Journal of Molecular Graphics and Modelling 137 (2025) 108985

Contents lists available at ScienceDirect

Journal of Molecular Graphics and Modelling

journal homepage: www.elsevier.com/locate/jmgm

Fragment-level feature fusion method using retrosynthetic fragmentation algorithm for molecular property prediction

Qifeng Jia [a], Yekang Zhang [a], Yihan Wang [c], Tiantian Ruan [c], Min Yao [c,*], Li Wang [b,**]

[a] School of Information Science and Technology, Nantong University, Nantong, 226001, China
[b] Research Center for Intelligent Information Technology, Nantong University, Nantong, China
[c] Department of Immunology, Medical School, Nantong University, Nantong, China

ARTICLE INFO

Keywords:
Artificial Intelligence
Molecular property prediction
Fragmentation
Feature fusion

ABSTRACT

Recent advancements in Artificial Intelligence (AI) and deep learning have had a significant impact on drug discovery. The prediction of molecular properties, such as toxicity and blood-brain barrier (BBB) permeability, is crucial for accelerating drug development. The accuracy of these predictions largely depends on the selection of molecular descriptors. Self-supervised learning (SSL) has gained prominence due to its strong generalization capabilities. Graph contrastive learning (GCL), a type of SSL, is particularly useful in this context. Current GCL methods for molecular graphs use various data augmentation techniques, which may potentially alter the inherent structure of molecules. Additionally, traditional single-perspective representations do not fully capture the complexity of molecules. We present RFA-FFM (Fragment-level Feature Fusion Method using Retrosynthetic Fragmentation Algorithm), which integrates molecular representations from multiple perspectives. This method employs two strategies: (1) contrasting chemical information from fragments generated by two retrosynthetic methods to provide detailed contrastive insights; (2) fusing chemical information at different levels of molecular hierarchy, including the entire molecule and its fragments. Experiments show that RFA-FFM enhances the performance of deep learning models in predicting molecular properties, improving ROC-AUC scores by 0.3 %–2.6 % compared to baselines across four classification benchmarks. Case studies on hepatitis B virus datasets demonstrate that RFA-FFM outperforms baselines by 7 %–11 %. When compared to BPE and CC-Single fragmentation algorithms, RFA-FFM shows a 2 %–4 % improvement in BBB permeability tasks, thus demonstrating its effectiveness in predicting molecular properties.

1. Introduction

1D sequence representation is a frequently employed approach for representing molecules. Wu et al. proposed an efficient high-utility motif finding algorithm that is significant for gene representation, the study of the mechanisms of gene expression regulation, and the discovery of biological functional sites [1]. 2D graph representations are frequently employed for the purpose of molecular characterization. In this approach, molecules are conceptualized as graphs, where atoms or functional groups are represented as nodes and chemical bonds are represented as edges. Graph Neural Networks (GNNs) have demonstrated significant achievements in a range of graph-related tasks. Considerable efforts have been dedicated to the development of GNN models for the prediction of molecular properties. The underlying concept involves treating the topological structure of atoms and bonds as a graph and utilizing GNN encoders to transform each molecule into a representation vector. This is then followed by a prediction module that is specific to certain attributes [2–8].

Substructure and fragment descriptors facilitate the analysis and prediction of the structure-activity relationships of molecules. These descriptors implicitly capture a significant amount of chemical information [9]. The breaking of retrosynthetically interesting chemical substructures (BRICS) is a molecular decomposition algorithm that is based on domain knowledge. It is used in conjunction with the Retrograde Computation Approach Program (RECAP), both of which are fragmentation algorithms rooted in retrosynthetic principles [10]. Zhu

* Corresponding author.
** Corresponding author.
E-mail addresses: 2230310054@stmail.ntu.edu.cn (Q. Jia), 2330310037@stmail.ntu.edu.cn (Y. Zhang), wyh18362641149@outlook.com (Y. Wang), rttgyfyyx@163.com (T. Ruan), erbei@ntu.edu.cn (M. Yao), wangli@ntu.edu.cn (L. Wang).

https://doi.org/10.1016/j.jmgm.2025.108985
Received 18 December 2024; Received in revised form 15 February 2025; Accepted 17 February 2025
Available online 21 February 2025

et al. utilized GNN encoders to input the attribute graphs and fragments of molecules, allowing for the simultaneous acquisition of global and hierarchical representations [11]. However, current methods primarily rely on a single strategy, such as feature fusion between molecules and fragments, for molecular characterization. Exploring molecular representation using multiple strategies has the potential to further optimize the quality of the representation.

Furthermore, the scarcity of labeled data is a significant obstacle that hinders the prediction of molecular properties by GNN models. Self-supervised learning addresses this issue by not requiring labeled data. One method that achieves this is Graph Contrastive Learning (GCL). However, existing GCL approaches often employ different data augmentation techniques for graphs, which can potentially alter the semantics of graphs across domains. Currently, most GCL methods for molecular graphs still adhere to this data augmentation paradigm, which inevitably leads to changes in the natural structure of molecules. For instance, You et al. proposed discarding atoms, perturbing edges, and masking attributes to increase the amount of data [12]. However, since every atom influences the molecular properties, randomly dropping and perturbing atoms may disrupt the molecular structure. In fact, the generation of 2D molecules and fragment views involves different methods. 2D molecular graphs are directly derived from the structural formulas of compounds, while fragment graphs are typically obtained using tools like RDKit and sequence segmentation algorithms. Considering that molecular graphs and their fragments provide chemical and geometric information at different levels and complement each other, examining a molecule from different levels presents an excellent opportunity to design a unique graph contrastive learning scheme for molecular graphs. The iMolCLR model enhances molecular contrastive learning by reducing false negative instances and contrasting decomposed chemical fragments. iMolCLR does not treat all negative pairs equally, but rather encourages similar molecules to have closer representations than dissimilar ones. In addition to molecular-level contrast, different substructures decomposed by the BRICS algorithm are considered as contrasting negative pairs [13]. This decomposition strategy retains the major structural features of compounds, compelling molecular representation to distinguish important functional groups within the molecule. However, the aforementioned fragment-based methods solely utilize fragmentation algorithms to obtain fragment representations without exploring the principles and roles of the fragmentation algorithms themselves. Both BRICS and RECAP obtain fragments by breaking retrosynthetically relevant chemical bonds, and chemical reactions are closely linked to the activity properties of molecules. Breaking these bonds may more accurately extract key fragments, thereby promoting the prediction of molecular properties.

To address the aforementioned issues, this study proposes RFA-FFM, as depicted in Fig. 1. This framework employs two retrosynthetic fragmentation algorithms to incorporate multi-perspective and multi-level molecular information, thereby optimizing molecular representation. The key components of the model are as follows: (1) Considering molecular representations from multiple perspectives: one aspect involves contrastive learning at the fragment level, while the other involves implementing cross-attention mechanisms at three levels between molecules and fragments. (2) Utilizing fragments obtained from two retrosynthetic fragmentation methods to compare their chemical information, refine the scope of negative sample pairs, and derive more detailed contrastive information. (3) Integrating chemical information at multiple levels of molecules, including the molecule itself and the two types of fragments.

Experiments demonstrate that the RFA-FFM model significantly enhances the performance of GNN models in molecular property prediction benchmarks. It outperforms state-of-the-art supervised learning baseline models by a range of 0.3 %–2.6 % across four classification tasks and reduces errors by approximately 0.7 %–3 % in regression tasks.

The RFA-FFM model comprises two modules: the fragment contrastive learning module and the hierarchical attention module. These modules investigate molecular representation and perform property prediction from two distinct perspectives. The fragment contrastive learning module implements contrastive learning on molecular fragments to obtain a contrastive loss. The hierarchical attention module applies cross-attention mechanisms at three levels between molecules and fragments to compute the normalized temperature-scaled cross-

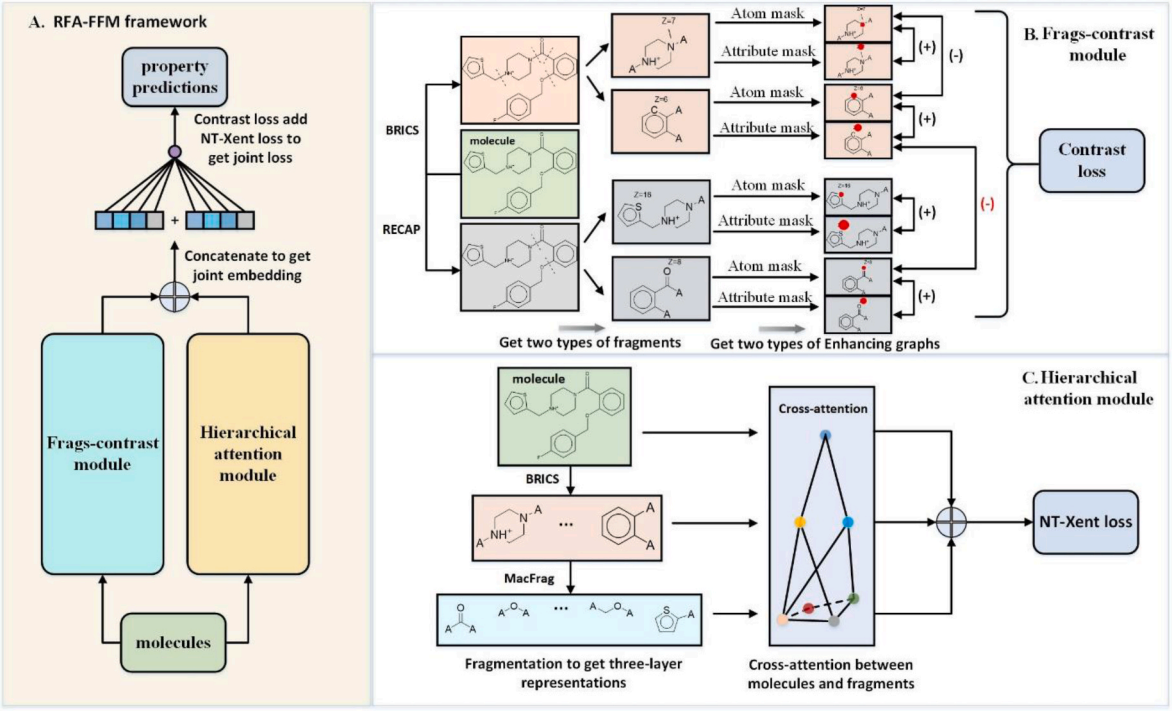

Fig. 1. Framework of RFA-FFM. (A) Overview of the overall workflow of RFA-FFM. (B) Schematic overview of the Frags-contrast module. (C) Schematic overview of the Hierarchical attention module.

entropy (NT-Xent) loss function. Finally, the two loss functions are combined to form a joint loss function, which is used to fine-tune downstream property prediction tasks. Detailed descriptions of both modules will be provided below.

2. Fragment contrastive learning module

The contrastive learning module for the fragment is depicted in Fig. 1 (A). Molecular processing is carried out using the BRICS and RECAP fragment decomposition algorithms to acquire molecular fragments. Subsequently, two graph augmentation methods are employed to produce augmented graphs. These augmented graphs are encoded using GIN networks, and contrastive learning is performed between the fragments. The workflow of this framework will be elaborated below.

2.1. RECAP and BRICS decomposition algorithms

RECAP is a unique method that applies 11 distinct rules to break down active molecules into active building blocks. It specifically targets bonds that are commonly formed through chemical reactions and splits these bonds. The 11 predefined bond types ensure that the resulting fragments are suitable for synthetic chemistry. This process creates a fragment space, which is not just a collection of fragments but also includes rules for reassembling the fragments by combining their respective chemical motifs. A significant aspect of this algorithm is its ability to preserve ring structures. The specific rules are depicted in Fig. 2. Besides, we select three molecules as examples to describe the fragmentation methods of rules 2, 4 and 6 in RECAP. The SMILES of these three molecules are CC(=O)CC(=O)OC, C1=CC=CC=C1NC(=O)NC(=O)O, and CCCC=C(C)C, respectively. Fig. 3 shows the three example molecules and the segmentation positions according to the rules. The dashed lines in the figure indicate the chemical bonds that are broken during the splitting process.

BRICS was developed after RECAP and provides a complementary set of rules for reassembling corresponding chemical motifs to compile the fragment space. It consists of 16 cleavage rules that correspond to 16 different chemical environments. This algorithm breaks the chemically relevant bonds in molecules to obtain various fragment information, considering the chemical environment and surrounding substructures of each cleaved bond. Unlike RECAP, which identifies the bonds that can be broken, BRICS identifies the resulting fragment structures. An important feature of BRICS is the inclusion of ring bond cleavage. The specific rules are presented in Fig. 4.

Both the RECAP and BRICS fragmentation algorithms share similarities and differences. They both involve breaking retrosynthetically relevant chemical bonds to obtain fragments. However, they differ in that BRICS defines additional cleavable bonds, such as certain ring bonds. While both algorithms perform molecular fragmentation based on different principles—BRICS focuses more on the relationship between substrates and products in chemical reactions, and RECAP concentrates on identifying potential bond cleavage sites—they provide different molecular fragments, thereby increasing the diversity of fragment data. Utilizing diversified molecular fragments can reduce the risk of overfitting on specific datasets, thereby enhancing the model's generalization capability on unseen data.

Traditional fragmentation algorithms, such as Byte-Pair Encoding (BPE), are primarily used for preprocessing textual data, especially suitable for Natural Language Processing (NLP) tasks like machine translation [14]. However, BPE merges character pairs based on frequency, which may overlook important information in molecular structures. The properties of chemical molecules often depend on the interactions between their atoms, not just the most frequent parts. Therefore, BPE may fail to capture the structural information crucial for molecular properties.

In contrast, BRICS and RECAP obtain fragments by breaking retrosynthetically relevant chemical bonds and identifying important functional groups involved in chemical reactions. These functional groups are often associated with specific biological activities or other molecular properties. By decomposing molecules into these functional groups, it becomes easier to discover correlations between them and the properties to be predicted. Fragments obtained through retrosynthetic analysis may be directly related to certain molecular properties. For instance, in drug design, specific chemical structures can be closely linked to the activity of drugs. By learning the relationship between these fragments and their properties, prediction accuracy can be improved. Consequently, these two algorithms have complementary features, and contrastive learning between fragments obtained from these algorithms is meaningful for molecular property prediction tasks.

2.2. Contrastive learning framework

The objective of contrastive learning is to construct positive and negative sample pairs in order to learn representations. As depicted in Fig. 1(A), the contrastive learning framework consists of three main components: molecular graph augmentation, a GIN-based encoder, and a contrastive loss. MolCLR introduces three novel methods for molecular graph augmentation: atom masking, edge deletion, and subgraph removal [15]. However, these methods often lead to the loss of original graph information. Therefore, the fragment contrastive learning module utilizes attribute masking, which masks the attributes of nodes and edges instead of the nodes and edges themselves [16]. The fragment contrastive learning module employs GIN to predict these attributes,

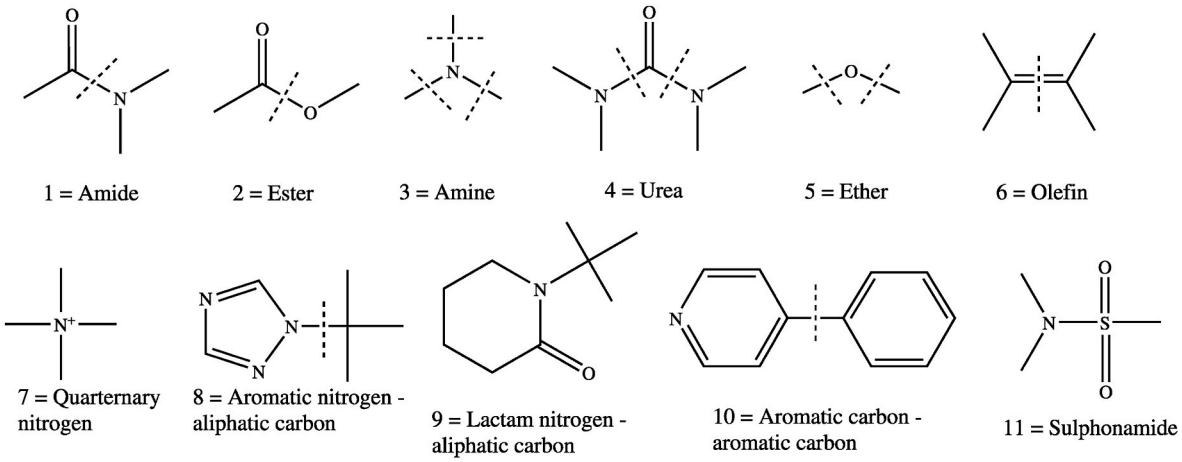

Fig. 2. Splitting rules of the RECAP algorithm.

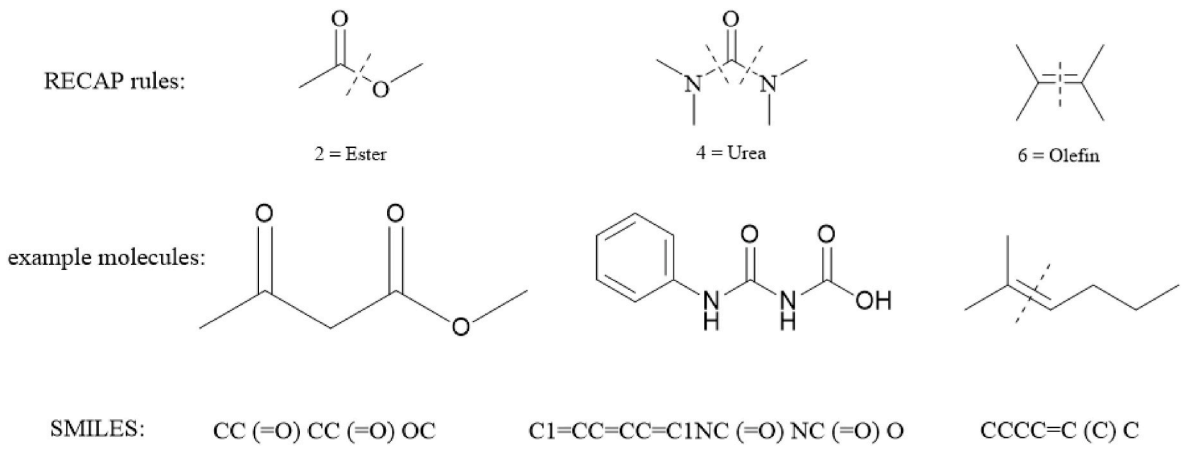

Fig. 3. Splitting positions of example molecules using the RECAP Algorithm.

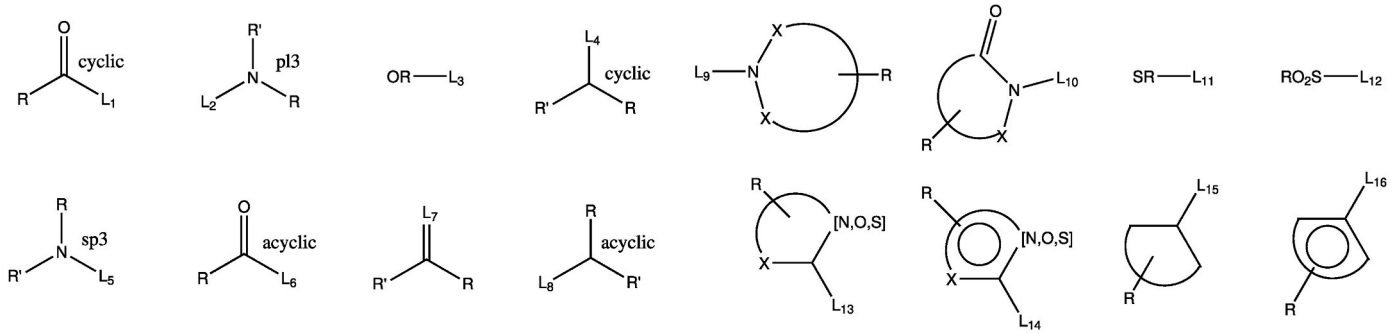

Fig. 4. Splitting rules of the BRICS algorithm.

thus avoiding the information loss caused by directly masking nodes and edges. Attribute masking encourages the model to utilize contextual information (i.e., remaining attributes) to recover the masked vertex attributes. The underlying assumption is that the absence of partial vertex attributes does not significantly impact the model's predictions. Atom masking and feature masking are used in combination, with a random masking ratio of 25 % for both.

Given a batch of molecules $\{m_1, \ldots, m_k\}$, two augmented graphs G_i and G_j are obtained, where $i = 2K - 1$ and $j = 2K$. Augmented graphs from the same molecule form positive pairs, while those from different molecules form negative pairs. These augmented graphs are processed through a GIN-based encoder to produce representations h_i. The representations h_i are then mapped to latent vectors z_i via a nonlinear projection head $g(\cdot)$. The contrastive loss function is applied to the 2K latent vectors from the projection heads, aiming to maximize the consistency between positive pair vectors (e.g., z_i and z_j) and minimize the consistency between negative pair vectors (e.g., z_i and z_n, z_j and z_n).

2.3. Graph Isomorphism Network (GIN) encoder

A molecular graph G is described as $G = (V,E)$, where each node $v \in V$ represents an atom, and each edge $e_{uv} \in E$ represents a chemical bond between atoms u and v. Each node v has a feature vector x_v, and each edge has a feature vector e_{uv}.

In this study, we utilize a Graph Isomorphism Network (GIN) as the basis for a GNN encoder. To incorporate edge features, the node aggregation operation is extended in the following manner:

$$a_v^{(n)} = \sum_{u \in N(v)} \sigma\left(h_u^{(n-1)} + \varepsilon_{uv}\right) \quad (1)$$

where $\sigma(\cdot)$ is a nonlinear activation function. The combination operation is modeled using a sum followed by a Multilayer Perceptron (MLP):

$$h_v^{(n)} = \mathrm{MLP}\left(h_v^{(n-1)} + a_v^{(n)}\right) \quad (2)$$

Pooling operations are utilized to combine node embeddings into a singular graph-level representation. More specifically, the readout operation is executed by employing average pooling across all nodes to acquire a graph representation for each molecule.

The rationale behind selecting GIN as the encoder is that, in comparison to other GNN networks, GIN introduces a straightforward and effective message-passing operation that is easily implementable. Furthermore, GIN incorporates a superior sum aggregation operation, enabling effective learning of the structural information within the network.

2.4. Handling negative sample pairs

Most current graph contrastive learning approaches operate at a coarse-grained level. For instance, positive sample pairs are defined as a pair of graphs derived from the same molecule using different graph augmentation methods, while all other pairs are categorized as negative samples. This approach often overlooks the information contained in structurally similar molecules. The iMolCLR model improves molecular contrastive learning by reducing false negatives; it does not treat all negative sample pairs equally. Instead, it encourages representations of similar molecules to be closer to each other compared to dissimilar molecules. In addition to molecular-level contrast, substructures obtained through BRICS decomposition are considered as negative pairs in the contrastive learning process. This decomposition strategy retains the major structural features of compounds, forcing the molecular representation to distinguish important functional groups within the molecule [13].

Most existing models perform contrastive learning at the molecular level, with only a few studies focusing on motif-level contrast, which are typically based on a single fragmentation rule, resulting in relatively uniform motif types. The fragment contrastive learning module draws inspiration from the motif-level contrast approach in the iMolCLR model and combines two similar fragmentation methods, BRICS and RECAP, encouraging the model to learn the similarity between the fragments obtained from these methods. For similar molecular fragments derived from the two fragmentation methods, the model is encouraged to generate representations that are closer in the embedding space. Additionally, the module calculates the similarity within the fragments obtained from both BRICS and RECAP separately.

In this study, fingerprint similarity metrics are selected to represent molecular similarity using MACCS molecular fingerprints. MACCS fingerprints have a predetermined length of 166 bits, which facilitates the storage and calculation of similarity. Additionally, each bit in the fingerprint corresponds to a distinct chemical structural characteristic, such as functional groups and ring structures, enabling the model to acquire significant information within and across fragments. Using the NT-Xent loss function, we can obtain 2N latent vectors $\{z_1, \ldots, z_{2N}\}$ given a batch of N molecules. The NT-Xent loss function for positive pairs (z_i, z_j) is formulated as shown in Equation (1):

$$\ell_{i,j} = -\log \frac{\exp\left(\frac{\cos(z_i, z_j)}{\tau}\right)}{\sum_{n=1}^{2N} 1_{n \neq i} \exp\left(\frac{\cos(z_i, z_n)}{\tau}\right)} \tag{3}$$

where τ is the temperature parameter. By minimizing Equation (1), the cosine similarity of positive pairs is expected to be the highest among all $2N - 1$ pairs.

However, this contrastive loss treats all negative sample pairs equally, which may pose a challenge for the model to effectively learn information from different negative sample pairs. This is because there may be instances where negative pairs exhibit structural similarities.

To tackle this issue, the NT - Xent loss function is optimized. Specifically, to measure the similarity between fragments, the iMolCLR model employs ECFP fingerprints. ECFP fingerprints (Extended-Connectivity Fingerprints) are a type of molecular fingerprint technology widely utilized in fields such as cheminformatics and drug design. ECFP, which is based on the binary vector representation of molecular structures, is employed to describe the chemical features of molecules. By encoding the information of atoms and bonds within a molecule, it transforms the molecule into a vector composed of 0 and 1, where each bit represents the presence or absence of a specific structural feature or substructure within the molecule. This method effectively captures the structural and chemical features of molecules, including the molecular skeleton structure, functional groups, and types of chemical bonds. Utilizing ECFP fingerprints facilitates the calculation of similarity between molecules or molecular fragments. The iMolCLR model further enhances this optimization through ECFP. Pairs of fragments with high ECFP similarity are encouraged to be closer in the representation domain.

Similarly, our RFA-FFM model employs MACCS (Molecular Access System Chemical Structure Keys) to calculate the similarity between fragments, particularly those obtained from the BRICS and RECAP fragmentation algorithms. MACCS is another molecular fingerprint technology based on a binary coding system of molecular structures. It transforms the chemical structure information of a molecule into a fixed-length vector composed of 0 and 1, thereby allowing for the rapid identification and comparison of the structural features of molecules. MACCS encodes based on a predefined set of atom types and structural fragments. By matching the structures in the molecule with these fixed templates, the values at each position in the fingerprint vector are established. It can also be employed to calculate the similarity between molecules. The MACCS fingerprint adheres to a clear and unified standard. Its predefined set of structures is specifically targeted at important and common chemical structures, enabling the swift identification of key structural features. This capability is particularly advantageous for research focused on the effects related to specific functional groups or structural fragments, offering an advantage in comparison to ECFP. Hence, we employ the MACCS fingerprint. The similarity measure between the two latent vectors (z_i, z_n) of a negative sample pair (M_i, M_n) is adjusted by a weight coefficient w_{in} to compensate for false negative pairs, as shown in Equations (4) and (5).

$$\ell_{i,j}^{w} = -\log \frac{\exp\left(\frac{\cos(z_i, z_j)}{\tau}\right)}{\sum_{n=1}^{2N} 1_{n \neq i} \exp\left(\frac{w_{in} \cos(z_i, z_n)}{\tau}\right)} \tag{4}$$

$$w_{in} = 1 - \lambda \mathrm{FPSim}(Mi, Mn) \tag{5}$$

where $w_{in} \in [0, 1]$. The function $\mathrm{FPSim}(M_i, M_n)$ computes the MACCS fingerprint similarity of the given two samples (M_i, M_n). The hyperparameter $\lambda \in [0, 1]$ determines the degree of weight adjustment. A higher λ leads to greater weight adjustment. When $\lambda = 0$, no weight adjustment is performed. In other words, for fragment pairs with high fingerprint similarity, the model is encouraged to learn representations that are closer in the embedding space by adding extra weight to them. For instance, in Fig. 1(A), the fragment pairs connected by (−) with orange color exhibit structural similarity. The model assigns additional weight to these pairs, thereby reducing their negative impact. Additionally, two types of augmentation methods are employed for molecular fragments: atom masking and feature masking. The augmented graphs obtained through these two methods for the same fragment are classified as positive sample pairs, which encourages the model to discern the distinctions between the two types of graph augmentation methods.

By optimizing the NT-Xent loss function, the model can prevent the learning of incorrect negative sample pairs and can also acquire knowledge about the differences and relationships among the fragments obtained from the two fragmentation algorithms. This approach facilitates the prediction of molecular properties.

3. Hierarchical attention module

The hierarchical attention framework for molecules is depicted in Fig. 1(B). Using the fragment decomposition algorithm MacFrag, the molecules are processed with both BRICS and MacFrag decomposition algorithms to obtain molecular fragments. Subsequently, a cross-attention mechanism is applied across three levels: the whole molecule, fragments obtained from BRICS, and fragments obtained from MacFrag. This encourages the model to learn the associations between different levels of information. Finally, the NT-Xent loss function is utilized for downstream molecular property prediction tasks.

3.1. MacFrag decomposition algorithm

Diao et al. developed the MacFrag decomposition algorithm, which establishes systematic guidelines for breaking ring bonds and expands upon the BRICS method to generate smaller molecular fragments, thereby resulting in a larger fragment space [17]. In comparison to the BRICS algorithm, the fragments obtained from MacFrag are more detailed, enabling them to convey a greater amount of information. The relationship between the fragments obtained from BRICS and those obtained from MacFrag can be likened to the relationship between molecules and fragments.

3.2. Graph neural network encoder

A molecule can be represented by an undirected graph $G = (V, E)$, where V is a set of nodes (atoms) with $|V| = N$, and $E \neq V \times V$ is a set of

edges (bonds) with $|E| = M$. Let each node $v_i \in V$, edge $e_{ij} = (v_i, v_j) \in E$, have initial attributes $x_i \in R^{d_n}$ and $e_{ij} \in R^{d_e}$, where d_n represents the dimensionality of node features and d_e represents the dimensionality of edge features. The BRICS algorithm decomposes G into a set of fragments, denoted by the virtual atoms $S = \{S_1, S_2, \ldots, S_p\} = \text{BRICS}(G)$. Similarly, the MacFrag algorithm decomposes G into another set of fragments, denoted by the virtual atoms $T = \{T_1, T_2, \ldots, T_q\} = \text{MacFrag}(G)$.

For the task of predicting molecular properties, the majority of GNN-based models adhere to the message-passing paradigm, which consists of three functions (message-passing function, aggregation function, and update function) that are used to iteratively extract atomic features. The message-passing stage at the n-th layer can be represented as:

$$m_i^{(n)} = \text{aggregate}^{(n)}\left(\left\{\text{message}^{(n)}\left(h_i^{(n-1)}, h_j^{(n-1)}, e_{ij}\right) : j \in N(i)\right\}\right) \quad (6)$$

$$h_i^{(n)} = \text{update}^{(n)}\left(h_i^{(n-1)}, m_i^{(n)}\right) \quad (7)$$

where message, aggregate and update denote the message-passing, aggregation, and update functionalities, respectively. $h_i^{(n-1)}$ is the hidden state of node i at the $(n-1)$-th layer, e_{ji} is the feature vector of the edge from node j to node i, and $N(i)$ defines the neighborhood set of node v. Further, a readout function is utilized to generate the graph-level representation according to:

$$h_G = \text{readout}\left(\left\{h_i^{(N)} | v_i \in G\right\}\right) \quad (8)$$

where N is the last iteration, and the readout function is a permutation-invariant function acting on the set of nodes.

After K iterations of the message-passing phase, the final updated set of atomic features $\left\{h_1^{(K)}, h_2^{(K)}, \ldots, h_N^{(K)}\right\}$ is used to compute the representation of the molecule:

$$h_G = \sum_{i=1}^{N} h_i^{(n)} \quad (9)$$

Additionally, the sets of BRICS and MacFrag fragments S_G and T_G are inputted into the GNN encoder. In this case, the encoders for the molecular graph and fragments utilize the same network structure and weights. According to $S_G = \{S_1, S_2, \ldots, S_T\}$ and $R_G = \{R_1, R_2, \ldots, R_P\}$, the representations of the BRICS fragments are acquired.

3.3. Multi-level mutual information fusion

Zhu et al. employed a GNN encoder to generate global and hierarchical representations by inputting the attribute graph of a molecule and its fragments [10]. Building on this approach, the present study utilizes the MacFrag algorithm to simultaneously input the molecular graph, fragments obtained from BRICS, and fragments obtained from MacFrag into the encoder structure. The model incorporates feature attention to learn information across three hierarchical levels and fuses them. Additionally, mutual attention mechanisms are employed to capture the intrinsic relationships between the molecule and its fragments. It should be noted that for the fragments obtained by the two algorithms, BRICS and MacFrag, we calculate the fragment similarity using MACCS fingerprints. Then, we remove the fragments with a Tanimoto coefficient >0.8 to exclude those with significantly repetitive chemical structures, thus avoiding information redundancy.

Based on the embeddings h_G and $S_G = \{s_1, s_2 \ldots s_T\}$ derived from the aforementioned GNN encoder, additive attention mechanisms are used to calculate the attention scores for the set of fragments as the first and second layers.

$$\alpha_t = \text{softmax}\left(\text{LeakyReLU}\left(a^T\left[W_1 h_G \parallel W_1 s_t\right]\right)\right) \quad (10)$$

where $W_1 \in R^{d \times d}$ and $a \in R^{2d \times 1}$. Then, the hierarchical information of BRICS fragments is compressed into $s_G = \sum_{t=1}^{T} \alpha_t s_t$.

Subsequently, $S_G = \{S_1, S_2, \ldots, S_T\}$ and $R_G = \{R_1, R_2, \ldots, R_P\}$ are considered as the second and third layers, respectively. Additive attention mechanisms are once again employed to compute the attention scores for the set of MacFrag fragments.

$$\beta_g = \text{softmax}\left(\text{LeakyReLU}\left(b^T\left[W_2 s_G \parallel W_2 r_p\right]\right)\right) \quad (11)$$

where $W_2 \in R^{d \times d}$ and $b \in R^{2d \times 1}$. Then, the hierarchical information of MacFrag fragments is compressed into $r_G = \sum_{p=1}^{G} \beta_p R_p$.

Specifically, multi-head attention is applied with four heads to comprehensively capture the relationships between the molecule and its fragments. Subsequently, h_G, S_G and r_G is concatenated for downstream prediction of molecular properties.

4. Experiments

4.1. Datasets

In order to evaluate the performance of the RFA-FFM model, this study selected ten widely used datasets related to drug discovery from MoleculeNet [18], which included seven classification tasks and three regression tasks. These benchmarks cover a range of molecular properties, including physiology, biophysics, physical chemistry, and quantum mechanics.

The BACE dataset was used as a classification task, consisting of 1513 molecules, and provided quantitative and qualitative combinatorial results for a set of inhibitors. The BBBP classification task included 2039 molecules and focused on the permeability of compounds through the blood-brain barrier. The HIV dataset was used to predict the activity of compounds inhibiting HIV replication and contained 41,127 compounds. ClinTox compared FDA-approved drugs with those eliminated due to toxicity in clinical trials and included 1478 compounds. The Tox21 dataset, a public database measuring compound toxicity, was used for the 2014 Tox21 data challenge. ToxCast, a compound toxicity prediction project, utilized in vitro high-throughput screening to identify potentially toxic compounds. The SIDER dataset was used for predicting drug side effects. The ESOL and FreeSolv datasets were regression tasks used to represent the hydration free energy of molecules. The Lipophilicity dataset was a regression task used to represent the lipophilicity of molecules.

For the fragment contrastive learning module, the model was pretrained on approximately five million unlabeled molecules from PubChem. Pretrained models frequently exhibit intricate architectures and a significant number of parameters, necessitating a substantial volume of data to facilitate effective model convergence. Utilizing a considerable proportion of training data allows the model to acquire more comprehensive feature representations. This provision of adequate information enables the updating of model parameters, facilitating the gradual adjustment of these parameters throughout the training process to minimize the loss function and align towards the optimal solution. Concurrently, a validation set comprising a smaller proportion can adequately represent the overall distribution of the data. This validation set serves the dual purpose of monitoring the model's convergence and identifying potential challenges, such as overfitting. Consequently, the pretraining dataset was randomly partitioned into training and validation sets at a 95/5 ratio. The GIN model was pretrained on the training set and evaluated on the validation set to select the best-performing model. During the pretraining process, we use the cross-entropy loss function weighted by the MACCS fingerprint similarity to evaluate the performance of the model on the validation set. The cross-entropy loss function, as an evaluation metric, is widely used in the pre-training performance of deep learning models. The lower the loss value, the better the model performance. The cross-entropy loss function is shown

in Equation (4) in Section 2.4. For the hierarchical attention module, the model employed a training-from-scratch strategy. All training experiments were conducted on a server equipped with an A6000 GPU.

To evaluate the performance of the RFA-FFM framework, the pre-trained GNN model was fine-tuned on ten benchmarks from MoleculeNet, including seven classification benchmarks and three regression benchmarks. These benchmarks cover a range of molecular properties, including physiology, biophysics, physical chemistry, and quantum mechanics. During the fine-tuning phase, the relatively small size of the benchmark datasets in MoleculeNet, with an average of only a few thousand molecular data entries, presents a challenge for establishing an appropriate training set size. A split ratio of 6/2/2 would yield an insufficient amount of data for the training set, potentially compromising the model's performance. In contrast, employing an 8/1/1 ratio enables the model to learn a greater number of features and patterns, particularly when data availability is constrained. Consequently, each dataset was partitioned into training, validation, and test sets at an 8/1/1 ratio to facilitate enhanced learning of features and patterns under limited data conditions. Random splits were applied to all datasets. To address the issue of uneven data distribution resulting from a single random split, we have employed 10 random seeds, ranging from value 2015 to value 2024 in prior experiments, for partitioning. Subsequently, we calculated the average value to improve the reliability of the partitioned data. A random seed is a value that is utilized to initiate a random number generator. For instance, the value 2015 signifies one of the possible outcomes of a random splitting process, while the value 2024 represents a different outcome.

4.2. Molecular property prediction

Molecular self-supervised learning (SSL) methods are typically evaluated based on their ability to predict various molecular properties. Eleven baseline models were selected for comparison.

Duvenaud et al. utilized Graph Convolutional Networks (GCN) to encode molecular graphs using atomic features. The fundamental concept of graph convolution algorithms involves updating the information of target atoms by aggregating information from neighboring atoms to obtain atomic representations. This is followed by a READOUT function to acquire molecule-level representations. Alternatively, a node connected to all other atoms in the molecular graph can be considered, and its representation can be taken as the molecular representation [19].

Message Passing Neural Networks (MPNN) incorporate information about atoms, chemical bonds, and graph structures in drug representation. Unlike traditional graph convolution networks, MPNNs consider both adjacent nodes and the connecting edges when updating node features. As a result, the features generated by MPNNs can more comprehensively represent the two-dimensional structural information of drugs [20].

Yang et al. proposed the Dynamic Message Passing Neural Network (DMPNN) model, which performs edge-based message passing processes on directed graphs. This model simultaneously obtains vector representations for both atoms and bonds, allowing for a more comprehensive understanding of the molecular structure [21].

Knowledge-centric Learning (KCL) attempts to use pretraining with knowledge graphs constructed from the two-dimensional graphs of molecules, enhancing the model's representational capabilities by weighting atomic nodes in molecular graphs [22].

3D Infomax utilizes the three-dimensional structure of molecules for pretraining, enabling the model to incorporate 3D structural information, which is crucial for improving performance in handling molecules [23].

The Uni-mol model transcends traditional one-dimensional sequences or two-dimensional graph structures by directly utilizing the three-dimensional structure of molecules as input and output, training the model with molecular 3D information [24].

The Geometric-aware Message Passing Network (GEM) introduces multiple geometric-level self-supervised learning tasks to learn the three-dimensional spatial knowledge of molecules [25].

Table 1 presents the computed area under the receiver operating characteristic curve (ROC-AUC) of RFA-FFM on classification benchmarks. These models were re-evaluated using laboratory equipment in the same manner. All datasets were randomly divided, and the best-performing models are highlighted in bold, while the second-best results are underlined. Additionally, the root mean square error (RMSE) performance of the model and baselines was assessed on regression benchmarks, as displayed in Table 2. RFA-FFM achieved the highest performance on four classification tasks (BBBP, Tox21, SIDER, and HIV) and the second-highest on BACE, ClinTox and ToxCast. For regression tasks, it achieved the highest performance on FreeSolv and the second-highest values on ESOL and Lipophilicity. Specifically, when compared to iMolCLR, RFA-FFM demonstrated improvements on five classification tasks and three regression tasks. Compared to HiGNN, RFA-FFM showed improvements on six classification tasks and one regression task.

To prevent overfitting on the datasets, ten-fold cross-validation was conducted on RFA-FFM across ten datasets. The results are visualized as line charts, contrasting with bar charts for regular validation, as depicted in Fig. 5.

4.3. Ablation study

To assess the influence of each component of the input on the outcomes, ablation studies were conducted from three perspectives: module ablation, ablation of two graph augmentation methods, and ablation of two fragment cutting algorithms. The findings of each ablation experiment are elaborated below.

(a) **Module Ablation.** The RFA-FFM comprises two main modules: the fragment contrastive learning module and the hierarchical attention module. In this study, the performance of the benchmark datasets was individually tested for each module. The evaluation metrics of each module tested on the benchmark datasets are presented in the last two rows of Tables 1 and 2 It can be observed that, compared to removing the hierarchical attention module, using only the fragment contrastive learning module improves performance by approximately 1 %–7 % across all classification tasks. Furthermore, compared to using only the hierarchical attention module, the performance improvement is approximately 2 %–7 % across all classification tasks. This indicates that RFA-FFM has the ability to learn fragment-level information from different perspectives, and the two modules complement each other.

(b) **Ablation of Graph Augmentation Methods.** The fragment contrastive learning module utilizes two graph augmentation methods for pretraining: masking a portion of the atoms in the molecule and masking partial features of the atoms. The masking ratio for both methods is 25 %. Evaluations were conducted on the seven classification datasets using only one of the augmentation methods at a time, and the results are presented in Table 3. The abbreviation 'w/o atom-mask' denotes the exclusive utilization of feature masking, whereas 'w/o attr-mask' denotes the exclusive utilization of atom masking. It can be observed that, overall, the performance of the fragment contrastive learning module is superior to the results of the single masking methods. Compared to using only feature masking, there is an improvement of approximately 1 %–4 %, and compared to using only atom masking, there is an improvement of approximately 2 %–6 %. This indicates that feature masking has a greater impact on the model. This suggests that the two masking methods provide complementary information, demonstrating the effectiveness of combining multiple graph augmentation methods.

(c) **Ablation of Fragmentation Algorithm.** To obtain molecular representations at the fragment level, the fragment contrastive

Table 1
Comparison results of RFA-FFM with baselines on classification tasks.

Dataset	BACE	BBBP	ClinTox	Tox21	ToxCast	SIDER	HIV
Molecule Numbers	1513	2039	1478	7831	8575	1427	41127
GCN [19]	0.854	0.877	0.807	0.772	0.65	0.638	0.74
MPNN [20]	0.815	0.913	0.815	0.741	0.691	0.621	0.771
D-MPNN [21]	0.852	0.919	0.852	0.821	0.718	0.632	0.770
MolCLR [15]	0.850	0.724	0.880	0.784	0.691	0.597	0.778
KCL [22]	**0.924**	0.956	0.898	0.821	0.714	0.671	0.770
3D infomax [23]	0.794	0.691	0.594	0.745	0.644	0.534	0.761
uni-mol[24]	0.857	0.729	0.919	0.791	0.696	0.655	0.808
GEM [25]	0.856	0.724	0.901	0.781	0.692	0.672	0.806
HiGNN [11]	0.890	0.932	0.930	0.856	**0.781**	0.651	0.816
iMolCLR [13]	0.885	0.764	**0.954**	0.799	0.736	0.699	0.808
RFA-FFM (Ours)	0.901	**0.959**	0.947	**0.859**	0.743	**0.725**	**0.835**
Frags-contrast module	0.892	0.943	0.935	0.841	0.739	0.693	0.801
Multi-layers attention module	0.839	0.935	0.942	0.812	0.735	0.683	0.797

Table 2
Comparison results of RFA-FFM with baselines on regression tasks.

Dataset	ESOL	FreeSolv	Lipophilicity
Molecule Numbers	1128	642	4200
GCN	1.431	2.900	0.712
MPNN	1.167	2.185	0.852
D-MPNN	1.050	2.177	0.672
MolCLR	0.911	2.021	0.875
KCL	0.670	0.854	0.789
3D infomax	0.798	1.855	0.880
uni-mol	0.788	1.480	0.603
GEM	0.798	1.877	0.660
HiGNN	**0.532**	0.915	**0.549**
iMolCLR	1.130	2.090	0.640
RFA-FFM (Ours)	0.648	**0.574**	0.572
Frags-contrast module	0.941	1.082	0.624
Multi-layers attention module	0.691	0.967	0.640

learning module utilizes two fragment cutting algorithms: BRICS and RECAP. The performance was evaluated using only one of these algorithms and compared with the overall module that incorporates both methods. The results are presented in only the RECAP algorithm, while "w/o RECAP" indicates the use of only the BRICS algorithm. The results demonstrate that the module combining both algorithms outperforms the use of only BRICS by approximately 1 %–4 %, and it outperforms the use of only RECAP by approximately 2 %–5 %. This suggests that employing both cutting algorithms provides complementary information for the model to learn fragment-level details. Furthermore, since BRICS generates finer fragments compared to RECAP, it can be inferred that using only the BRICS algorithm leads to better fragment-level contrastive learning, which is supported by the experimental findings.

4.4. Visualization

Fig. 5 presents the visualization analysis of the BBBP dataset after undergoing pretraining and fine-tuning. The distribution is depicted using a three-dimensional scatter plot with t-distributed stochastic neighbor embedding (t-SNE). Fig. 6(a) shows that before pretraining and fine-tuning, there is minimal clustering observed between penetrable and non-penetrable compounds. Fig. 6(b) demonstrates that after fine-tuning, the BBBP dataset achieves a significant separation and successful clustering of penetrable and non-penetrable compounds.

This finding indicates that the clustering effect on the BBBP dataset was significantly poor before pretraining. Furthermore, it demonstrates that the RFA-FFM model, after undergoing pretraining and fine-tuning on fragment-level molecular representations, can effectively predict

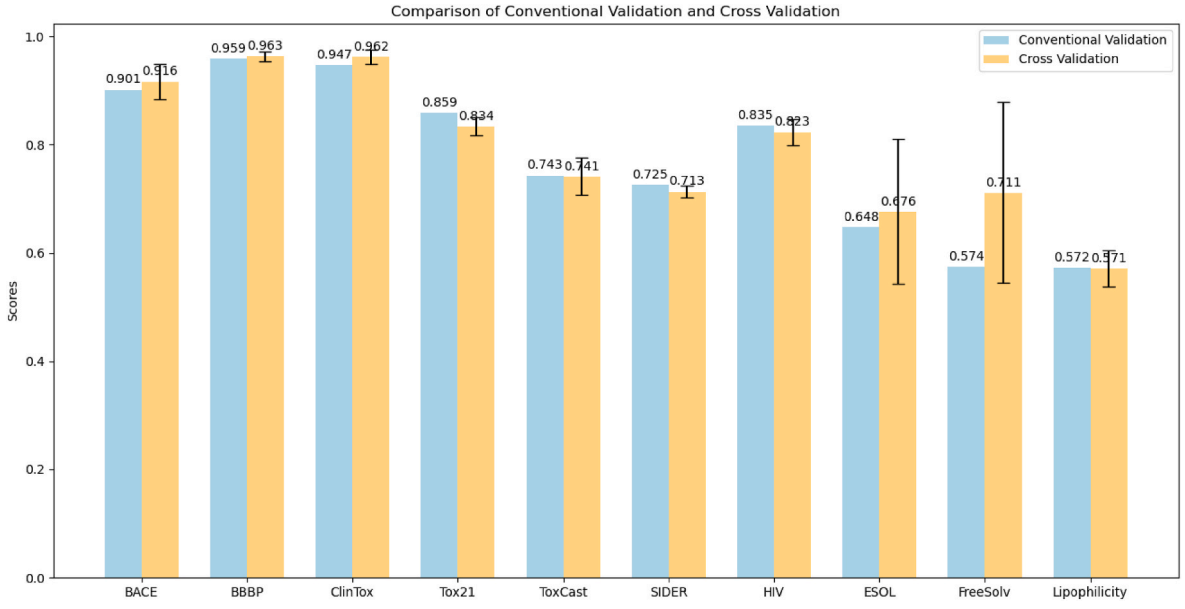

Fig. 5. 10-Fold cross validation and Conventional validation results of RFA-FFM

Table 3
Ablation results of the fragment contrastive learning module on classification tasks.

Dataset	BACE	BBBP	ClinTox	Tox21	ToxCast	SIDER	HIV
Molecule Numbers	1513	2039	1478	7831	8575	1427	41127
Frags-contrast module	**0.892**	**0.943**	**0.935**	**0.841**	**0.739**	**0.693**	**0.801**
w/o attr-mask	0.843	0.891	0.894	0.831	0.734	0.676	0.774
w/o atom-mask	0.864	0.925	0.889	0.837	0.732	0.680	0.780
w/o BRICS	0.841	0.921	0.851	0.836	0.692	0.676	0.773
w/o RECAP	0.850	0.926	0.874	0.832	0.697	0.669	0.781

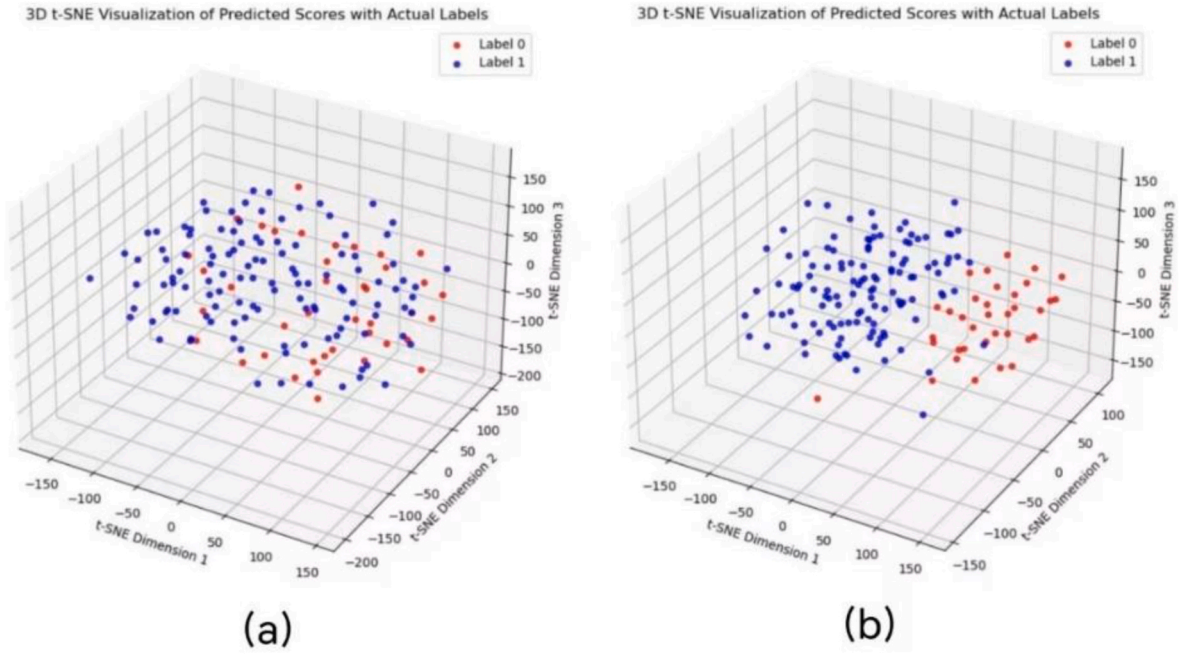

Fig. 6. t-SNE Visualization of the BBBP Dataset.

molecular properties.

4.5. Effectiveness of the retrosynthetic fragmentation algorithm

The BRICS and RECAP algorithms utilized in this study are both retrosynthetic fragmentation algorithms. In order to assess the efficacy of these algorithms, they were compared with the linguistically-based BPE fragmentation algorithm. The benzene ring serves as a fundamental component in numerous drug molecules, and its hydrophobic nature aids in the penetration of molecules through the blood-brain barrier by increasing their lipophilicity. In the study, an example molecule was selected from the BBBP dataset, with its SMILES representation as N(C)C(=O)Cc1n2cc(C)ccc2nc-1c3ccc(C)cc3, and its structure is depicted in Fig. 7(A). This molecule exhibits positive blood-brain barrier penetration, indicating its ability to traverse the barrier. When this molecule is fragmented using the BRICS and RECAP algorithms, both algorithms yield the same fragment with the SMILES representation [16*]c1ccc(C)cc1, as illustrated in Fig. 7(B), where the benzene ring structure is c1ccc(C)cc1. Both the BRICS and RECAP algorithms preserve this structure intact, thereby facilitating the molecule's blood-brain barrier penetration.

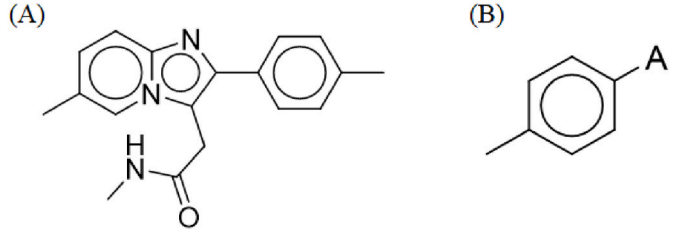

Fig. 7. Example molecule. (A) Structure diagram of the example molecule. (B) Structure diagram of the fragment obtained using the BRICS algorithm.

Conversely, when the BPE algorithm is employed to segment the SMILES string, it results in a list of distinct fragments: ['(', '1′, 'C', 'cc', '2′, 'c', '3′, 'O', 'n', '(', 'C', '(', ' = ']. It can be observed that this algorithm completely separates the benzene ring structure, which would hinder the molecule's blood-brain barrier penetration and could potentially lead to a failure in penetration, thereby impacting the predictive performance of the BBBP dataset. Through this example, it can be inferred that the utilization of the BRICS and RECAP algorithms effectively enhances the performance of molecular property prediction tasks such as BBBP. Thus, the selection of retrosynthetic fragmentation algorithms in this study effectively improves the prediction performance.

4.6. Comparison experiment with CC-single

To further investigate the advantages of the two fragmentation algorithms, BRICS and RECAP, the CC-Single fragmentation algorithm [26] was utilized. This algorithm was combined with the aforementioned two algorithms to generate molecular fragment libraries. Three combinations were formed: BRICS& CC-Single, RECAP & CC-Single, and BRICS & RECAP. These three combinations were employed to create three fragment libraries, which were subsequently used to pretrain the Frag-Contrast module. Each pretraining process utilized the same molecular SMILES library consisting of 500k molecules. The experimental

workflow is depicted in Fig. 8, and the testing results of the three combinations are presented in Table 4. It can be observed that the BRICS & RECAP combination yielded the most favorable results, exhibiting an improvement in performance ranging from approximately 2.4 %–4.2 % compared to the other two combinations. This is attributed to the fact that both algorithms are based on chemical reaction rules, but their fragmentation rules differ, thereby complementing each other. Furthermore, it was discovered that the RECAP & CC-Single combination performed relatively poorer. This could be attributed to the fact that RECAP has the fewest fragmentation rules among the three algorithms, making it challenging to learn more detailed features. However, combining RECAP with BRICS resulted in a superior outcome.

Table 4
Comparison of fragment cutting algorithm combinations on the BBBP task.

Fragmentation Algorithm Combinations	ROC-AUC (BBBP)
BRICS & CC-Single	0.882
RECAP & CC-Single	0.864
BRICS & RECAP (ours)	**0.906**

4.7. Case study

Hepatitis B virus (HBV) is the pathogen responsible for hepatitis B, which is a current global public health issue. Traditional screening of HBV drugs is expensive, but drugs can be further screened based on their cytotoxicity rate against normal liver cells. For this study, we selected 2271 compounds with hepatotoxicity (CC50) from the CHEMBL database as training data. The ChEMBL database is an open-access database manually curated by the European Bioinformatics Institute (EMBL-EBI). It provides data on bioactive molecules with drug - like properties. ChEMBL contains data on over 2.39 million compounds, sourced from a wide range of literature, including numerous scientific publications, ensuring the diversity and scientific nature of the data sources. Through meticulous manual curation, the accuracy and consistency of the data are guaranteed. In addition, it is integrated with other large-scale chemical resources such as PubChem and ChemSpider systems. Cross-validation can be carried out to further enhance the reliability of the data [27].

A compound is considered toxic if its CC50 is less than 30 μM, and non-toxic if it is greater than or equal to 30 μM. The RFA-FFM model was trained on the HBV dataset to determine its feasibility for the specific application of screening drugs for hepatitis B. Table 5 presents the comparison results of RFA-FFM and its two modules on the HBV dataset against baseline models. It can be observed that the fragment contrastive learning module achieved suboptimal performance, while the integrated RFA-FFM model achieved optimal performance. Specifically, compared with the GIN model without fragment-level pretraining, RFA-FFM improved the performance by approximately 6 %–7 %, indicating that fragment-level pretraining significantly enhances the prediction performance for hepatotoxicity.

Table 5
Comparison of classification models on the HBV dataset using RFA-FFM

HBV Dataset	CC50
Compounds Numbers	2271
Transformer	0.695
MPNN	0.716
GIN	0.738
Frags-contrast module	0.791
Multi-layers attention module	0.782
RFA-FFM	**0.804**

5. Conclusions

The RFA-FFM model is divided into two specific modules. Firstly, during contrastive learning pretraining, RFA-FFM introduces finer-

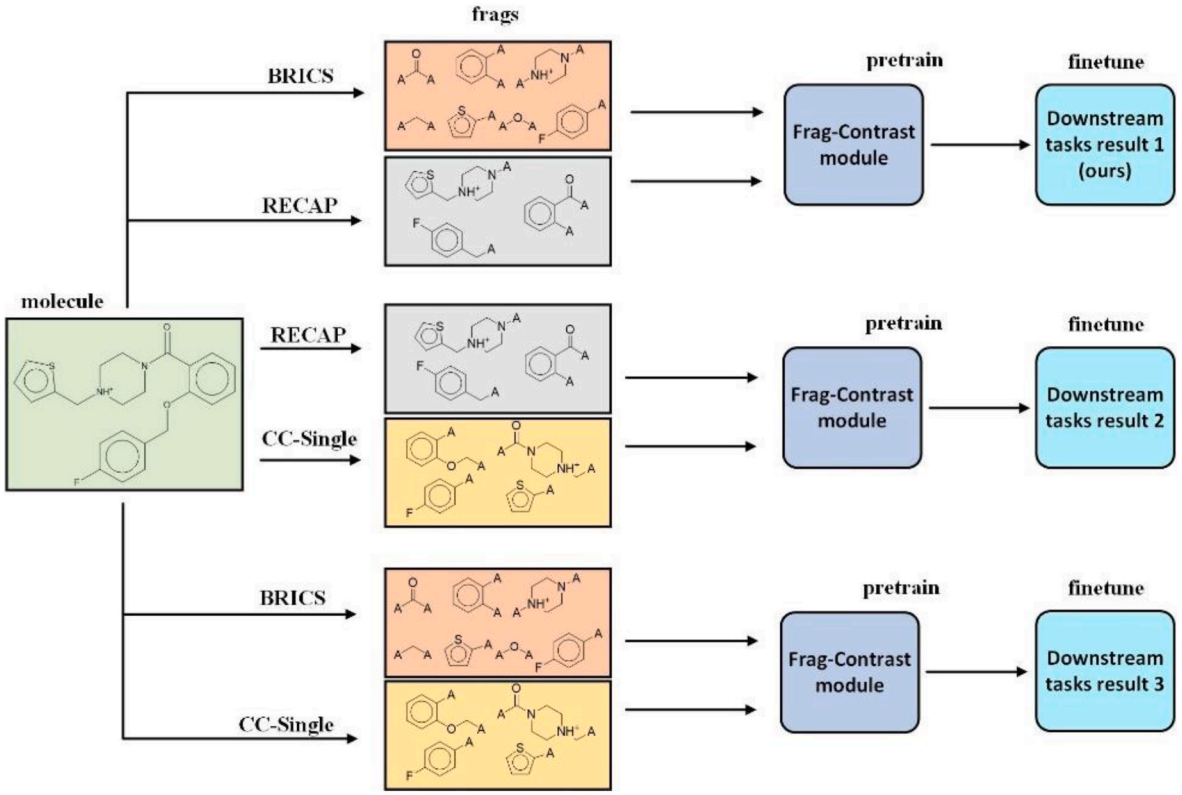

Fig. 8. Comparative experiment workflow.

grained representations by performing fragment-level contrastive learning on the substructures of molecules using decomposition algorithms. In addition, it incorporates two decomposition algorithms, BRICS and RECAP, which are both based on the same logic and utilize rules related to retrosynthetic fragmentation. Secondly, RFA-FFM introduces a third finer-grained decomposition algorithm MacFrag. The fragments obtained from the three algorithms are then subjected to a cross-attention mechanism to study the relationships between fragments generated by different decomposition algorithms. With the help of these two modules, the integrated model RFA-FFM outperforms other SSL baseline models on various molecular property prediction benchmarks, achieving improvements of approximately 1 %–3 % across four classification tasks. Through comparisons with new decomposition algorithms and analysis of the specific case of HBV, these studies demonstrate that RFA-FFM is an effective and robust SSL framework that learns representations from limited input features, promising accurate molecular property predictions. This will greatly benefit applications such as drug and material discovery. Furthermore, our next step is to investigate the robustness of RFA-FFM in the field of retrosynthesis to enhance its generalization capabilities.

CRediT authorship contribution statement

Qifeng Jia: Writing – original draft. **Yekang Zhang:** Visualization. **Yihan Wang:** Resources. **Tiantian Ruan:** Software. **Min Yao:** Supervision. **Li Wang:** Project administration.

Data availability

The code, pretraining data and molecular property prediction benchmarks used in this work is available in the GitHub repository at https://github.com/NTU-MedAI/RFA-FFM.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Wang Li reports financial support was provided by National Natural Science Foundation of China. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The work was supported by the National Natural Science Foundation of China (No. 32470985) and Foreign Youth Talent Program of the Ministry of Science and Technology, China (No. QN2022014011L). We thank our partners who provided all the help during the research process and the team for their great support.

Data availability

Data will be made available on request.

References

[1] Xiang Wu, Yuyang Wei, Yaqing Mao, Liangbi Wang, Cluster Computing, 2019, pp. 1–13.
[2] Marcus Olivecrona, Thomas Blaschke, Ola Engkvist, Hongming Chen, J. Cheminf. 9 (2017).
[3] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, Philip S. Yu, IEEE Transact. Neural Networks Learn. Syst. (2019) 4–24.
[4] Tanjim Taharat Aurpa, Richita Khandakar Rifat, Md Shoaib Ahmed, Md Musfique Anwar, A.B.M. Shawkat Ali, Helyon 8 (2022).
[5] Kevin Yang, Kyle Swanson, Wengong Jin, Connor W. Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian P. Kelley, Miriam Mathea, Andrew Palmer, Volker Settels, T. Jaakkola, Klavs F. Jensen, Regina Barzilay, J. Chem. Inf. Model. (2019) 3370–3388.
[6] Yuansong Zeng, Zhuoyi Wei, Weijiang Yu, Rui Yin, Bingling Li, Zhonghui Tang, Yutong Lu, Yuedong Yang, Briefings in Bioinformatics, 2022.
[7] Ying Song, Shuangjia Zheng, Zhangming Niu, Zhang-Hua Fu, Yutong Lu, Yuedong Yang, International Joint Conference on Artificial Intelligence, 2020.
[8] Jianwen Chen, Shuangjia Zheng, Ying Song, Jiahua Rao, Yuedong Yang, Proceedings of theThirtieth International Joint Conference on Artificial Intelligence, 2021, pp. 2242–2248.
[9] S. Vangala, S.R. Krishnan, N. Bung, R. Srinivasan, A. Roy, Journal of Chemical Information and Modeling, 2023.
[10] Eugen Lounkine, José Batista, Jürgen Bajorath, Curr. Med. Chem. (2008) 2108–2121.
[11] Weimin Zhu, Yi Zhang, Duancheng Zhao, Jianrong Xu, Ling Wang, Journal of Chemical Information and Modelling, 2022.
[12] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, Y. Shen, Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS '20), 2020, pp. 5812–5823.
[13] Yuyang Wang, Rishikesh Magar, Chen Liang, Amir Barati Farimani, J. Chem. Inf. Model. (2022).
[14] Hakime Öztürk, Elif Ozkirimli Olmez and Arzucan Özgür, Bioinformatics (2018) i821–i829.
[15] Yuyang Wang, Jianren Wang, Zhonglin Cao, Amir Barati Farimani, Nat. Mach. Intell. (2021) 279–287.
[16] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay S. Pande, Jure Leskovec, International Conference on Learning Representations (2019).
[17] Yanyan Diao, Feng Hu, Zihao Shen, Honglin Li, Bioinformatics 39 (2023).
[18] Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, Vijay S. Pande, Chem. Sci. (2017) 513–530.
[19] Mahsa Ghorbani, Mahdieh Soleymani Baghshah, Hamid R. Rabiee, IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2019, pp. 208–211.
[20] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals and George E. Dahl, International Conference on Machine Learning, 2017.
[21] Kevin Yang, Kyle Swanson, Wengong Jin, Connor W. Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian P. Kelley, Miriam Mathea, Andrew Palmer, Volker Settels, T. Jaakkola, Klavs F. Jensen, Regina Barzilay, J. Chem. Inf. Model. (2019) 3370–3388.
[22] Yin Fang, Qiang Zhang, Haihong Yang, Xiang Zhuang, Shumin Deng, Zhang Wen, Minghai Qin, Zhuo Chen, Xiaohui Fan, Huajun Chen, AAAI Conference on Artificial Intelligence, 2021.
[23] Hannes Stärk, D. Beaini, Gabriele Corso, Prudencio Tossou, Christian Dallago, Stephan Gunnemann and Pietro Lio', International Conference on Machine Learning (2021).
[24] Gengmo Zhou, Zhifeng Gao, Qiankun Ding, Hang Zheng, Hongteng Xu, Zhewei Wei, Linfeng Zhang, Guolin Ke, International Conference on Learning Representation (2023).
[25] Xiaomin Fang, Lihang Liu, Jieqiong Lei, Donglong He, Shanzhuo Zhang, Jingbo Zhou, Fan Wang, Hua Wu, Haifeng Wang, Nat. Mach. Intell. (2021) 127–134.
[26] Ailin Xie, Ziqiao Zhang, Jihong Guan, Shuigeng Zhou, Briefings Bioinf. (2023).
[27] Anna Gaulton, Anne Hersey, Michal Nowotka, A. Patrícia Bento, Jon Chambers, David Mendez, Prudence Mutowo-Meullenet, Francis Atkinson, Louisa J. Bellis, Elena Cibrián-Uhalte, Mark Davies, Nathan Dedman, Anneli Karlsson, María P. Magariños, John P. Overington, Papadatos George, Ines Smit, Andrew R. Leach, Nucleic Acids Res. (2016) D945–D954.