zhu, H., Liu, L., and Hassoun, S. Using graph neural networks for mass spectrometry prediction. *arXiv preprint arXiv:2010.04661*, 2020.

A. Retrieval Accuracy on Scaffold Split

Following most peer methods, the retrieval accuracy evaluation in the main paper (Table 1) mainly focuses on random split and all positive adduct types. We also include the retrieval accuracy on scaffold split as follows. Scaffold split is considered more challenging than random split as it separates structures with different molecular scaffolds into different training or testing sets, requiring the model with more generalization ability for out-of-distribution structures. MARASON maintains its performance superiority over the baselines on scaffold split. Since scaffold split is less studied in peer methods there are fewer baselines in Table 1. Multiple random restarts are not considered here because the standard deviation in random split is relatively small, indicating that most peer methods are somewhat stable against random seeds.

Table 5. Retrieval Accuracy upon NIST20 Dataset with scaffold split on positive adduct types.

Accuracy @ Top-k	1	2	3	4	5	8	10
Graff-MS (Murphy et al., 2023)	0.142	0.265	0.36	0.446	0.508	0.636	0.703
MassFormer (Young et al., 2024a)	0.178	0.318	0.422	0.506	0.568	0.706	0.768
ICEBERG (Goldman et al., 2024)	0.206	0.396	0.519	0.604	0.658	0.769	0.815
MARASON (ours)	0.283	0.464	0.567	0.645	0.700	0.814	0.853

B. Retrieval Accuracy with [M+H]+ Only

One design choice in MS/MS simulator development is how many adduct types are supported by the model. Our MARASON aims to cover most adduct types in the NIST (2020) database, while some methods e.g. FraGNNet (Young et al., 2024b) only support the most common adduct type—[M+H]+. It is still feasible to compare these methods by restricting testing adduct type as [M+H]+, where the retrieval accuracy is shown as follows in Table 6. The performance of FraGNNet is quoted from their paper (therefore missing accuracies at $k = 2, 4, 8$, also missing error bars), and other baselines are implemented with our benchmark. It is worth noting that although Young et al. (2024b) report better retrieval accuracy of FraGNNet than all peer methods, our reevaluation of all methods on the [M+H]+-only subset shows that ICEBERG (Goldman et al., 2024) still outperforms, suggesting that single-bond breaking in MS/MS simulator design might not be superior to multiple-bond breaking. Incorporating collision energy for ICEBERG also improves the retrieval accuracy on [M+H]+, and our MARASON reaches state-of-the-art retrieval accuracy.

Table 6. Retrieval Accuracy (mean ± 1.96 standard deviation of three random seeds) upon NIST (2020) dataset with random (InChI key) split on the [M+H]+ adduct type.

Accuracy @ Top-k	1	2	3	4	5	8	10
Graff-MS (Murphy et al., 2023)	0.211±0.004	0.365±0.009	0.472±0.015	0.551±0.013	0.608±0.005	0.723±0.008	0.775±0.009
MassFormer (Young et al., 2024a)	0.252±0.001	0.422±0.002	0.539±0.005	0.617±0.007	0.675±0.004	0.794±0.010	0.843±0.006
FraGNNet (Young et al., 2024b)	0.238	-	0.504	-	0.652	-	0.831
ICEBERG (Goldman et al., 2024)	0.251±0.016	0.454±0.004	0.576±0.0060.718	0.654±0.004	0.711±0.007	0.810±0.001	0.850±0.009
ICEBERG (w/ collision energy)	0.270±0.016	0.487±0.009	0.611±0.011	0.679±0.013	0.735±0.013	0.840±0.009	0.877±0.008
MARASON (ours)	**0.331±0.002**	**0.520±0.005**	**0.633±0.010**	**0.706±0.005**	**0.754±0.002**	**0.849±0.003**	**0.885±0.003**

C. Cosine Similarity

We include the detailed cosine similarity numbers from Fig. 3 for better reproducibility in future works.

Table 7. Experimental results in line with Fig. 3 of cosine similarity. Random split has mean ± 1.96 standard deviation across 3 random seeds.

Models	Random	Scaffold
Retrieved Spec	0.359	0.334
3DMolMS (Hong et al., 2023)	0.51±0.001	0.466
NEIMS (FFN) (Wei et al., 2019)	0.617±0.001	0.546
NEIMS (GNN) (Zhu et al., 2020)	0.694±0.001	0.643
FixedVocab (Murphy et al., 2023)	0.704±0.001	0.658
MassFormer (Young et al., 2024a)	0.721±0.004	0.682
SCARF (Goldman et al., 2023)	0.726±0.002	0.669
ICEBERG (Goldman et al., 2024)	0.727±0.002	0.699
ICEBERG (w/ colli.)	0.735±0.005	0.711
MARASON (ours)	**0.765±0.002**	**0.725**

D. Evaluation of Matching Patterns: Visualizing Failure Modes of Traditional Graph Matching

For the qualitative analysis of matching pairs identified by MARASON, Fig. 5 in the main text visualizes only six fragment pairs due to space constraints. Here, ICEBERG-Generate predicts 78 fragments for the target molecule, and we provide a complete list of all 78 matched fragment pairs. As shown in Fig. 6, the original target structure (left) and the original reference structure (right) are included, with fragments highlighted by their respective atoms and bonds.

In Fig. 6, the top 50 matched pairs, which exhibit higher similarity scores, predominantly correspond to exact structural matches. In contrast, the lower-scored pairs include more fragments that, while structurally distinct, follow similar fragmentation pathways. Notably, the learned matching function in MARASON effectively identifies an informative and chemically interpretable fragment-matching strategy. This ensures that the retrieved intensity information is accurately assigned to the correct target fragments, thereby enhancing MS/MS simulation accuracy.

For matching pairs identified by RRWM (Reweighted Random Walk Matching) (Cho et al., 2010), which relies on fixed graph-matching affinity metrics, a greater number of unreasonable matchings occur, particularly when the fragment structures are not identical (see Fig. 7). This issue is especially pronounced for pairs 65–78, where the similarity score falls below 0.2. The limitation of traditional graph matching arises from its fixed affinity metric: for identical fragments, it functions effectively, producing accurate matches; however, for non-identical fragments, it assigns noisy similarity scores, often categorizing them as matching "outliers" (fragments with low affinity scores to all others). As a result, these fragments are incorrectly matched with unrelated structures.

Additionally, within the tail distribution of challenging fragment pairs, the learned neural graph-matching method yields higher similarity scores, demonstrating its superior ability to capture meaningful fragment correspondences.

Pair 1, Similarity:0.997 | Pair 2, Similarity:0.995 | Pair 3, Similarity:0.994 | Pair 4, Similarity:0.986 | Pair 5, Similarity:0.985 | Pair 6, Similarity:0.985

Pair 7, Similarity:0.98 | Pair 8, Similarity:0.975 | Pair 9, Similarity:0.968 | Pair 10, Similarity:0.966 | Pair 11, Similarity:0.964 | Pair 12, Similarity:0.961

Pair 13, Similarity:0.95 | Pair 14, Similarity:0.949 | Pair 15, Similarity:0.947 | Pair 16, Similarity:0.946 | Pair 17, Similarity:0.933 | Pair 18, Similarity:0.922

Pair 19, Similarity:0.912 | Pair 20, Similarity:0.885 | Pair 21, Similarity:0.883 | Pair 22, Similarity:0.883 | Pair 23, Similarity:0.869 | Pair 24, Similarity:0.868

Pair 25, Similarity:0.865 | Pair 26, Similarity:0.859 | Pair 27, Similarity:0.859 | Pair 28, Similarity:0.858 | Pair 29, Similarity:0.858 | Pair 30, Similarity:0.845

Pair 31, Similarity:0.837 | Pair 32, Similarity:0.829 | Pair 33, Similarity:0.811 | Pair 34, Similarity:0.805 | Pair 35, Similarity:0.782 | Pair 36, Similarity:0.776

Pair 37, Similarity:0.772 | Pair 38, Similarity:0.728 | Pair 39, Similarity:0.701 | Pair 40, Similarity:0.675 | Pair 41, Similarity:0.673 | Pair 42, Similarity:0.627

Pair 43, Similarity:0.613 | Pair 44, Similarity:0.608 | Pair 45, Similarity:0.579 | Pair 46, Similarity:0.567 | Pair 47, Similarity:0.566 | Pair 48, Similarity:0.566

Pair 49, Similarity:0.561 | Pair 50, Similarity:0.554 | Pair 51, Similarity:0.527 | Pair 52, Similarity:0.525 | Pair 53, Similarity:0.495 | Pair 54, Similarity:0.484

Pair 55, Similarity:0.454 | Pair 56, Similarity:0.454 | Pair 57, Similarity:0.449 | Pair 58, Similarity:0.445 | Pair 59, Similarity:0.435 | Pair 60, Similarity:0.413

Pair 61, Similarity:0.407 | Pair 62, Similarity:0.404 | Pair 63, Similarity:0.4 | Pair 64, Similarity:0.384 | Pair 65, Similarity:0.379 | Pair 66, Similarity:0.373

Pair 67, Similarity:0.373 | Pair 68, Similarity:0.355 | Pair 69, Similarity:0.348 | Pair 70, Similarity:0.328 | Pair 71, Similarity:0.32 | Pair 72, Similarity:0.301

Pair 73, Similarity:0.29 | Pair 74, Similarity:0.274 | Pair 75, Similarity:0.268 | Pair 76, Similarity:0.245 | Pair 77, Similarity:0.223 | Pair 78, Similarity:0.196
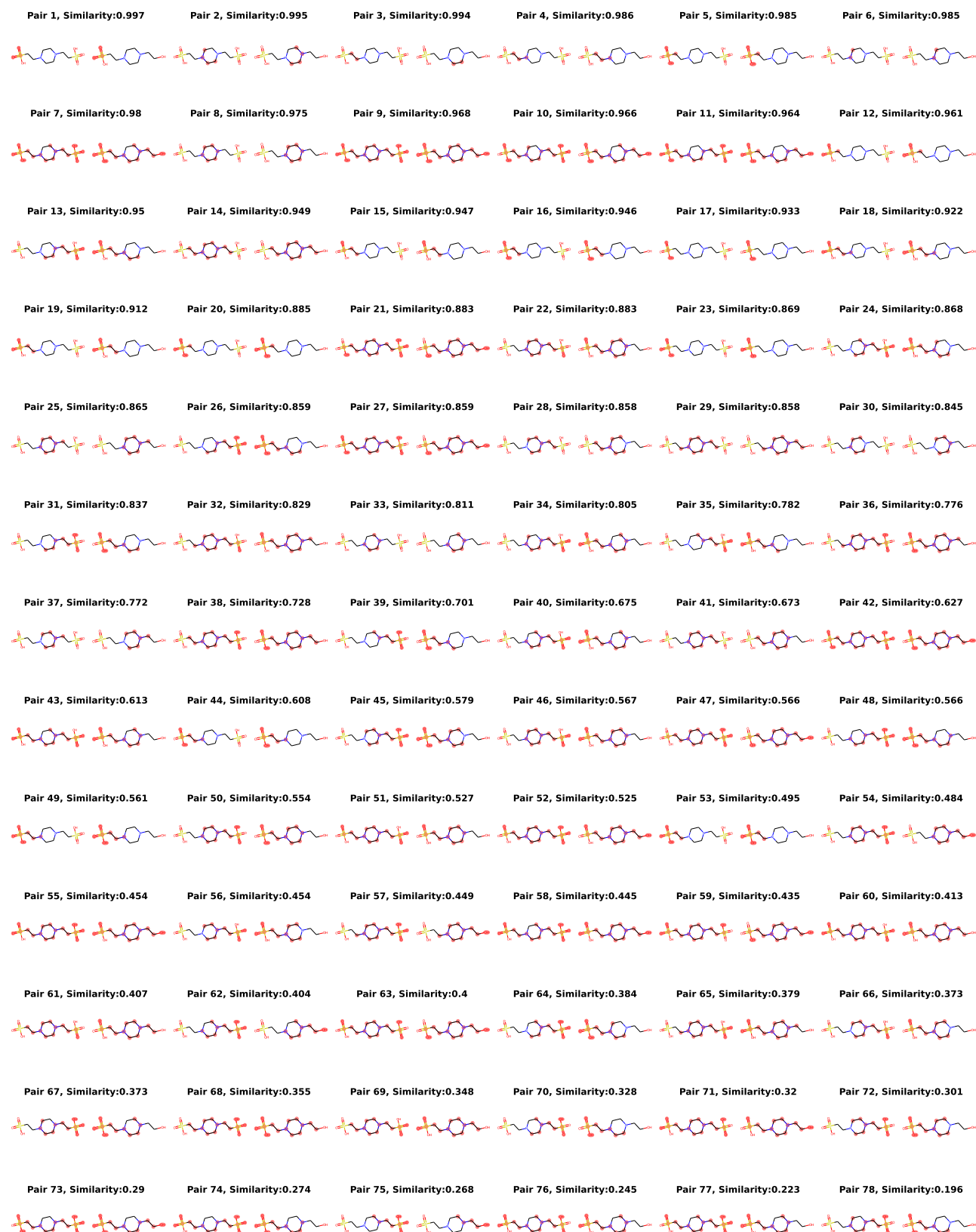
Figure 6. A full list of fragment pairs generated by MARASON, sorted by the learned Softmax scores. Structures on the left are from the target fragmentation DAG and structures on the right are from the reference.
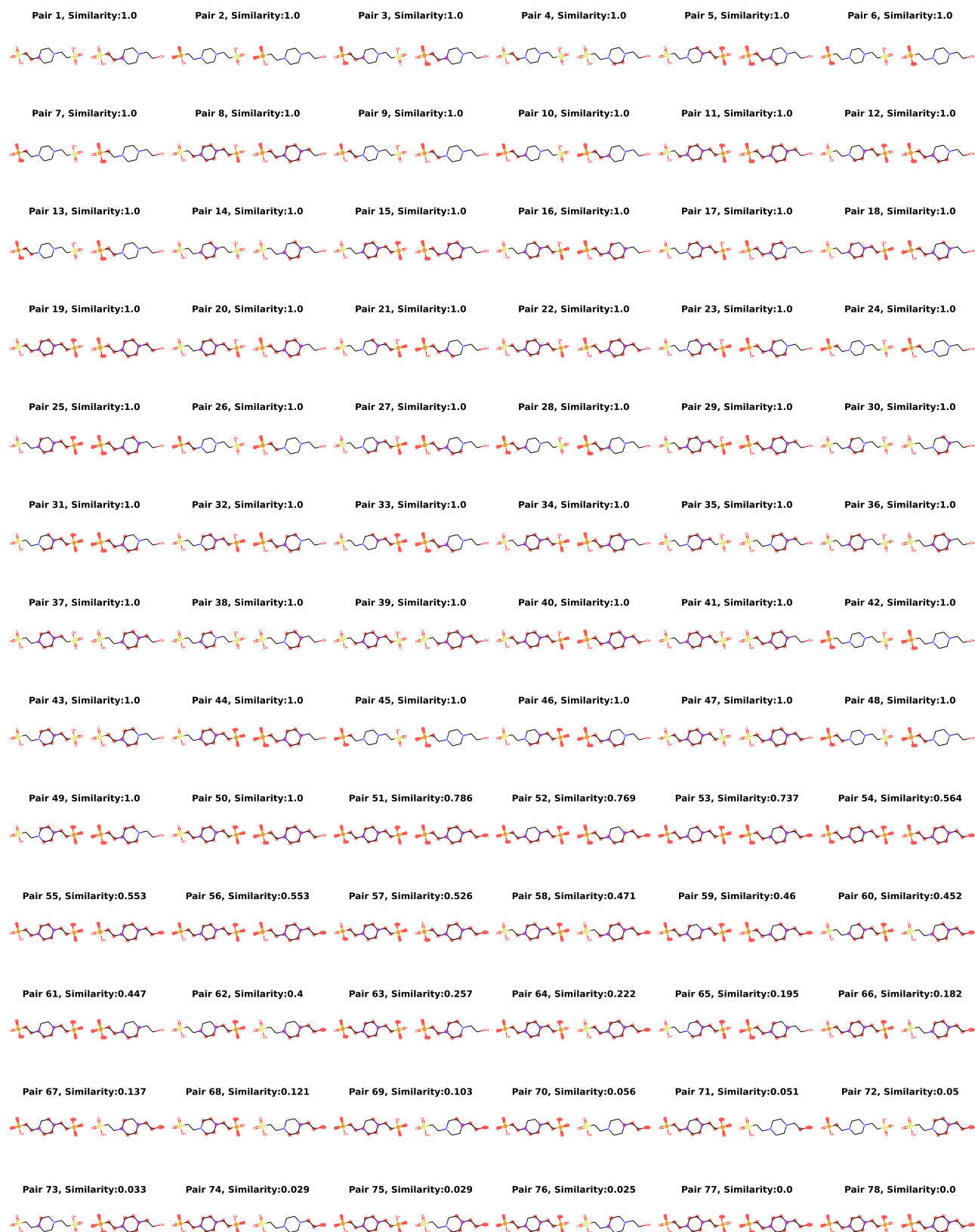
Figure 7. A full list of fragment pairs generated by RRWM graph matching ([Cho et al.](), [2010]()), sorted by fixed Tanimoto similarity scores. Structures on the left are from the target fragmentation DAG and structures on the right are from the reference.