SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

May 2026, Vol. 69, Iss. 5, 150102:1–150102:15
https://doi.org/10.1007/s11432-025-4913-5

Special Topic: Large Multimodal Models

Gelm: graph-based Tanimoto similarity grouping pretraining and entropy-guided conformer selection finetuning for large language models

Zhuo CHEN[1,2], Sihan WANG[1,2], Linjiang CHEN[3], Wenjie DU[1,2*] & Yang WANG[1,2*]

[1]*School of Software Engineering, University of Science and Technology of China, Hefei 230026, China*
[2]*Suzhou Institute for Advanced Research, University of Science and Technology of China, Suzhou 215123, China*
[3]*State Key Laboratory of Precision and Intelligent Chemistry, University of Science and Technology of China, Hefei 230026, China*

Received 31 March 2025/Revised 12 September 2025/Accepted 23 March 2026/Published online 20 April 2026

Abstract Molecular relationship learning (MRL) aims to understand interactions between molecular pairs, driving advancements in biochemical research. In recent years, large language models (LLMs), with their vast knowledge base and reasoning capabilities, have become important tools for MRL. However, existing LLMs primarily rely on SMILES strings and molecular graph representations, facing three major challenges: a lack of relational awareness, making it difficult to associate molecules with similar structures; overlooking the structural diversity of molecules, preventing the capture of key conformers in real-world reactions and the lack of a systematic evaluation of different LLM backbone models. To address these challenges, we propose Gelm (graph-based Tanimoto similarity grouping pretraining and entropy-guided conformer selection finetuning for large language models), a novel framework that enhances relationship learning through structure similarity-based pretraining and entropy-guided conformer selection. Additionally, we conduct extensive performance evaluations on various backbone models to provide scientific guidance on backbone selection. Our results demonstrate that Gelm, with DeepSeek as the backbone, achieves outstanding performance across 12 cross-domain datasets.

Keywords molecular relationship learning, multimodal large language model, pretraining, information entropy, DDI prediction

Citation Chen Z, Wang S H, Chen L J, et al. Gelm: graph-based Tanimoto similarity grouping pretraining and entropy-guided conformer selection finetuning for large language models. Sci China Inf Sci, 2026, 69(5): 150102, https://doi.org/10.1007/s11432-025-4913-5

1 Introduction

Molecular relationship learning (MRL) [1] aims to understand interactions between molecular pairs, which is crucial for drug discovery and materials science [2]. For instance, drug-drug interactions (DDIs) are essential in pharmacology and drug development, while solute-solvent interactions (SSIs) are fundamental in solution chemistry and chemical process design [3]. However, experimentally validating these interactions is both time-consuming and expensive [4]. With the advancement of large language models (LLMs), a promising alternative approach has emerged for MRL by leveraging the capabilities of LLMs. Models such as ReactionT5 [5] and MolTC [6] have demonstrated success in this area by integrating molecular graphs and SMILES.

Although these models have great potential, there are still some issues to be resolved. When LLMs are transferred to MRL tasks, their performance in MRL tasks is often limited if incremental pretraining is not performed, as they are initially trained on general-purpose corpora. Although ReactionT5 [5] and MolTC [6] have adopted appropriate incremental pretraining strategies to enhance their performance in MRL tasks, they have not fully considered the associative learning capabilities of LLMs during pretraining. As shown in Figure 1, when two structurally similar molecules are input in distant sequences, the LLM may treat them as independent samples and fail to link the information between these two molecules for joint learning, which is crucial for molecular category learning. In the text-based task field, similar associative pretraining methods have been proposed, such as in-context pretraining proposed by Shi et al. [7]. However, in the MRL field, there is still a gap in related research on LLM pretraining methods.

Moreover, existing LLMs for predicting molecular interactions primarily rely on 1D and 2D representations, such as SMILES strings and molecular graphs [5, 6]. While these representations can capture certain basic features of

* Corresponding author (email: duwenjie@ustc.edu.cn, angyan@ustc.edu.cn)

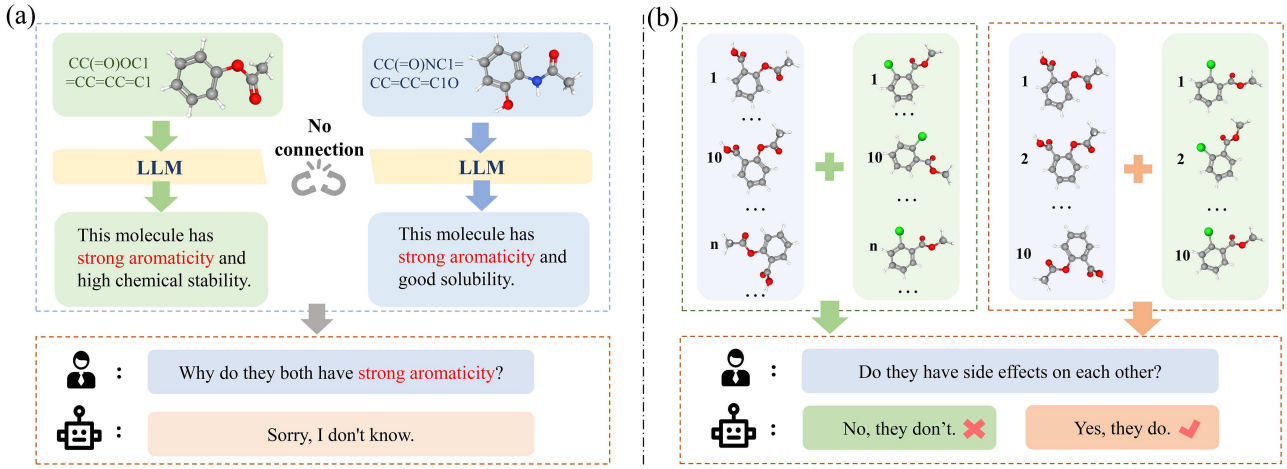

Figure 1 (Color online) The issues with current LLMs. (a) indicates that chemical LLMs lack associative thinking during pretraining. The phrase "Sorry, I don't know." is not necessarily the actual response of the LLM; in reality, the LLM may provide incorrect knowledge. (b) indicates that providing too many molecular conformations as input to the LLM can affect its judgment.

molecules, they have significant limitations in reflecting the true structure of molecules [8]. In reality, molecules have many different 3D conformations, and during actual reactions, it is usually specific 3D conformations that participate, such as in the binding of small molecules to target proteins [9]. Capturing this information can only be achieved through the 3D conformation of the molecule. Therefore, it is crucial to provide 3D conformation data to LLMs. However, current LLMs cannot effectively incorporate all possible 3D conformations into model training. A molecule can have hundreds or thousands of different 3D conformations, but not all of them contribute significantly to a reaction. As shown in Figure 1, for LLMs, only a small fraction of the conformations are closely related to the reaction, while the rest might introduce noise that interferes with the model's judgment and accuracy. Hence, effectively identifying and selecting key conformations relevant to the reaction and providing them to LLMs for better MRL performance is an important challenge.

Considering the above issues, we propose the model framework of graph-based Tanimoto similarity grouping pretraining and entropy-guided conformer selection finetuning for large language models, termed Gelm. Specifically, we design the graph-based Tanimoto similarity grouping pretraining strategy (GTSG), which enhances the LLM's associative recognition by grouping inputs based on a molecular structure similarity graph. The entropy-guided conformer selection finetuning (EGCS) method is used to identify the most crucial conformations by utilizing graph-based information entropy. Additionally, we systematically evaluate the performance differences brought by using LLMs with different architectures as backbones, providing guidance for selecting the appropriate backbone, thus addressing the issue of the lack of backbone selection guidance in frameworks like ReactionT5 and MolTC.

The main contributions of this paper could be summarized as follows.

- Considering the lack of associative recognition in existing chemical LLMs, we propose the GTSG pretraining strategy and introduce a unique prompt to help the LLM develop associative recognition of molecular structures.
- In the downstream task fine-tuning phase, we propose the EGCS method to identify conformations with critical information for molecular interactions, enabling comprehensive capture of molecular details.
- We conduct systematic testing of different LLMs on MRL and validate DeepSeek's [10] superiority in the MRL task. Our results demonstrate that DeepSeek [10] exhibits superiority in both qualitative and quantitative tasks.
- The effectiveness of Gelm is demonstrated through experiments on 12 datasets from diverse domains, including DDI, compound-solvent interaction (CSI), and SSI tasks, highlighting its superior performance compared to existing methods.

2 Related work

2.1 LLM pretraining

LLM pretraining is essential as it allows the model to learn general knowledge and language patterns, improving efficiency and accuracy for specific tasks. However, different data processing methods during pretraining can affect the performance of LLMs on downstream tasks. Gao et al. [11] trained variants of GPT-2 [12] models from scratch to compare the "Pile" dataset to CommonCrawl-derived corpora. Hernandez et al. [13] quantified the effect of

various amounts of artificially created data duplication and provided an analysis on interpreting the changes in the behavior of the models trained on duplicated data. Xie et al. [14] proposed using importance resampling to align the distribution of web data to high-quality reference corpora such as Wikipedia. Similarly, Gururangan et al. [15] explored data selection strategies for adapting LMs to a task-specific corpus. Another line of recent work explores how data mixture affects pretraining, with Xie et al. [16] demonstrating impressive improvements in downstream accuracy and perplexity across all datasets for 8B parameter models trained on the Pile. Similarly, Longpre et al. [17] explored the role of text quality, toxicity, age, and domain distribution of training data on LLM performance. Outside of data curation, there has been a recent surge of work exploring the impact of repeating data [18–20], generally concluding that repeating tokens is worse than training on new tokens. To address the impact of repeated tokens, Shi et al. [7] proposed in-context pretraining, which also enhances the connections between pretraining texts. However, these methods and data are limited to the text modality, and pretraining methods for multimodal LLMs still need improvement, which is the issue this paper aims to address.

2.2 LLMs in the molecular domain

To address the challenge of label insufficiency caused by costly laboratory experiments and to better utilize textual knowledge for molecular tasks, recent studies have introduced multimodal alignment into molecular representation learning. Existing approaches can be categorized into three types based on molecular modalities: 1D, 2D, and 3D [21,22]. For 1D modalities, such as SMILES or SELFIES, molecules are represented as text strings. For instance, MolT5 [23] pretrains a model on large-scale unlabeled natural language text and molecular strings using a simple denoising objective, while KV-PLM [24] employs the byte pair encoding (BPE) algorithm to segment SMILES representations into frequent substring patterns in a data-driven manner. 2D modalities represent molecules as graphs, where atoms are nodes and bonds are edges. Notable approaches, such as Text2Mol [25], use cross-modal contrastive learning, training a molecular graph encoder with GCNs [26] and a text encoder with Sci-BERT [27]. MolCA [8] bridges the representation spaces of graph encoders and language models through a QFormer-based cross-modal projector, enabling a deeper understanding of both text-based and graph-based molecular content. DrugChat [28] combines a GNN for encoding molecular graphs, an LLM for text understanding, and an adaptor to convert graph representations into LLM-compatible inputs. For 3D modalities, MolLM [29] proposes a unified pretraining framework that incorporates both 2D and 3D structural information through attention mechanisms, leveraging molecular graphs, edge features, and spatial relationships. Similarly, 3D-MoLM [30] enhances a language model with a 3D molecular encoder to interpret and analyze 3D molecular structures. Despite these advancements, these studies primarily focus on individual molecules, and research on LLMs in the context of molecular interactions remains limited. Our work aims to advance this field by addressing this gap.

2.3 Molecular relationship learning

MRL is an influential research area, especially in DDI prediction, which has a direct impact on both production and daily life. Due to the time-consuming and expensive nature of experimentally validating molecular interactions [31], machine learning-based methods have emerged as an efficient and effective alternative. Early research primarily focused on graph neural networks (GNNs) to construct predictive frameworks for molecular relationship learning [26, 32–34]. For example, Zhong et al. [35] proposed a substructure-substructure interaction framework, which utilizes graph attention network (GAT) layers for substructure extraction and a co-attention layer to model interactions between different substructures. To better capture molecular interactions, Lee et al. [31] introduced the conditional graph information bottleneck (CGIB) model, inspired by information bottleneck theory. This model aims to identify the core substructures between graph pairs and predict their interaction behaviors. With the rise of multimodal technologies, some studies [36] have integrated chemical knowledge with natural language information to improve cross-modal integration in biological tasks, yielding significant results in drug-target interaction prediction. Meanwhile, MolTC [6] proposed a unified framework for molecular relationship learning, boosting model performance through a four-stage training process combined with chain-of-thought reasoning techniques. However, none of these approaches have considered the impact of molecular conformation on LLM reasoning.

3 Methodology

In this section, we will discuss the design of Gelm in detail. Gelm consists of two main stages. The first stage is pretraining, where we introduce GTSG. This approach groups similar molecules as input, enabling the LLM to learn categorical information more effectively. The second stage is fine-tuning, where we propose an innovative method

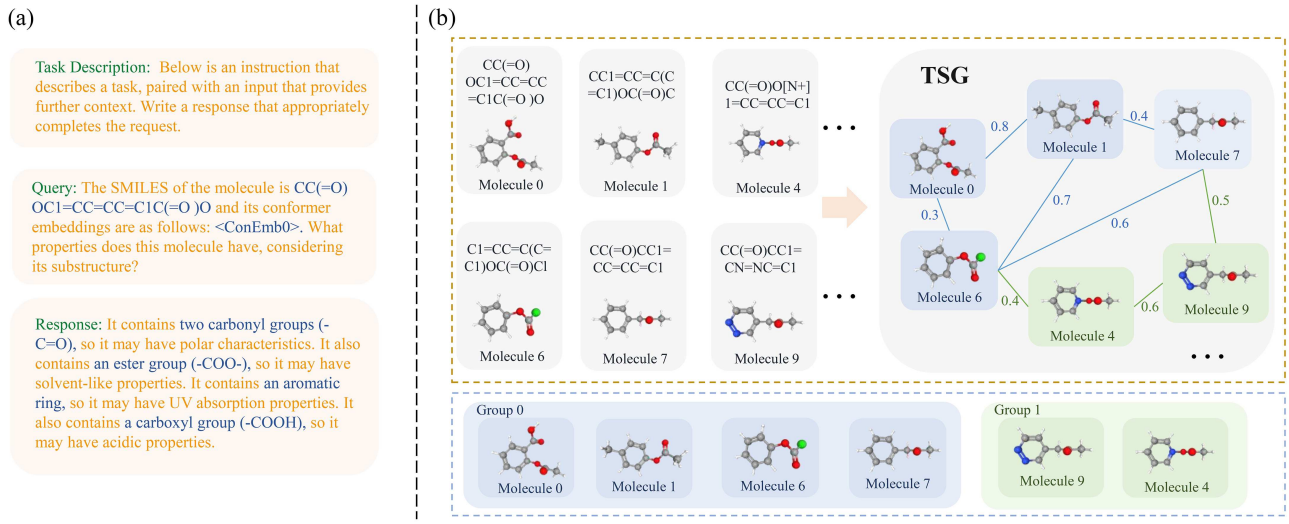

Figure 2 (Color online) (a) The prompt settings during pretraining, where the ten lowest-energy conformations are selected as inputs for multimodal learning; (b) how to construct pretraining inputs based on structural similarity, where the order of the groups can be shuffled, but the input order within each group is kept consistent to enhance the associative learning ability.

called EGCS. EGCS maximizes the mutual information between conformers and targets while simultaneously optimizing structural entropy and information entropy between selected conformers and the full conformer set. This helps identify the most critical conformers, allowing the LLM to make more accurate judgments.

3.1 Graph-based Tanimoto similarity grouping pretraining

The standard pretraining method for general chemical LLM is to randomly input molecular description texts into the model, allowing the LLM to perform understanding and generation. However, random input does not enhance the LLM's recognition of category knowledge. To address this, we conduct a survey of various authoritative biochemical databases such as PubChem[1)] and Drugbank [37], collecting a large amount of molecular property description texts based on molecular structure. Figure 2(a) is an example. We then design a pretraining method called graph-based Tanimoto similarity grouping pretraining, as shown in Figure 2(b), which inputs structurally similar molecules in groups to enhance the LLM's recognition of category knowledge and strengthen the LLM's understanding of the impact of molecular structure on molecular properties.

3.1.1 *Finding related molecules at scale: retrieving similar structures*

To identify structurally similar molecules within a large molecular corpus $\mathcal{M}$, we first use molecular fingerprints to encode the structure of each molecule. Then, we use approximate nearest neighbors search for efficient pairwise similarity comparison between any two molecules. Specifically, for each molecule $m_i \in \mathcal{M}$, our goal is to retrieve the top-k most similar molecules, represented as $N(m_i)$.

Retrieval. Our retrieval process uses Morgan fingerprints [38]. This fingerprint encodes each molecule $m_i \in \mathcal{M}$ by considering the substructures of each molecule through the calculation of the surrounding environment of different atoms within the molecule, and we denote the molecular representation obtained from the Morgan fingerprint as $\mathbf{E}(m_i)$. The Tanimoto similarity [39] is then used to determine the similarity between any two molecules

$$s(n_i, n_j) = \text{Tanimoto}(\mathbf{E}(m_i), \mathbf{E}(m_j)). \tag{1}$$

We use the FAISS library for approximate nearest neighbor search [40, 41], combined with big batch search to perform efficient pairwise similarity search. However, during the retrieval process, when calculating the pairwise similarity between each molecule, we often encounter the issue of the same molecule appearing multiple times. To address this, we group molecules based on similarity scores to prevent the repeated occurrence of a single molecule.

3.1.2 *Molecule grouping: traversing the Tanimoto similarity graph*

Given a set of molecules $\mathcal{M} = \{m_1, m_2, \ldots, m_i\}$ and the nearest neighbors for each molecule $N(m_i)$, our goal is to sort and group the molecules such that each group of pretraining data contains molecules with the most related

1) https://pubchem.ncbi.nlm.nih.gov.

substructures. Formally, we aim to form a set of inputs $\mathcal{S}_1 \cdots \mathcal{S}_m$, where each molecule has a corresponding textual description $\mathcal{S}_i = \{s_1, s_2, \ldots, s_k\}$ and $\bigcup_{i=1}^{m} \mathcal{S}_i = \mathcal{M}$. Ideally, the molecules in $\mathcal{S}_i$ are nearest neighbors of each other.

A straightforward approach to form $\mathcal{S}_1 \cdots \mathcal{S}_m$ is to directly place each molecule and its retrieved top-k molecules together as a group. Although this method maintains structural similarity within each group of molecules, it introduces the data repetition problem, where the same molecule groups appear multiple times. This data repetition problem exposes the LLM to a less diverse set of molecules given a fixed computational budget, potentially leading to overfitting to the common structures of the molecules. Therefore, our goal is to construct molecule groups in a way that each molecule appears only once, which can be cast as a graph traversal problem.

Tanimoto similarity graph traversal. To maximize the grouping of related molecules, an intuitive approach is to find a path that visits each molecule once and maximizes the probability that related molecules are visited sequentially. We then treat the path as a molecular group. We formulate this as a variant of the maximum traveling salesman problem [42], which aims to find multiple paths where each node appears exactly once across these paths, with each path being a maximum-weight path. We represent each molecule as a node in the graph and use Tanimoto similarity as the edge weight. We design a Tanimoto similarity graph (TSG), denoted as $\mathcal{G}_{\mathcal{T}} = (\mathcal{M}, \mathcal{E})$. Here, $\mathcal{G}_{\mathcal{T}}$ represents the set of molecules, while $(n, n^*) \in \mathcal{E}$ is an edge if $n^* \in N(n_i)$ or $n_i \in N(n^*)$. The weight of each edge corresponds to the Tanimoto similarity (Eq. (1)).

Solving large-scale traveling salesman problems exactly is NP-hard, but greedy algorithms are known to provide efficient approximate solutions. We adopt this approach with modifications to better suit our specific problem. Algorithm 1 presents the method for constructing maximum-weight paths, and the identified path is illustrated in Figure 2. Our algorithm begins by selecting the molecule with the smallest degree that has not yet been visited as the starting node (Molecule 0). It then progressively extends the current path by visiting the unvisited neighboring molecule with the highest weight (Molecule 6), adding it to the path. This process continues until the path reaches a node where all its neighboring molecules have already been visited, as our graph is not a complete graph and only contains edges between molecules where one is among the other's k-nearest neighbors. In such cases, we extend the graph by again selecting the unvisited molecule with the smallest degree (Molecule 4) and repeat the process to obtain a new path as a molecular group. The motivation for starting from the minimum-degree molecule is that these molecules are most likely to have all their neighbors visited first, leading them to be connected to dissimilar molecules in the final path.

Algorithm 1 Modified maximum traveling salesman for Tanimoto similarity graph traversal.

Input: TSG $\mathcal{G}_T = (\mathcal{M}, \mathcal{E})$;
$N(n_i)$: the neighbor set of molecule n_i in $\mathcal{G}_T$;
`min_deg`$(\mathcal{M})$: returns a molecule with minimum degree in the current graph;
$\mathrm{sim}(n_i, n_j)$: Tanimoto similarity between molecules n_i and n_j.
Output: A set of traversal paths P, where each path corresponds to one molecular group.

Initialize the output path set: $P \leftarrow []$;
Initialize the set of unvisited molecules as $\mathcal{M}$;
while $|\mathcal{M}| > 0$ **do**
  Initialize a new path: $P_i \leftarrow []$;
  Select the starting molecule with minimum degree:
    $n_i \leftarrow$ `min_deg`$(\mathcal{M})$;
  Add n_i into the current path P_i;
  Remove n_i from the unvisited set $\mathcal{M}$;
  Compute the candidate neighbor set of the current molecule:
    $\mathcal{C} \leftarrow N(n_i) \cap \mathcal{M}$;
  while $\mathcal{C} \neq \emptyset$ **do**
    Select the most similar unvisited neighbor:
      $n_j \leftarrow \arg\max_{n \in \mathcal{C}} \mathrm{sim}(n_i, n)$;
    Move to the selected molecule: $n_i \leftarrow n_j$;
    Append n_i to the current path P_i;
    Remove n_i from the unvisited set $\mathcal{M}$;
    Update the candidate neighbor set:
      $\mathcal{C} \leftarrow N(n_i) \cap \mathcal{M}$;
  end while
  Add the completed path P_i to the output set P;
end while
Return all traversal paths as the molecular grouping result;
return P.

Finally, we treat the obtained path as a molecular group and input the molecules and their corresponding descriptive texts into the LLM for pretraining according to the path order. It is worth noting that when inputting the description of each molecule, we also include information on its ten lowest-energy conformations. The purpose of this is to enable the LLM to develop a certain degree of multimodal understanding during the pretraining process.

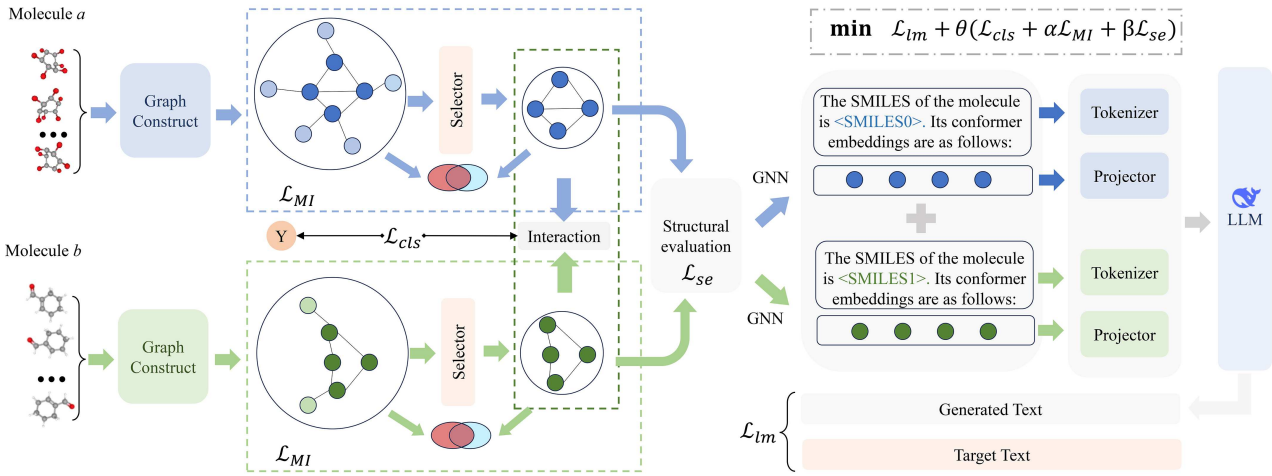

Figure 3 (Color online) A model fine-tuning framework incorporating the EGCS algorithm. The process is as follows: after inputting multiple molecular conformers, a conformer relationship graph is first constructed. Based on the constructed graph, the optimization objective of GIB is computed. Finally, it is iteratively updated in combination with the loss of the large model.

It is worth noting that in our implementation, we do not need to explicitly consider the selection of an ending node during graph traversal. As mentioned above, at the beginning of each grouping process, we select the unvisited molecule with the smallest degree in the graph as the starting point. Initially, we randomly choose one of the minimum-degree molecules to begin, and then start our iterative algorithm to visit the unvisited neighboring molecule with the highest edge weight, marking and recording it as belonging to the same molecular group. Our algorithm maximizes the grouping of related molecules while avoiding multiple occurrences of the same molecule. Therefore, once all molecules have been grouped, the iterations naturally terminate without the need to deliberately design an ending node.

3.2 Entropy-guided conformer selection finetuning

When incorporating molecular conformational information into an LLM, structural differences among conformers can lead to information interference if they are fed into the model without selection. This interference may affect the LLM's ability to accurately identify the reactive conformers. Therefore, it is crucial to select appropriate conformers to ensure the correct assessment of molecular interactions. Our conformer selection algorithm is shown in Figure 3.

3.2.1 *Conformation graph modeling: relationship capture*

Molecular conformers exist in a dynamic equilibrium, with each conformer having a certain probability of converting into other conformers. During the transition from one conformer to another, there is typically one conformer that is the most easily converted [43]. Generally, when one conformation transitions to another, only partial structural changes occur. As a result, the features of the two conformations involved in the transition tend to be more similar. Therefore, we can establish a conformation transition relationship graph through conformation feature calculations.

For multiple conformations of a molecule, we first need to encode each conformation. Let $\mathcal{C}^a = [\mathcal{C}_1^a, \ldots, \mathcal{C}_{n_a}^a]$ and $\mathcal{C}^b = [\mathcal{C}_1^b, \ldots, \mathcal{C}_{n_b}^b]$ represent the conformer pairs for molecule a and molecule b, where $\mathcal{C}_i^a$ and $\mathcal{C}_i^b$ are the i-th conformations, n_a and n_b are the number of conformations for each molecule. We leverage UniMol [44], a powerful 3D molecular feature encoder f, along with attention pooling p, to capture the embeddings of conformer pairs.

$$\begin{aligned} \mathbf{V}^a &= [v_1^a, v_2^a, \ldots, v_{n_a}^a], \text{where } v_i^a = p(f(\mathcal{C}_i^a)), \\ \mathbf{V}^b &= [v_1^b, v_2^b, \ldots, v_{n_b}^b], \text{where } v_i^b = p(f(\mathcal{C}_i^b)), \end{aligned} \tag{2}$$

where $\mathbf{V}^a$ and $\mathbf{V}^b$ represent the sets of conformation feature embeddings for the two molecules. Then, we calculate the feature similarity of each conformation in the molecule using Euclidean distance. We retain the edges with a similarity greater than a given threshold τ and use the similarity as the edge weight, constructing a conformation relationship graph with conformations as nodes. Taking molecule a as an example, specifically

$$e_{ij} = \begin{cases} \|v_i^a - v_j^a\|_2, & \text{if } \|v_i^a - v_j^a\|_2 \geqslant \tau, \\ 0, & \text{if } \|v_i^a - v_j^a\|_2 < \tau, \end{cases} \tag{3}$$

where e_{ij} represents the edge weight, and 0 indicates no edge. In this way, we obtain the conformation relationship graphs for molecule a and molecule b, denoted as $\mathcal{G}^a$ and $\mathcal{G}^b$.

3.2.2 *Graph information bottleneck: conformation selection*

We improve the graph information bottleneck (GIB) principle and integrate it into the conformation relationship graph, thereby enabling the selection of conformations.

Theorem 1 (GIB)**.** Given a graph $\mathcal{G}$ and its label Y, the GIB aims to find a compressed representation of the graph Z that retains key information by optimizing the following objective:

$$\max_{Z} I(Y,Z) \text{ s.t. } I(\mathcal{G},Z) \leqslant I_c, \tag{4}$$

where I_c is the information constraint between $\mathcal{G}$ and Z. By introducing a Lagrange multiplier β to (4), we reach its unconstrained form

$$\max_{Z} I(Y,Z) - \beta I(\mathcal{G},Z). \tag{5}$$

Eq. (5) gives a general formulation of GIB. In conformation selection, we focus on how to retain the conformation nodes that are closest to the prediction target while minimizing redundant information as much as possible. By combining (5) and structural entropy, we obtain the optimization objective for the key conformation subgraph (KC-subgraph). In order to better select subgraphs for subsequent updates, we use MLP to map node importance for selection.

Theorem 2 (KC-subgraph)**.** For conformation relationship graphs $\mathcal{G}^a$ and $\mathcal{G}^b$, their key conformation subgraphs, which maximize information while minimizing redundant structures, namely the KC-subgraphs, can be obtained by optimizing the following objective, taking G^a as an example:

$$\max_{\mathcal{G}^a_{\text{sub}} \in \mathsf{G}^a_{\text{sub}}} \frac{I(Y,\mathcal{G}^{ab}_{\text{sub}})}{2} - \alpha I(\mathcal{G}^a,\mathcal{G}^a_{\text{sub}}) + \beta \text{SE}(\mathcal{G}^a_{\text{sub}}), \tag{6}$$

where $\mathsf{G}^a_{\text{sub}}$ denotes the set of all subgraphs of $\mathcal{G}^a$, $\mathcal{G}^b_{\text{sub}}$ represents the selected subgraph of $\mathcal{G}^b$, and $\mathcal{G}^{ab}_{\text{sub}}$ denotes the fused representation of $\mathcal{G}^a_{\text{sub}}$ and $\mathcal{G}^b_{\text{sub}}$. As shown in (6), to select the most critical conformation, we need to optimize the model with respect to $I(Y,\mathcal{G}^{ab}_{\text{sub}})$, $I(\mathcal{G}^a,\mathcal{G}^a_{\text{sub}})$, and $\text{SE}(\mathcal{G}^a_{\text{sub}})$.

For the first term $I(Y,\mathcal{G}^{ab}_{\text{sub}})$, it represents the mutual information between the selected subgraph and the target. Therefore, we first expand it

$$I(Y,\mathcal{G}^{ab}_{\text{sub}}) = \int p(y,\mathcal{G}^{ab}_{\text{sub}}) \log p(y|\mathcal{G}^{ab}_{\text{sub}}) dy \; d\mathcal{G}^{ab}_{\text{sub}} + H(Y), \tag{7}$$

where $H(Y)$ is the entropy of Y and thus can be ignored. In practice, we approximate $p(y,\mathcal{G}^{ab}_{\text{sub}})$ with an empirical distribution $p(y,\mathcal{G}^{ab}_{\text{sub}}) \approx \frac{1}{N}\sum_{i=1}^{N} \delta_{y_i}(y)\delta_{\mathcal{G}^{ab}_{\text{sub},i}}(\mathcal{G}^{ab}_{\text{sub}})$, where $\mathcal{G}^{ab}_{\text{sub}}$ is the fused subgraph and Y is the label. By substituting the true posterior $p(y|\mathcal{G}^{ab}_{\text{sub}})$ with a variational approximation $q_{\phi_1}(y|\mathcal{G}^{ab}_{\text{sub}})$, we obtain a tractable lower bound of the first term in (6):

$$\begin{aligned} I(Y,\mathcal{G}^{ab}_{\text{sub}}) &\geqslant \int p(y,\mathcal{G}^{ab}_{\text{sub}}) \log q_{\phi_1}(y|\mathcal{G}^{ab}_{\text{sub}}) dy \; d\mathcal{G}^{ab}_{\text{sub}} \\ &\approx \frac{1}{N}\sum_{i=1}^{N} q_{\phi_1}(y_i|\mathcal{G}^{ab}_{\text{sub},i}) =: -\mathcal{L}_{\text{cls}}(q_{\phi_1}(y|\mathcal{G}^{ab}_{\text{sub}}), y_{gt}), \end{aligned} \tag{8}$$

where y_{gt} is the ground truth label of the graph. Eq. (8) indicates that maximizing $I(Y,\mathcal{G}^{ab}_{\text{sub}})$ is achieved by the minimization of the classification loss between Y and $\mathcal{G}^{ab}_{\text{sub}}$ as $\mathcal{L}_{\text{cls}}$. Intuitively, minimizing $\mathcal{L}_{\text{cls}}$ encourages the subgraph to be predictive of the graph label. In practice, we choose the cross entropy loss for categorical Y and the mean squared loss for continuous Y, respectively. The detailed derivation process can be found in Appendix C.

Then, we optimize $I(\mathcal{G}^a,\mathcal{G}^a_{\text{sub}})$. To reduce the dependence on prior assumptions about the graph data distribution, we adopt the Donsker-Varadhan representation [45] to express the KL divergence:

$$I(\mathcal{G}^a,\mathcal{G}^a_{\text{sub}}) = \sup_{f^a_{\phi_2}:\mathsf{G}\times\mathsf{G}\to\mathsf{R}} \mathbb{E}_{\mathcal{G}^a,\mathcal{G}^a_{\text{sub}}\in p(\mathcal{G}^a,\mathcal{G}^a_{\text{sub}})} f_{\phi^a_2}(\mathcal{G}^a,\mathcal{G}^a_{\text{sub}}) - \log\left(\mathbb{E}_{\mathcal{G}^a\in p(\mathcal{G}^a),\mathcal{G}^a_{\text{sub}}\in p(\mathcal{G}^a_{\text{sub}})} \mathrm{e}^{f_{\phi^a_2}(\mathcal{G}^a,\mathcal{G}^a_{\text{sub}})}\right), \tag{9}$$

where $f^a_{\phi_2}$ is the statistics network that maps from the graph set to the set of real numbers. Then, we use GIN [46] to extract embeddings from both $\mathcal{G}^a$ and $\mathcal{G}^a_{\text{sub}}$, and concatenate the embeddings of $\mathcal{G}^a$ and $\mathcal{G}^a_{\text{sub}}$, feeding them into a multi-layer perceptron to finally produce a real number. In conjunction with the sampling method to approximate $p(\mathcal{G}^a, \mathcal{G}^a_{\text{sub}})$, $p(\mathcal{G}^a)$, and $p(\mathcal{G}^a_{\text{sub}})$, we arrive at the following optimization problem to approximate:

$$\max_{\phi^a_2} \mathcal{L}_{\text{MI}}(\phi^a_2, \mathcal{G}^a_{\text{sub}}) = \frac{1}{N}\sum_{i=1}^{N} f_{\phi^a_2}(\mathcal{G}^a_i, \mathcal{G}^a_{\text{sub},i}) - \log \frac{1}{N}\sum_{i=1, j\neq i}^{N} \mathrm{e}^{f_{\phi^a_2}(\mathcal{G}^a_i, \mathcal{G}^a_{\text{sub},j})}. \tag{10}$$

For the selected subgraph, in order to reduce its internal complexity and enable the LLM to make more accurate judgments, we evaluate and optimize its structural entropy $\text{SE}(\mathcal{G}^a_{\text{sub}})$, denoted as $\mathcal{L}_{\text{SE}}(\mathcal{G}^a_{\text{sub}})$.

$$\mathcal{L}_{\text{SE}}(\mathcal{G}^a_{\text{sub}}) = -\sum_{i\in V^a_{\text{sub}}}\sum_{j\in V^a_{\text{sub}}} D_i D_j \left(1 + w^*_{ij}\right) \log\left(D_i D_j \left(1 + w^*_{ij}\right)\right), \tag{11}$$

where V^a_{sub} denotes the set of nodes in the subgraph $\mathcal{G}^a_{\text{sub}}$, $D_i = \frac{d_i}{\sum_{k\in V_{\text{sub}}} d_k}$ is the normalized degree of node v_i, d_i is the degree of node v_i, and $w^*_{ij} = \frac{w_{ij}}{\sum_{(i,j)\in E_{\text{sub}}} w_{ij}}$ is the normalized weight of the edge e_{ij}, reflecting the degree of connection between the two conformations.

Combining (8), (9), and (11), the final optimization objective for KC-subgraph $\mathcal{G}^a_{\text{sub}}$ is

$$\begin{aligned} &\min_{\mathcal{G}^a_{\text{sub}}, \phi_1} \mathcal{L}(\mathcal{G}^a_{\text{sub}}, \phi_1, \phi^{a*}_2) = \frac{\mathcal{L}_{\text{cls}}(q_{\phi_1}(y|\mathcal{G}^a_{\text{sub}}), y_{\text{gt}})}{2} + \alpha\mathcal{L}_{\text{MI}}(\phi^{a*}_2, \mathcal{G}^a_{\text{sub}}) + \beta\mathcal{L}_{\text{SE}}(\mathcal{G}^a_{\text{sub}}), \\ &\text{s.t.} \quad \phi^{a*}_2 = \arg\max_{\phi^a_2} \mathcal{L}_{\text{MI}}(\phi^a_2, \mathcal{G}^a_{\text{sub}}). \end{aligned} \tag{12}$$

By optimizing in the same way, we obtain the KC-subgraphs $\mathcal{G}^a_{\text{sub}}$ and $\mathcal{G}^b_{\text{sub}}$. The feature representation encoded by GIN is denoted as $\mathbf{Z}_a$ and $\mathbf{Z}_b$.

3.3 LLM generation: judgment of intermolecular relationships

In order for the LLM to correctly understand information from the two different modalities of semantics and conformation, the next step is to map them into the same semantic space using projectors f_{proj1}, f_{proj2}, where f_{proj1} and f_{proj2} share the same parameters. These projectors act as critical connectors, translating $\mathbf{Z}_a$ and $\mathbf{Z}_b$ into LLM-comprehensible encodings $\mathbf{M}_a$ and $\mathbf{M}_b$. We instantiate f_{proj1} and f_{proj2} using trainable projection matrices, which share the same dimensionality as the word embedding space in the language model. More formally, the encodings can be expressed as

$$\begin{aligned} \mathbf{M}_a &= [\mathbf{m}^a_1, \mathbf{m}^a_2, \ldots, \mathbf{m}^a_{\text{dim}}] = f_{\text{proj1}}(\mathbf{Z}_a), \\ \mathbf{M}_b &= [\mathbf{m}^b_1, \mathbf{m}^b_2, \ldots, \mathbf{m}^b_{\text{dim}}] = f_{\text{proj2}}(\mathbf{Z}_b), \end{aligned} \tag{13}$$

where dim denotes the feature dimensionality of the LLM and m_i represents the embedded representation of a single aligned conformer. Next, we use DeepSeek [10], which has undergone incremental pretraining, as the backbone of Gelm. Compared to other LLMs, DeepSeek demonstrates better logical reasoning ability and can more accurately determine intermolecular interaction relationships based on the given information. The prompt sequence $\mathbf{X}$ is as follows:

$$\mathbf{X} = \{\mathbf{P}, \mathbf{M}_a, \mathbf{M}_b\} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \quad \text{s.t. } \mathbf{P} \sim \mathcal{P}, \tag{14}$$

where n denotes the total integrated input length, $\mathbf{P}$ represents the task-specific prompt, and $\mathcal{P}$ refers to a collection of manually designed prompts, each specifically crafted for the molecular interaction task $\mathbf{r}$. The generation process employs a causal mask to produce a response that encapsulates the key interactive properties, with a length of L:

$$\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_L]. \tag{15}$$

The training objective is to predict the target response from the input prompt $\mathbf{X}$. Specifically, the output for the i-th token, represented as $\hat{\mathbf{x}}_i$, is determined based on its preceding tokens as follows for $t \in (1, L)$:

$$p\left(\hat{\mathbf{X}}_{[1:t]}|\mathbf{X}\right) = \prod_{i=1}^{t} p\left(\hat{\mathbf{x}}_i|\mathbf{X}, \hat{\mathbf{X}}_{[1:i-1]}\right). \tag{16}$$

Prompt for DDI Tasks (Fine-tuning)
Here are two molecules: the first molecule has a SMILES representation of <SMILES0>, with corresponding conformation embedding features <ConEmb0_0>, <ConEmb0_1>, ..., while the second molecule has a SMILES representation of <SMILES1>, with corresponding conformation embedding features <ConEmb1_0>, <ConEmb1_1>, ... Given these conformations, what are the potential side effects of these two drugs?

Target Response for DDI Tasks (Fine-tuning)
Considering the molecular conformation, the first molecule may exhibit the property [Property0], while the second molecule may possess the property [Property1]. This interaction could potentially enhance the photoreactivity of the second molecule. Therefore, they are likely to interact with each other.

Prompt for SSI Tasks (Fine-tuning)
Here are two molecules: the first molecule has a SMILES representation of <SMILES0>, with corresponding conformation embedding features <ConEmb0_0>, <ConEmb0_1>, ..., while the second molecule has a SMILES representation of <SMILES1>, with corresponding conformation embedding features <ConEmb1_0>, <ConEmb1_1>, ... Given these conformations, what is the solvation Gibbs free energy of this pair of molecules?

Target Response for SSI Tasks (Fine-tuning)
Considering the molecular conformation, the first molecule may exhibit the property [Property0], while the second molecule may possess the property [Property1]. Therefore, the solvation Gibbs free energy of these two molecules is expected to be between `10.0` and `10.5`, with the predicted value being `10.232`.

Figure 4 Prompts for different tasks, including qualitative tasks (DDI) and quantitative tasks (SSI).

Therefore, in Gelm, the total LOSS is

$$\begin{aligned}\mathcal{L} &= -\sum_{t=1}^{n} \log p(x_t \mid x_1, x_2, \ldots, x_{t-1}) + \theta(\mathcal{L}(\mathcal{G}^a_{\text{sub}}, \phi_1, \phi_2^{a*}) + \mathcal{L}(\mathcal{G}^b_{\text{sub}}, \phi_1, \phi_2^{b*})), \\ \text{s.t.} \quad & \phi_2^{a*} = \arg\max_{\phi_2^a} \mathcal{L}_{\text{MI}}(\phi_2^a, \mathcal{G}^a_{\text{sub}}),\ \phi_2^{b*} = \arg\max_{\phi_2^b} \mathcal{L}_{\text{MI}}(\phi_2^b, \mathcal{G}^b_{\text{sub}}),\end{aligned} \tag{17}$$

where the first term is $\mathcal{L}_{lm}$ and θ is a hyperparameter that controls the balance between the two losses.

For different tasks related to molecular interactions, we have designed distinct prompts. The details of the prompt are presented in Figure 4. Here, SMILES0 and SMILES1 represent the SMILES notations, ConEmb0 and ConEmb1 represent multiple conformer tokens $\mathbf{M}_a$ and $\mathbf{M}_b$. For DDI tasks, the goal is to enable the LLM to leverage the conformational information of the two input molecules. By doing so, the LLM actively determines and provides an answer as to whether there is an interaction between the two molecules, drawing on their respective conformations to make a judgment.

For SSI tasks, where LLMs are less adept, we design a prompt that encourages the model to consider molecular conformations during inference. Instead of a direct answer, the model predicts a range of possible interaction outcomes based on the conformations. It then synthesizes the data to estimate a specific value within that range.

4 Experimental results

4.1 Experimental setting

We evaluate Gelm on well-established downstream molecule interaction tasks involving qualitative and quantitative analysis. Here we provide an overview of our experimental setup. Detailed descriptions are presented in Appendixes A and B.

Datasets. We employ 12 datasets across various domains such as DDI, SSI, and CSI. Specifically, we collect Drugbank (Version 5.0.3), ZhangDDI [47], ChChMiner [48], DeepDDI [49], TWOSIDES [50], Chromophore [51], MNSol [52], CompSol [53], Abraham [54], CombiSolv [55], FreeSolv [56] and CombiSolv-QM [55].

Baselines. For a comprehensive evaluation, we conduct various baseline methods encompassing distinct categories such as methods based on: GNNs, DL models other than GNN, and LLMs. Specifically, for DDI tasks, we employ IGIB-ISE [57], MHCADDI [58], DeepDDI [49], SSI-DDI, CGIB, CMRL [1], MDF-SA-DDI [59], DSN-DDI [60], MolTC [6] as the baseline. For SSI and CSI tasks, we utilize D-MPNN [55], SolvBert [61], SMD [62], CGIB, MMGNN [63], GEM [64], GROVER [65], Uni-Mol [44] as the baseline. Furthermore, all downstream tasks adopt LLM-based methods, such as Galactica, Chem T5 [66], MolT5, MolCA [8] and MolTC as the baseline.

Training epochs. At the beginning of the experiment, we first perform incremental pretraining, conducting 10 epochs on the collected pretraining dataset. Then, during the fine-tuning phase, the number of training epochs varies for different tasks. For the DDI task, we typically fine-tune for 50 epochs. For SSI datasets with more than 3000 molecular pairs, we first fine-tune on the CombiSolv-QM dataset for 100 epochs, followed by an additional 30 epochs on their respective datasets. For SSI datasets with fewer than 3000 molecular pairs, this number is

Table 1 Comparative performance of various methods in qualitative interactive tasks. The best-performing methods are highlighted in bold, while the second-best methods are marked with * for emphasis.

Baseline model		Drugbank		ZhangDDI		ChChMiner		DeepDDI	
		Accuracy	AUC-ROC	Accuracy	AUC-ROC	Accuracy	AUC-ROC	Accuracy	AUC-ROC
GNN-based	IGIB-ISE	$95.03_{\pm 0.37}$	$98.79_{\pm 0.54}$	$88.64_{\pm 0.32}$	$94.58_{\pm 0.38}$	$94.81_{\pm 0.36}$	$97.71_{\pm 0.33}$	$96.15_{\pm 0.34}$	$98.44_{\pm 0.32}$
	SSI-DDI	$94.09_{\pm 0.31}$	$98.24_{\pm 0.28}$	$86.94_{\pm 0.33}$	$93.55_{\pm 0.39}$	$93.07_{\pm 0.22}$	$97.83_{\pm 0.13}$	$94.85_{\pm 0.24}$	$98.33_{\pm 0.31}$
	DSN-DDI	$94.91_{\pm 0.11}$	$98.96^{*}{}_{\pm 0.10}$	$87.38_{\pm 0.12}$	$94.71_{\pm 0.29}$	$84.18_{\pm 0.17}$	$94.12_{\pm 0.27}$	$95.36_{\pm 0.27}$	$98.09_{\pm 0.16}$
	CMRL	$94.89_{\pm 0.12}$	$98.68_{\pm 0.11}$	$87.75_{\pm 0.33}$	$94.45_{\pm 0.21}$	$94.12_{\pm 0.25}$	$98.28_{\pm 0.13}$	$96.22_{\pm 0.35}$	$98.93^{*}{}_{\pm 0.30}$
	CGIB	$94.62_{\pm 0.33}$	$98.48_{\pm 0.25}$	$87.75_{\pm 0.71}$	$93.92_{\pm 0.60}$	$94.30_{\pm 0.37}$	$98.39^{*}{}_{\pm 0.31}$	$96.07_{\pm 0.48}$	$97.91_{\pm 0.63}$
ML-based	DeepDDI	$93.12_{\pm 0.26}$	$98.41_{\pm 0.53}$	$83.81_{\pm 0.47}$	$91.38_{\pm 0.57}$	$90.58_{\pm 0.63}$	$95.97_{\pm 0.28}$	$92.53_{\pm 0.39}$	$98.18_{\pm 0.42}$
	MHCADDI	$79.10_{\pm 0.81}$	$86.13_{\pm 0.46}$	$77.85_{\pm 0.48}$	$87.21_{\pm 0.69}$	$84.71_{\pm 0.54}$	$90.13_{\pm 0.81}$	$87.81_{\pm 0.78}$	$88.76_{\pm 0.74}$
	MDF-SA-DDI	$94.02_{\pm 0.33}$	$97.72_{\pm 0.30}$	$86.83_{\pm 0.26}$	$94.28_{\pm 0.34}$	$93.81_{\pm 0.22}$	$98.19_{\pm 0.20}$	$94.94_{\pm 0.32}$	$97.80_{\pm 0.35}$
LLM-based	Galactica	$79.35_{\pm 0.33}$	$86.33_{\pm 0.33}$	$67.47_{\pm 0.55}$	$79.15_{\pm 0.58}$	$74.63_{\pm 0.42}$	$84.15_{\pm 0.66}$	$71.34_{\pm 0.43}$	$79.21_{\pm 0.39}$
	Chem T5	$86.10_{\pm 0.30}$	$92.20_{\pm 0.36}$	$72.57_{\pm 0.38}$	$89.60_{\pm 0.31}$	$81.12_{\pm 0.53}$	$85.48_{\pm 0.47}$	$75.95_{\pm 0.64}$	$84.63_{\pm 0.45}$
	MolCA	$87.96_{\pm 0.48}$	$94.12_{\pm 0.36}$	$68.70_{\pm 0.61}$	$88.42_{\pm 0.51}$	$90.27_{\pm 0.44}$	$93.14_{\pm 0.60}$	$83.11_{\pm 0.56}$	$89.15_{\pm 0.73}$
	MolT5	$89.72_{\pm 0.36}$	$93.28_{\pm 0.36}$	$77.02_{\pm 0.40}$	$87.79_{\pm 0.52}$	$81.05_{\pm 0.36}$	$90.26_{\pm 0.37}$	$89.34_{\pm 0.36}$	$94.10_{\pm 0.32}$
	MolTC	$95.94^{*}{}_{\pm 0.17}$	$98.93_{\pm 0.21}$	$89.45^{*}{}_{\pm 0.12}$	$95.69^{*}{}_{\pm 0.16}$	$95.68^{*}{}_{\pm 0.22}$	$98.18_{\pm 0.19}$	$96.78^{*}{}_{\pm 0.27}$	$98.91_{\pm 0.41}$
Gelm (ours)		$\mathbf{96.38}_{\pm 0.14}$	$\mathbf{99.10}_{\pm 0.12}$	$\mathbf{92.60}_{\pm 0.30}$	$\mathbf{96.13}_{\pm 0.11}$	$\mathbf{96.44}_{\pm 0.33}$	$\mathbf{98.52}_{\pm 0.16}$	$\mathbf{97.34}_{\pm 0.24}$	$\mathbf{99.08}_{\pm 0.32}$

adjusted to 20 epochs. Furthermore, both the pretraining and fine-tuning phases employ the same configuration for the optimizer and learning rate scheduler, as detailed in the following section.

Training strategy. We employ the AdamW optimizer with a weight decay set at 0.05. Our learning rate strategy utilizes a combination of linear warm-up and cosine decay, optimizing the training process by initially increasing the learning rate to promote faster convergence, and then gradually decreasing it according to a cosine curve to fine-tune the model parameters. LoRA is implemented using the Open Delta library, and the PEFT library. LoRA's rank r is set to 16, while LoRA is applied to DeepSeek's modules of `[q_proj, k_proj, v_proj, o_proj, gate_proj, up_proj, down_proj]`. Regarding the hyperparameter settings of the loss, we provide in Appendix D the effects of different configurations to assist readers in making their choice and judgment.

4.2 Evaluation metrics

We employ prediction accuracy (%) and AUC-ROC (area under the receiver operating characteristic curve) as comparative metrics, while for quantitative tasks, MAE (mean absolute error) and RMSE (root mean square error) are utilized as the metrics.

4.3 Experimental results and analysis

Here, we present models based on GNNs, ML, and LLMs, all of which demonstrate outstanding performance across various domains. Since the evaluation of Gelm involves two distinct primary tasks, regression (DDI) and quantization (SSI), we will discuss its performance separately for each task. Additional experimental results are provided in Appendix D.

Qualitative prediction performance. Table 1 presents the comparative performance of Gelm and various baseline methods on four widely used chemical datasets. The experimental results clearly demonstrate that Gelm consistently outperforms existing methods across all datasets. For example, on the ZhangDDI dataset, our model achieves nearly a 3% improvement in accuracy compared to MolTC. This improvement can be attributed to our innovative combination of graph-based Tanimoto similarity grouping pretraining and 3D molecular conformational information, which allows for more effective integration of conformational data. This method enables our model to better capture the 3D structural features of molecules, leading to a deeper understanding of how molecular structures influence molecular properties. Furthermore, compared to the best-performing GNN model, IGIB-ISE, and the top machine learning-based method, MDF-SA-DDI, Gelm achieves significant performance improvements across multiple dimensions. This breakthrough is attributed to the DeepSeek model we use, which, with its extensive knowledge base and incremental pretraining mechanism, provides enriched knowledge support for the model. This knowledge enhancement significantly improves the model's performance in reasoning tasks, especially in the field of molecular property prediction, showcasing its unique advantages in understanding molecular data.

Quantitative prediction performance. Table 2 presents the performance of Gelm on quantitative tasks. The data indicates that while LLM-based models generally perform slightly worse than GNN-based and ML-based

Table 2 Comparative performance of various methods in quantitative interactive tasks. The best-performing methods are highlighted in bold, while the second-best methods are marked with * for emphasis.

Baseline model		FreeSolv		Abraham		CompSol		CombiSolv	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
GNN-based	MMGNN	$0.548_{\pm 0.032}$	$0.924_{\pm 0.038}$	$0.199_{\pm 0.009}$	$0.403_{\pm 0.012}$	$0.175^{*}_{\pm 0.005}$	$0.302_{\pm 0.005}$	$0.189_{\pm 0.004}$	$0.393^{*}_{\pm 0.015}$
	D-MPNN	$0.718_{\pm 0.015}$	$1.240_{\pm 0.030}$	$0.501_{\pm 0.012}$	$0.725_{\pm 0.028}$	$0.213_{\pm 0.008}$	$0.380_{\pm 0.009}$	$0.498_{\pm 0.015}$	$0.910_{\pm 0.056}$
	GEM	$0.615_{\pm 0.019}$	$1.210_{\pm 0.053}$	$0.259_{\pm 0.006}$	$0.542_{\pm 0.009}$	$0.208_{\pm 0.007}$	$0.348_{\pm 0.006}$	$0.303_{\pm 0.010}$	$0.785_{\pm 0.020}$
	CGIB	$0.552_{\pm 0.012}$	$0.934_{\pm 0.058}$	$0.264_{\pm 0.009}$	$0.536_{\pm 0.010}$	$0.177_{\pm 0.005}$	$0.317_{\pm 0.005}$	$0.234_{\pm 0.004}$	$0.397_{\pm 0.010}$
ML-based	GROVER	$0.642_{\pm 0.027}$	$1.087_{\pm 0.046}$	$0.360_{\pm 0.010}$	$0.628_{\pm 0.018}$	$0.189_{\pm 0.007}$	$0.374_{\pm 0.016}$	$0.421_{\pm 0.018}$	$0.742_{\pm 0.036}$
	SolvBert	$0.595_{\pm 0.032}$	$1.061_{\pm 0.055}$	$0.493_{\pm 0.009}$	$0.692_{\pm 0.016}$	$0.194_{\pm 0.010}$	$0.353_{\pm 0.009}$	$0.433_{\pm 0.019}$	$0.714_{\pm 0.022}$
	Uni-Mol	$0.579_{\pm 0.062}$	$1.015_{\pm 0.074}$	$0.368_{\pm 0.009}$	$0.618_{\pm 0.025}$	$0.199_{\pm 0.003}$	$0.348_{\pm 0.004}$	$0.272_{\pm 0.006}$	$0.675_{\pm 0.018}$
	SMD	$0.615_{\pm 0.038}$	$1.224_{\pm 0.037}$	$0.404_{\pm 0.023}$	$0.654_{\pm 0.038}$	$0.199_{\pm 0.007}$	$0.351_{\pm 0.008}$	$0.663_{\pm 0.013}$	$1.018_{\pm 0.032}$
LLM-based	Galactica	$0.889_{\pm 0.012}$	$1.395_{\pm 0.068}$	$0.648_{\pm 0.009}$	$1.062_{\pm 0.017}$	$0.598_{\pm 0.009}$	$0.863_{\pm 0.009}$	$0.839_{\pm 0.022}$	$1.454_{\pm 0.040}$
	Chem T5	$0.815_{\pm 0.037}$	$1.355_{\pm 0.058}$	$0.641_{\pm 0.011}$	$0.918_{\pm 0.018}$	$0.450_{\pm 0.009}$	$0.740_{\pm 0.011}$	$0.895_{\pm 0.016}$	$1.311_{\pm 0.025}$
	MolCA	$0.775_{\pm 0.036}$	$1.298_{\pm 0.042}$	$0.587_{\pm 0.008}$	$0.891_{\pm 0.012}$	$0.479_{\pm 0.009}$	$0.730_{\pm 0.024}$	$0.641_{\pm 0.044}$	$1.098_{\pm 0.038}$
	MolT5	$0.719_{\pm 0.048}$	$1.120_{\pm 0.076}$	$0.555_{\pm 0.010}$	$0.845_{\pm 0.007}$	$0.487_{\pm 0.004}$	$0.703_{\pm 0.009}$	$0.678_{\pm 0.032}$	$1.118_{\pm 0.030}$
	MolTC	$0.496^{*}_{\pm 0.014}$	$0.698^{*}_{\pm 0.043}$	$0.197^{*}_{\pm 0.012}$	$0.391^{*}_{\pm 0.011}$	$0.181_{\pm 0.007}$	$0.298^{*}_{\pm 0.005}$	$0.186^{*}_{\pm 0.005}$	$0.414_{\pm 0.011}$
Gelm (ours)		$\mathbf{0.471}_{\pm 0.014}$	$\mathbf{0.655}_{\pm 0.030}$	$\mathbf{0.176}_{\pm 0.009}$	$\mathbf{0.374}_{\pm 0.013}$	$\mathbf{0.168}_{\pm 0.007}$	$\mathbf{0.284}_{\pm 0.003}$	$\mathbf{0.175}_{\pm 0.006}$	$\mathbf{0.384}_{\pm 0.013}$

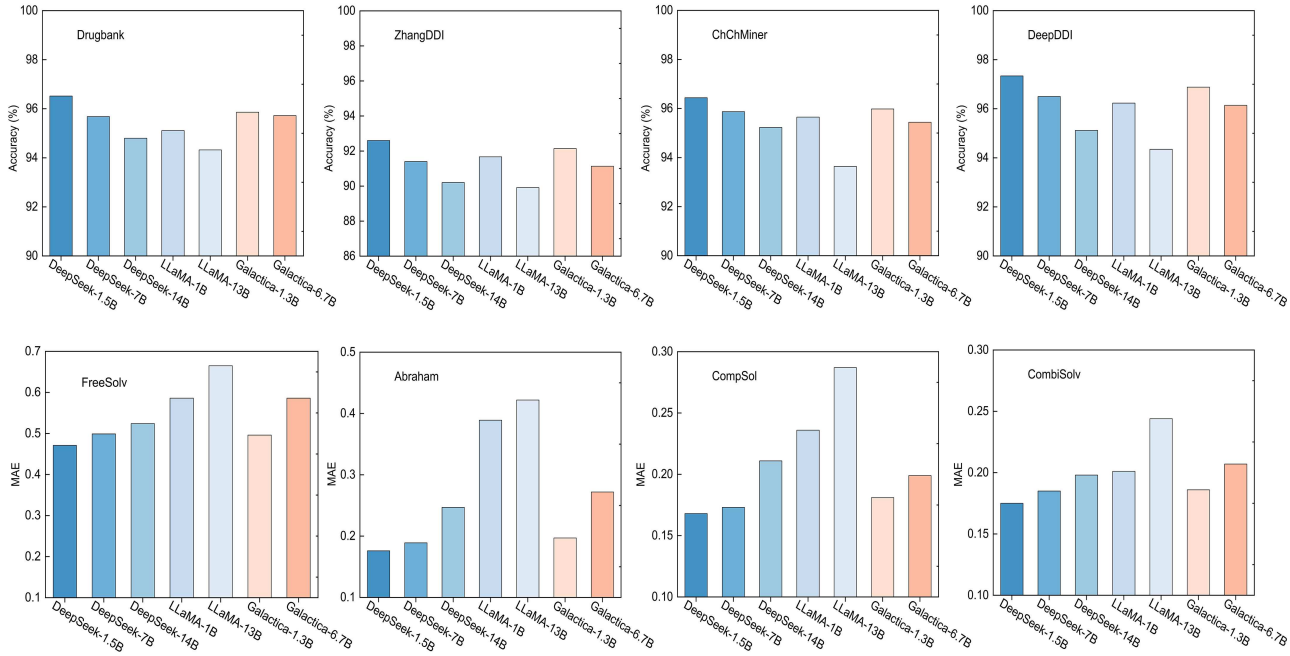

Figure 5 (Color online) Experimental results of different LLMs as the backbone.

models, Gelm has maintained a leading position in quantitative analysis tasks, which are traditionally considered challenging for LLMs. Notably, on the CombiSolv dataset, Gelm achieved an RMSE of 0.384, representing an improvement of nearly 7% compared to the second-best LLM-based model, MolTC, which had an RMSE of 0.414. Furthermore, Gelm significantly outperformed the best GNN-based model, MMGNN. For instance, on the FreeSolv dataset, Gelm reduced the RMSE by nearly 29%, demonstrating its superior predictive capabilities. These substantial improvements can be attributed to Gelm's enhanced understanding of molecular structures, enabling the model to capture molecular interactions with greater accuracy. Unlike conventional LLMs, which often struggle with numerical precision in chemical property predictions, Gelm leverages advanced molecular representation techniques to bridge this gap. The integration of chain-of-thought reasoning further compensates for LLMs' inherent weaknesses in quantitative analysis by guiding the model through a more structured and logical reasoning process. Additionally, Gelm's ability to incorporate 3D molecular conformational information provides a more holistic perspective on molecular behavior, ultimately leading to more precise and reliable predictions.

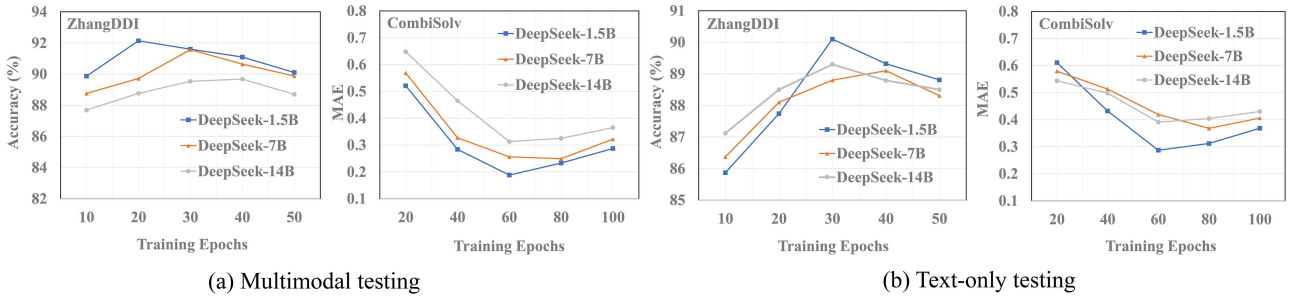

Figure 6 (Color online) Model performance evaluation across different scales under both multimodal and non-multimodal settings.

4.4 Performance comparison of different LLMs as backbones

For the selection of the backbone model, we conduct comprehensive testing across multiple models of varying parameter scales, including DeepSeek-1.5B, DeepSeek-7B, DeepSeek-14B [10], LLaMA-1B, LLaMA-13B [67], Galactica-1.3B, and Galactica-6.7B [68]. Our experimental results indicate that, at comparable parameter scales, the DeepSeek models consistently outperform other LLMs, demonstrating superior task adaptability and overall performance. The experimental results are shown in Figure 5. We can observe that at equivalent or similar parameter scales, DeepSeek demonstrates a clear advantage over other LLMs in SSI tasks. This suggests that DeepSeek not only possesses a stronger understanding of molecular properties but also excels in quantitative reasoning, making it a highly promising model for scientific computation tasks that demand high numerical precision.

Interestingly, we find that larger models do not necessarily achieve better results as model size increases. This phenomenon, known as inverse scaling, has been discussed in prior work [69,70], where enlarging a model or dataset can actually worsen performance on certain tasks, thereby challenging traditional assumptions about scaling in machine learning. Compared with smaller models, larger language models are less likely to avoid questions or tasks beyond their capabilities and instead tend to respond with greater confidence-even when incorrect [70]. In the context of molecular relationship learning, a domain underrepresented in the training data of LLMs compared with more common knowledge areas, this effect becomes particularly evident: larger models may not perform better and may even generate more confident yet wrong answers, which aligns with our experimental findings where smaller models sometimes outperform their larger counterparts on MRL tasks.

To further explain this phenomenon, we conducted additional experiments using the three DeepSeek models of different scales as the testing basis. Taking the ZhangDDI dataset and CombiSolv as examples, we re-split the datasets to reduce the influence of data splitting strategy. We also plotted the performance curves during training. As shown in Figure 6, models of different scales all reached their best performance before training reached the maximum epoch. Therefore, this phenomenon is not caused by insufficient fine-tuning.

In addition, as shown in Figure 6, our two testing settings reveal an interesting trend. In the text-only evaluation, larger models initially achieve the best performance at the early stages of training. However, as training progresses, smaller models adapt more quickly to the domain-specific dataset, while larger models adapt much more slowly. Even as the number of training epochs increases and the models begin to overfit, the larger models still exhibit a certain inertia, persisting in their earlier beliefs and struggling to revise incorrect knowledge. We believe this is partly related to the general-purpose knowledge acquired during the pretraining phase of LLMs. Compared with the vast amount of general knowledge, the chemical-domain SMILES-based pretraining we provide constitutes only a small proportion. As a result, when confronted with domain knowledge that conflicts with their pretrained general knowledge, larger models may struggle to adjust their reasoning, leaving them at a disadvantage compared with smaller models. This effect becomes even more pronounced when conformational information is incorporated into training.

4.5 Ablation study

Overall ablation. Table 3 presents the results of the ablation experiments. In our ablation experiments, we first compared the performance of models based on three different pretraining strategies: in-context pretraining (ICP) [7], random molecular description input (Random), and no pretraining (Free). All of these models were benchmarked using the GTSG-based pretraining method. From the data presented in the Table 3, it is clear that when replaced with other pretraining methods, the model performance generally decreases across various datasets. Both qualitative and quantitative tasks show varying degrees of performance degradation. It is particularly noteworthy that in some quantitative tasks, pretraining with random input can lead to the model learning disorganized knowledge, which

Table 3 The results of the ablation study. w/o indicates that this method is not used.

Dataset	Metric	w/o GTSG			w/o EGCS		
		ICP	Random	Free	All	Partial	None
DDI	Accuracy	$0.39_{\pm 0.07}$	$0.84_{\pm 0.14}$	$1.41_{\pm 0.23}$	$0.73_{\pm 0.07}$	$0.87_{\pm 0.16}$	$2.24_{\pm 0.35}$
	Rate (↓)	0.48%	1.03%	1.72%	0.90%	1.07%	2.75%
	AUC-ROC	$0.47_{\pm 0.09}$	$0.99_{\pm 0.18}$	$1.69_{\pm 0.27}$	$0.87_{\pm 0.14}$	$1.05_{\pm 0.17}$	$2.52_{\pm 0.39}$
	Rate (↓)	0.51%	1.08%	1.83%	0.94%	1.13%	2.73%
SSI	MAE	$0.008_{\pm 0.002}$	$0.013_{\pm 0.004}$	$0.029_{\pm 0.007}$	$0.018_{\pm 0.006}$	$0.022_{\pm 0.007}$	$0.040_{\pm 0.011}$
	Rate (↑)	3.64%	5.92%	13.20%	8.20%	10.02%	18.22%
	RMSE	$0.014_{\pm 0.004}$	$0.019_{\pm 0.005}$	$0.042_{\pm 0.010}$	$0.023_{\pm 0.007}$	$0.031_{\pm 0.008}$	$0.052_{\pm 0.015}$
	Rate (↑)	3.10%	4.21%	9.30%	5.09%	6.86%	11.52%
CSI Abs.	MAE	$0.38_{\pm 0.04}$	$0.67_{\pm 0.11}$	$0.54_{\pm 0.12}$	$0.61_{\pm 0.10}$	$0.78_{\pm 0.12}$	$1.51_{\pm 0.25}$
	Rate (↑)	3.18%	5.60%	4.51%	5.10%	6.52%	12.63%
	RMSE	$0.44_{\pm 0.05}$	$0.88_{\pm 0.14}$	$0.68_{\pm 0.13}$	$0.72_{\pm 0.06}$	$0.90_{\pm 0.15}$	$2.15_{\pm 0.34}$
	Rate (↑)	2.33%	4.67%	3.60%	4.64%	4.77%	11.39%
CSI Emis.	MAE	$1.37_{\pm 0.24}$	$2.07_{\pm 0.17}$	$1.86_{\pm 0.21}$	$2.35_{\pm 0.24}$	$1.89_{\pm 0.16}$	$3.81_{\pm 0.31}$
	Rate (↑)	7.35%	9.44%	8.48%	10.72%	8.77%	17.65%
	RMSE	$1.69_{\pm 0.17}$	$2.58_{\pm 0.28}$	$2.33_{\pm 0.22}$	$2.79_{\pm 0.32}$	$2.33_{\pm 0.19}$	$5.10_{\pm 0.35}$
	Rate (↑)	6.18%	9.42%	8.51%	10.19%	8.51%	18.64%
CSI Life.	MAE	$0.027_{\pm 0.005}$	$0.047_{\pm 0.008}$	$0.042_{\pm 0.007}$	$0.042_{\pm 0.007}$	$0.051_{\pm 0.009}$	$0.085_{\pm 0.013}$
	Rate (↑)	4.29%	7.64%	6.83%	6.67%	8.10%	13.50%
	RMSE	$0.041_{\pm 0.008}$	$0.068_{\pm 0.010}$	$0.053_{\pm 0.012}$	$0.054_{\pm 0.010}$	$0.073_{\pm 0.012}$	$0.110_{\pm 0.016}$
	Rate (↑)	4.63%	7.68%	5.99%	6.10%	8.25%	12.43%

negatively impacts its judgment. For example, in the CSI-related task, the model without pretraining outperforms the one pretrained with random input, and this phenomenon is especially pronounced. However, when molecular text descriptions are concatenated and then pretrained, the model shows improved performance on the CSI task. We believe that this enhancement is primarily due to the molecular description pretraining, which helps the model achieve better category recognition, thereby boosting performance in specific tasks. Overall, these experimental results highlight the superiority of our GTSG-based pretraining method.

After adopting our GTSG pretraining method, we conducted further experiments to compare the performance when EGCS was removed. Specifically, we performed experiments under three different scenarios: using all conformations (All), randomly retaining partial conformations (Partial), and not using any conformations (None). The results presented in Table 3 clearly indicate that when no conformations are used as supplementary information, the model performance experiences a significant drop. This finding validates the necessity of incorporating conformations as additional information in our experiments. Interestingly, when only partial conformations are randomly retained, we observe that, due to the inherent randomness, the model performs better on certain datasets compared to using all conformations. This further emphasizes the importance of carefully selecting conformations, rather than simply using all available conformations. However, it is also evident that using all conformations can lead to a certain degree of performance degradation. These results collectively support the superiority of our EGCS method. They demonstrate that while conformations are critical for model performance, the way in which they are selected and incorporated plays a crucial role in optimizing results. Our findings suggest that a well-balanced and thoughtful approach to conformation selection is essential for achieving the best performance, highlighting the effectiveness of the EGCS method in enhancing model outcomes.

In-depth analysis of GTSG. We propose a pretraining method based on GTSG. However, verifying the effectiveness of this strategy solely through downstream tasks and ablation studies cannot directly demonstrate that our approach enhances the model's ability to perceive structural similarity. Therefore, we designed an additional experiment to explicitly validate that the GTSG-based pretraining method indeed strengthens the model's structural awareness, while also evaluating the generalization performance of the model trained under the GTSG approach. Specifically, we further split the pretraining dataset by holding out 10% of the data, ensuring that it is excluded from the pretraining process and reserved solely for evaluation. We then perform incremental pretraining on the backbone model using the remaining data and subsequently evaluate the pretrained backbone on the held-out 10% subset through structure-based property prediction tasks.

Here, we compare several pretraining strategies, including random splitting, the ICP method [7], and a no-

Table 4 Model performance under different pretraining strategies.

Experiment	GTSG	Random	ICP [7]	MolTC [6]	Free
Accuracy	0.848 ± 0.016	0.702 ± 0.020	0.792 ± 0.019	0.736 ± 0.017	0.611 ± 0.014

pretraining baseline. In addition, we adapt the pretraining strategy from MolTC [6] by modifying it for integration into LLM pretraining. Specifically, in the modified strategy, molecular pairs are fed into the LLM for incremental pretraining, where the model is tasked with predicting potential interactions between the paired molecules. The specific experimental results are shown in Table 4. Here, we adopt DeepSeek-1.5B as the testing backbone, which is also the final backbone selected for our model.

From the data in Tables 3 and 4, it can be seen that our GTSG pretraining strategy not only effectively enhances the LLM's ability to perceive molecular structures but also improves its generalization in predicting molecular properties based on structural information, thereby significantly boosting performance on downstream MRL tasks.

5 Conclusion

In this paper, we introduce an innovative pretraining method for chemical datasets based on LLMs, addressing the lack of relational awareness in traditional chemical LLM pretraining. During the fine-tuning stage, we incorporate information entropy theory and structural entropy to guide the selection of key molecular conformers, enhancing the model's ability to capture essential structural information. Furthermore, Gelm has been extensively tested across multiple LLM backbones, providing concrete insights for selecting suitable LLM architectures for MRL tasks. This resolves the long-standing challenge of the lack of guidance on LLM selection in traditional MRL research. Experiments across twelve diverse datasets from various domains demonstrate the superiority of Gelm, with DeepSeek as the backbone, over existing GNN-based and LLM-based baselines, setting a new benchmark for integrating multimodal data in LLM-driven MRL. Limitations can be referred to in Appendix E.

Acknowledgements This work was partially supported by National Natural Science Foundation of China (Grant Nos. 62502491, 62072427), Project of Stable Support for Youth Team in Basic Research Field, CAS (Grant No. YSBR-005), Anhui Science Foundation for Distinguished Young Scholars (Grant No. 1908085J24), and Jiangsu Natural Science Foundation (Grant No. BK20191193). The AI-driven experiments, simulations and model training were performed on the robotic AI-Scientist platform of Chinese Academy of Sciences.

Supporting information Appendixes A–E. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Lee N, Yoon K, Na G S, et al. Shift-robust molecular relational learning with causal substructure. ArXiv:2305.18451
2 Lin X, Quan Z, Wang Z J, et al. Kgnn: Knowledge graph neural network for drug-drug interaction prediction. In: Proceedings of IJCAI, 2020. 2739–2745
3 Varghese J J, Mushrif S H. Origins of complex solvent effects on chemical reactivity and computational tools to investigate them: A review. React Chemistry Eng, 2019, 4: 165–206
4 Li H, Gao Z, Kang L, et al. TarFisDock: A web server for identifying drug targets with docking approach. Nucleic Acids Res, 2006, 34: 219–224
5 Sagawa T, Kojima R. Reactiont5: A large-scale pre-trained model towards application of limited reaction data. ArXiv:2311.06708
6 Fang J, Zhang S, Wu C, et al. Moltc: Towards molecular relational modeling in language models. ArXiv:2402.03781
7 Shi W, Min S, Lomeli M, et al. In-context pretraining: Language modeling beyond document boundaries. ArXiv:2310.10638
8 Liu Z, Li S, Luo Y, et al. Molca: Molecular graph-language modeling with cross-modal projector and uni-modal adapter. ArXiv:2310.12798
9 Wu F, Zhang Q, Radev D, et al. Pre-training protein models with molecular dynamics simulations for drug binding. 2022. https://assets-eu.researchsquare.com/files/rs-1566483/v1_covered.pdf?c=1654263781
10 Bi X, Chen D, Chen G, et al. Deepseek LLM: Scaling open-source language models with longtermism. ArXiv:2401.02954
11 Gao L, Biderman S R, Black S, et al. The pile: An 800GB dataset of diverse text for language modeling. ArXiv:2101.00027
12 Radford A, Wu J, Child R, et al. Language models are unsupervised multitask learners. 2019. https://api.semanticscholar.org/CorpusID:160025533
13 Hernandez D, Brown T B, Conerly T, et al. Scaling laws and interpretability of learning from repeated data. ArXiv:2205.10487
14 Xie S M, Santurkar S, Ma T, et al. Data selection for language models via importance resampling. ArXiv:2302.03169
15 Gururangan S, Marasović A, Swayamdipta S, et al. Don't stop pretraining: Adapt language models to domains and tasks. ArXiv:2004.10964
16 Xie S M, Pham H, Dong X, et al. Doremi: Optimizing data mixtures speeds up language model pretraining. ArXiv:2305.10429
17 Longpre S, Yauney G, Reif E, et al. A pretrainer's guide to training data: Measuring the effects of data age, domain coverage, quality, & toxicity. ArXiv:2305.13169
18 Biderman S, Schoelkopf H, Anthony Q, et al. Pythia: A suite for analyzing large language models across training and scaling. ArXiv:2304.01373
19 Muennighoff N, Rush A M, Barak B, et al. Scaling data-constrained language models. ArXiv:2305.16264
20 Xue F, Fu Y, Zhou W, et al. To repeat or not to repeat: Insights from scaling LLM under token-crisis. ArXiv:2305.13230
21 Chen Z, Zhang J, Wang S, et al. Do-colm: Dynamic 3D conformation relationships capture with self-adaptive ordering molecular relational modeling in language models. In: Proceedings of the 34th International Joint Conference on Artificial Intelligence, 2025
22 Du W, Zhang S, Cai Z, et al. Molecular merged hypergraph neural network for explainable solvation Gibbs free energy prediction. Research, 2025, 8: 0740
23 Edwards C, Lai T, Ros K, et al. Translation between molecules and natural language. ArXiv:2204.11817

24 Zeng Z, Yao Y, Liu Z, et al. A deep-learning system bridging molecule structure and biomedical text with comprehension comparable to human professionals. Nat Commun, 2022, 13: 862
25 Edwards C, Zhai C, Ji H. Text2mol: Cross-modal molecule retrieval with natural language queries. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2021. 595–607
26 Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks. In: Proceedings of ICLR, 2017
27 Beltagy I, Lo K, Cohan A. Scibert: A pretrained language model for scientific text. ArXiv:1903.10676
28 Liang Y, Zhang R, Zhang L, et al. Drugchat: Towards enabling ChatGPT-like capabilities on drug molecule graphs. ArXiv:1903.10676
29 Tang X, Tran A, Tan J, et al. Mollm: A unified language model for integrating biomedical text with 2d and 3d molecular representations. Bioinformatics, 2024, 40: 357–368
30 Li S, Liu Z, Luo Y, et al. 3d-molm: Towards 3d molecule-text interpretation in language models. In: Proceedings of ICLR, 2024
31 Lee N, Hyun D, Na G S, et al. Conditional graph information bottleneck for molecular relational learning. In: Proceedings of ICML, 2023. 18852–18871
32 Zhong Y, Chen X, Zhao Y, et al. Graph-augmented convolutional networks on drug-drug interactions prediction. ArXiv:1912.03702
33 Zhang D, Xia S, Zhang Y. Accurate prediction of aqueous free solvation energies using 3D atomic feature-based graph neural network with transfer learning. J Chem Inf Model, 2022, 62: 1840–1848
34 Fu T, Xiao C, Sun J. Core: Automatic molecule optimization using copy & refine strategy. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2020. 638–645
35 Zhong Y, Li G, Yang J, et al. Learning motif-based graphs for drug-drug interaction prediction via local-global self-attention. Nat Mach Intell, 2024, 6: 1094–1105
36 Pei Q, Zhang W, Zhu J, et al. Biot5: Enriching cross-modal integration in biology with chemical knowledge and natural language associations. ArXiv:2310.07276
37 Kim S, Chen J, Cheng T, et al. Pubchem 2023 update. Nucleic Acid Res, 2023, 51: 1373–1380
38 Cereto-Massagué A, Ojeda M J, Valls C, et al. Molecular fingerprint similarity search in virtual screening. Methods, 2015, 71: 58–63
39 Bajusz D, Rácz A, Héberger K. Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations? J Cheminformatics, 2015, 7: 1–13
40 Johnson J, Douze M, Jegou H. Billion-scale similarity search with GPUs. IEEE Trans Big Data, 2019, 7: 535–547
41 Douze M, Guzhva A, Deng C, et al. The faiss library. 2024. ArXiv:2401.08281
42 Flood M M. The traveling-salesman problem. Oper Res, 1956, 4: 61–75
43 Padhi A K, Janežič M, Zhang K Y. Molecular dynamics simulations: Principles, methods, and applications in protein conformational dynamics. In: Proceedings of Advances in Protein Molecular and Structural Biology Methods, 2022. 439–454
44 Zhou G, Gao Z, Ding Q, et al. Uni-mol: a universal 3d molecular representation learning framework. In: Proceedings of ICLR, 2023
45 Donsker M D, Varadhan S R S. Asymptotic evaluation of certain Markov process expectations for large time. IV. Comm Pure Appl Math, 1983, 36: 183–212
46 Xu K, Hu W, Leskovec J, et al. How powerful are graph neural networks? In: Proceedings of ICLR, 2019
47 Zhang W, Chen Y, Liu F, et al. Predicting potential drug-drug interactions by integrating chemical, biological, phenotypic and network data. BMC Bioinformat, 2017, 18: 1–12
48 Zitnik M, Sosi R, Maheshwari S, et al. Stanford biomedical network dataset collection. 2018. https://snap.stanford.edu/biodata/
49 Ryu J Y, Kim H U, Lee S Y. Deep learning improves prediction of drug-drug and drug-food interactions. Proc Natl Acad Sci USA, 2018, 115: E4304
50 Tatonetti N P, Ye P P, Daneshjou R, et al. Data-driven prediction of drug effects and interactions. Sci Transl Med, 2012, 4: 125ra31
51 Joung J F, Han M, Jeong M, et al. Experimental database of optical properties of organic compounds. Sci Data, 2020, 7: 295
52 Marenich A V, Kelly C P, Thompson J D, et al. Minnesota solvation database (MNSOL) version 2012. Data Repository for the University of Minnesota, 2020
53 Moine E, Privat R, Sirjean B, et al. Estimation of solvation quantities from experimental thermodynamic data: Development of the comprehensive compsol databank for pure and mixed solutes. J Phys Chem Ref Data, 2017, 46: 033102
54 Grubbs L M, Saifullah M, de la Rosa N E, et al. Mathematical correlations for describing solute transfer into functionalized alkane solvents containing hydroxyl, ether, ester or ketone solvents. Fluid Phase Equilib, 2010, 298: 48–53
55 Vermeire F H, Green W H. Transfer learning for solvation free energies: From quantum chemistry to experiments. Chem Eng J, 2021, 418: 129307
56 Mobley D L, Guthrie J P. FreeSolv: a database of experimental and calculated hydration free energies, with input files. J Comput Aided Mol Des, 2014, 28: 711–720
57 Zhang S, Fang J, Li X, et al. Iterative substructure extraction for molecular relational learning with interactive graph information bottleneck. In: Proceedings of the 13th International Conference on Learning Representations, 2025
58 Deac A, Huang Y H, Veličković P, et al. Drug-drug adverse effect prediction with graph co-attention. ArXiv:1905.00534
59 Lin S, Wang Y, Zhang L, et al. MDF-SA-DDI: predicting drug-drug interaction events based on multi-source drug fusion, multi-source feature fusion and transformer self-attention mechanism. Brief BioInf, 2022, 23: bbab421
60 Li Z, Zhu S, Shao B, et al. DSN-DDI: an accurate and generalized framework for drug-drug interaction prediction by dual-view representation learning. Brief BioInf, 2023, 24: bbac597
61 Yu J, Zhang C, Cheng Y, et al. SolvBERT for solvation free energy and solubility prediction: a demonstration of an NLP model for predicting the properties of molecular complexes. Digital Discov, 2023, 2: 409–421
62 Meng F, Zhang H, Collins-Ramirez J S, et al. Something for nothing: Improved solvation free energy prediction with Δ-learning. Theor Chem Acc, 2023, 142: 106
63 Du W, Zhang S, Wu D, et al. MMGNN: A molecular merged graph neural network for explainable solvation free energy prediction. In: Proceedings of the 33rd International Joint Conference on Artificial Intelligence, 2024. 5808–5816
64 Fang X, Liu L, Lei J, et al. Geometry-enhanced molecular representation learning for property prediction. Nat Mach Intell, 2022, 4: 127–134
65 Rong Y, Bian Y, Xu T, et al. Self-supervised graph transformer on large-scale molecular data. In: Proceedings of NeurIPS, 2020
66 Christofidellis D, Giannone G, Born J, et al. Unifying molecular and textual representations via multi-task language modelling. ArXiv:2301.12586
67 Touvron H, Lavril T, Izacard G, et al. Llama: Open and efficient foundation language models. ArXiv:2302.13971
68 Taylor R, Kardas M, Cucurull G, et al. Galactica: A large language model for science. ArXiv:2211.09085
69 McKenzie I R, Lyzhov A, Pieler M, et al. Inverse scaling: When bigger isn't better. ArXiv:2306.09479
70 Zhou L, Schellaert W, Martínez-Plumed F, et al. Larger and more instructable language models become less reliable. Nature, 2024, 634: 61–68