A Self-feedback Knowledge Elicitation Approach for Chemical Reaction Predictions

Pengfei Liu, Jun Tao, and Zhixiang Ren

Abstract—The task of chemical reaction predictions (CRPs) plays a pivotal role in advancing drug discovery and material science. However, its effectiveness is constrained by the vast and uncertain chemical reaction space and challenges in capturing reaction selectivity, particularly due to existing methods' limitations in exploiting the data's inherent knowledge. To address these challenges, we introduce a data-curated self-feedback knowledge elicitation approach. This method starts from iterative optimization of molecular representations and facilitates the extraction of knowledge on chemical reaction types (RTs). Then, we employ adaptive prompt learning to infuse the prior knowledge into the large language model (LLM). As a result, we achieve significant enhancements: a 14.2% increase in retrosynthesis prediction accuracy, a 74.2% rise in reagent prediction accuracy, and an expansion in the model's capability for handling multi-task chemical reactions. This research offers a novel paradigm for knowledge elicitation in scientific research and showcases the untapped potential of LLMs in CRPs.

Index Terms—Knowledge Elicitation, Prompt Learning, Large Language Model, Chemical Reaction Predictions.

1 Introduction

The applications of CRPs [1] span drug discovery, material science, and synthetic pathway optimization, which have a critical role in advancing various scientific fields. The primary challenges within this domain arise from the vast and uncertain chemical reaction space, coupled with the complexities of reaction selectivity. Moreover, most existing methods fail to harness the intrinsic knowledge within reaction data. Traditional methods in CRPs [2] [3] struggle to navigate the intricate and variable dynamics of chemical reactions, which rely on domain expertise and heuristic strategies. While existing computational methods have strived to predict chemical reactions, they often fall short in handling the inherent complexity and selectivity due to limited datasets and the lack of detailed reaction mechanism guidance. With advancements in artificial intelligence (AI), AI methods [4] [5] [6] have led to notable improvements in the accuracy of CRPs. However, the challenge of generalizing these improvements across diverse chemical reactions persists.

With the emergence of ChatGPT [7] and GPT-4 [8], LLMs have gained attention for their potential in various domains, including science. LLMs contribute a new trend in scientific language modeling (SLM) [9], with models like Galactica [10] focusing on the scientific domain. Smaller pre-trained models such as MolT5 [11] and BioT5 [12] have started to be applied in molecular SLM tasks. Similarly, Text+Chem T5 [13] and InstructMol [14] cover CRP tasks, yet large models still face challenges in interpretability and the demand for extensive training data.

Prompt learning, coupled with LLMs for fine-tuning, has emerged as a standard paradigm, offering a way to integrate domain-specific knowledge into model training [15]. However, static template prompts can lead to rigid guiding patterns in LLMs, potentially impacting their generalizability. The dynamic prompts can tackle the limitations of static templates with the injection of prior knowledge into LLMs. In CRPs, identifying RTs can narrow down the chemical space to be explored. However, the datasets typically lack RT labels, including only reactants and products. While too few categories in annotation methods limit their effectiveness, too many can reduce annotation accuracy. Meanwhile, the synergistic effect of multi-task cooperative learning, treating chemical reactions as a unified domain of molecular knowledge, has yet to be fully leveraged. Therefore, we can conclude several key issues: **(1) How can we balance annotation accuracy and number of RTs in knowledge elicitation by LLMs? (2) Can LLMs perform better through prompt-based knowledge infusion? (3) Can a multi-task collaborative approach improve the performance of LLMs?**

To address these issues, we introduce a prompt-based knowledge elicitation [16] approach that combines knowledge distillation and integration through adaptive prompts, aiming to boost the accuracy of LLMs. The task of CRP is broken down into RT prediction and molecule generation. By applying a self-feedback knowledge elicitation method for high-accuracy annotating RTs and utilizing prompt learning for knowledge infusion into the LLM, we enhance model performance and achieve the synergistic benefits of multi-tasking.

In summary, our main contributions are the following:

- **Self-Feedback Knowledge Elicitation:** We propose a novel knowledge elicitation approach by integrating a self-feedback mechanism with data curation using LLM, enhancing accuracy in CRPs.

- *Pengfei Liu is with the School of Computer Science and Engineering, Sun Yat-sen University, and Peng Cheng Laboratory.*
- *Jun Tao is with the School of Computer Science and Engineering, Sun Yat-sen University.*
- *Zhixiang Ren* is with Peng Cheng Laboratory. E-mail: jason.zhixiang.ren@outlook.com*

**Corresponding author.*

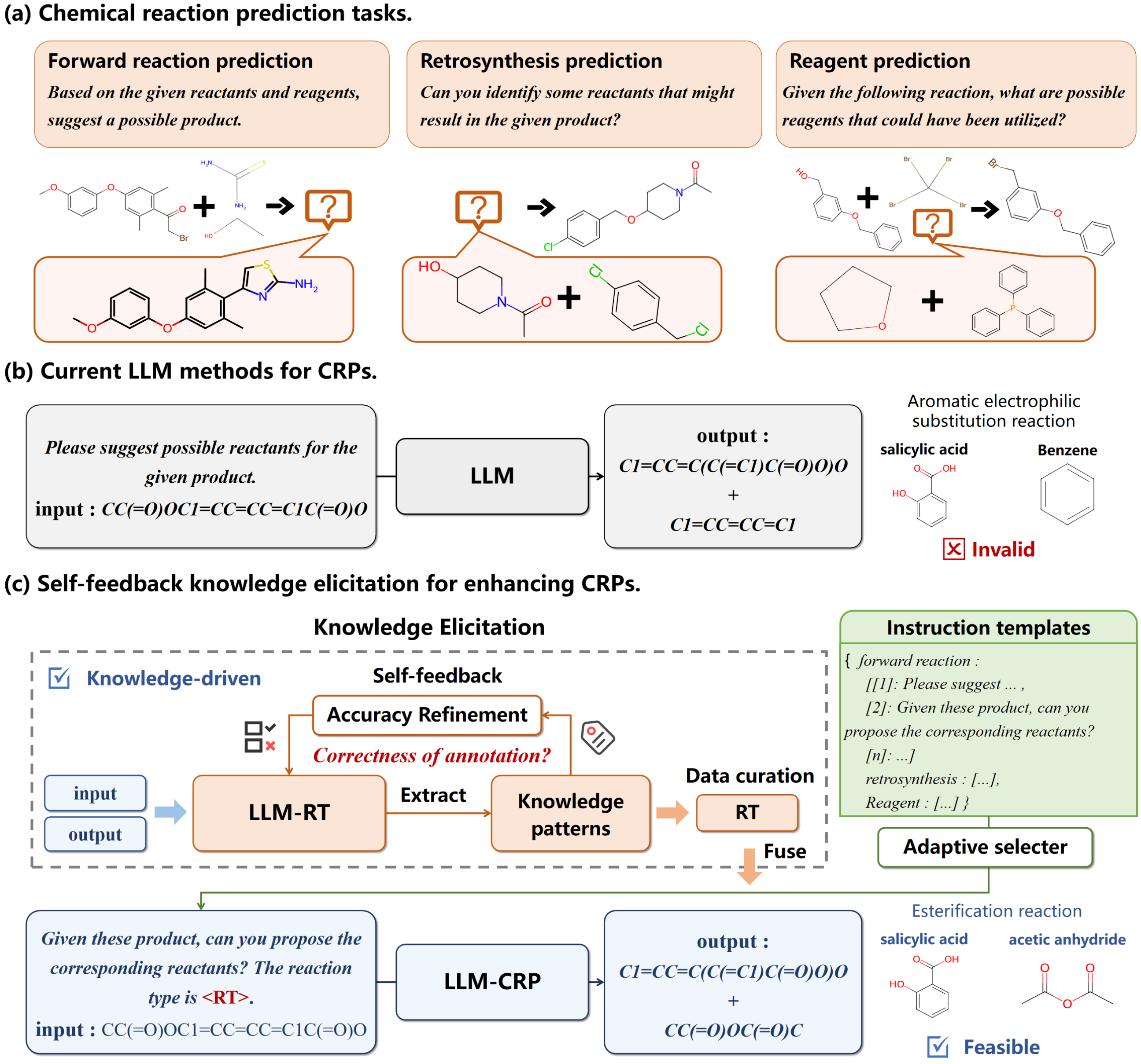
(a) Chemical reaction prediction tasks.
Forward reaction prediction
Based on the given reactants and reagents, suggest a possible product.
Retrosynthesis prediction
Can you identify some reactants that might result in the given product?
Reagent prediction
Given the following reaction, what are possible reagents that could have been utilized?
?
(b) Current LLM methods for CRPs.
Please suggest possible reactants for the given product.
input : CC(=O)OC1=CC=CC=C1C(=O)O
LLM
output : C1=CC=C(C(=C1)C(=O)O)O + C1=CC=CC=C1
Aromatic electrophilic substitution reaction
salicylic acid
Benzene
☒ Invalid
(c) Self-feedback knowledge elicitation for enhancing CRPs.
Knowledge Elicitation
☑ Knowledge-driven
Self-feedback
Accuracy Refinement
Correctness of annotation?
input
output
LLM-RT
Extract
Knowledge patterns
Data curation
RT
Fuse
Instruction templates
{ forward reaction :
[[1]: Please suggest ... ,
[2]: Given these product, can you propose the corresponding reactants?
[n]: ...]
retrosynthesis : [...],
Reagent : [...] }
Adaptive selecter
Given these product, can you propose the corresponding reactants? The reaction type is <RT>.
input : CC(=O)OC1=CC=CC=C1C(=O)O
LLM-CRP
output : C1=CC=C(C(=C1)C(=O)O)O + CC(=O)OC(=O)C
Esterification reaction
salicylic acid
acetic anhydride
☑ Feasible

Fig. 1. **Overview of tasks and approaches**. **(a) Chemical reaction prediction tasks**, showcasing three tasks along with examples. **(b) Current LLM methods for CRPs**, indicating rational predictions but lacking in reactive validity. **(c) Self-feedback knowledge elicitation for enhancing CRPs**, demonstrating the enhancement of CRPs through the refinement of knowledge patterns, notably RTs, utilizing a self-feedback knowledge elicitation technique. Knowledge elicitation serves as a method of data curation for knowledge distillation, where RT is integrated into large language models via adaptive prompt learning, facilitating the planning of reaction pathways in CRPs.

- **Dynamic Prompting for LLMs:** We introduce a dynamic prompt learning to address the limitations of static prompts, achieving a 10% additional increase in the knowledge injection adaptability of LLMs.
- **Synergistic Multi-Task Enhancement:** By injecting prior knowledge, we facilitate a synergistic improvement of 14.9% across reaction prediction tasks.

The rest of this paper is organized as follows. Section 2 offers a review of existing studies, laying the groundwork for our approach. In Section 3, we delve into the methodology, detailing the data, models, and strategies used. Section 4 is dedicated to evaluating our methods and includes an ablation study to highlight key findings. The discussion in Section 5 centers on our annotation approach, the knowledge-learning capabilities of large models, and the benefits of multi-tasking. Finally, Section 6 summarizes our main contributions and suggests future research directions. The data and software can be accessed at https://github.com/AI-HPC-Research-Team/SLM4CRP.

2 Related Work

In this section, we focus on the models for CRPs, the application of LLMs, and the innovative use of prompt learning and knowledge priors [17], laying the groundwork for our proposed approach.

2.1 Chemical Reaction Predictions

As illustrated in Figure 1 (a), CRP tasks involve determining the products or reactants of chemical reactions from given molecules. The key to the CRPs lies in accurately identifying the mechanisms and outcomes involving bond breakage and reformation under a variety of conditions. It makes the predictions of reactions particularly complex due to the vast number of possible reaction mechanisms and products. It requires a deep understanding of chemical knowledge and molecular interactions to forecast the most probable pathways for forward reactions, retrosynthetic routes, and necessary reagents for specific transformations.

For traditional methods, CAMEO [2] leverages detailed heuristics across chemical classes to predict multistep reactions, while EROS [3] utilizes a graph-based rule library enhanced by additional constraints from physical data and kinetic simulations. Despite their sophistication, these traditional methods often struggle with the vast complexity and variability of chemical reactions, leading to limitations in predictive accuracy and scalability.

With the development of AI methods, the ReactionPredictor [4] model narrows down the reaction space through a filtering model and then ranks the prioritized likely reactions. Another method [5] combines fingerprints of reactants and reagents into a reaction fingerprint, used as input to a neural network predicting probabilities across 17 RTs. Additionally, the GTPN [6] model leverages graph neural networks [18] (GNN) to comprehend the molecular graph structures of input reactants and reagents, refining the prediction of correct products through reinforcement learning. The Molecular Transformer [19], based on Transformer [20], treats reaction prediction as a machine translation issue between molecules represented in SMILES (Simplified Molecular Input Line Entry System) [21] strings, further enabling the assessment of prediction uncertainty.

Traditional methods in CRPs rely on heuristics and rule-based systems, often limiting their adaptability and scalability across the diverse landscape of chemical reactions. As for AI approaches, despite their innovative frameworks and predictive power, their challenges include data dependency, model interpretability, and the generalization of predictions to unseen reactions.

2.2 Knowledge Distillation and Elicitation

Knowledge distillation [22] is a machine learning technique in which knowledge is transferred from a larger, more complex model (teacher) to a smaller, simpler model (student). This method is commonly employed for model compression [23], aiming to enhance model performance while adhering to a fixed capacity constraint. It allows the compact student model to mimic the behavior of the larger teacher model, thereby achieving efficiency without significantly compromising the quality of the model. The survey [16] categorizes the pipeline of distilling knowledge from LLMs into two main phases: knowledge elicitation and the distillation algorithm. Moreover, they identify six methods of knowledge elicitation from teacher LLMs. Among these, the data curation approach has gained attention for its focus on producing high-quality and scalable data generation for knowledge distillation purposes. Unlike data augmentation [24], which primarily aims at increasing the quantity of training data, data curation emphasizes constructing a high-quality training dataset.

For instance, InPars [25] leverages the LLM to generate labeled data in a few-shot manner, creating synthetic datasets that enhance performance in information retrieval tasks. Similarly, ZEROGEN [26] employs LLMs to generate unsupervised datasets, upon which a smaller task-specific model is trained, facilitating efficient inference across various Natural Language Processing (NLP) tasks. These data curation techniques can act as variants of knowledge distillation, producing high-quality datasets for CRPs and other domains, thereby enriching the pool of strategies for effective knowledge distillation.

2.3 Large Language Models

Transformer-based models such as BERT [27], GPT [28] and T5 [29], showcasing remarkable capabilities in understanding and generating human language. As the scale of models has grown with data availability, models like Chinchilla [30], LLaMA [31], and GLM [32], which demonstrate an enhanced capacity to process and generate text.

In specific fields, models are increasingly tailored to domain knowledge learning. In molecular science, MolT5 [11], based on the T5 architecture, pioneers tasks in molecular description and text-based molecular design. Text+Chem T5 [13] represents a multi-task language model approach capable of handling a variety of tasks across both chemical and linguistic domains. GIT-Mol [33] introduces an innovative approach by aligning and integrating molecular text, graphs, and images through cross-attention mechanisms and contrastive learning. Meanwhile, BioT5 [12] merges knowledge from the molecular and protein domains, presenting a cross-disciplinary pre-trained model that underscores the potential of LLMs to bridge and enhance research across fields. The MOLGEN [34] model, built on the BART [35] and utilizing SELFIES [36], is capable of generating novel molecules and optimizing molecular structures based on desired properties.

Although LLMs display enhanced generalization capabilities and a stronger understanding of knowledge, as shown in Figure 1(b), they may select incorrect synthetic pathways in CRPs. Despite these advancements, they still face challenges with data scarcity in specialized domains and lack interpretability.

2.4 Prompt-based Knowledge Priors

Prompt learning involves designing input 'prompts' that guide the LLMs to perform specific tasks or generate certain types of responses. This technique leverages the knowledge in pre-trained models, enabling them to apply their understanding of language to new tasks without extensive retraining. The Mol-Instructions [15] dataset facilitates LLM fine-tuning with diverse molecule and protein instructions, including the CRPs datasets of USPTO [37] and USPTO_500MT [38]. InstructMol [14], a multi-modal LLM, employs instruction tuning to correlate molecular structures with textual data, using a dual-phase training approach that smartly leverages limited domain-specific datasets for molecule captioning and CRPs. Following BioT5, BioT5+

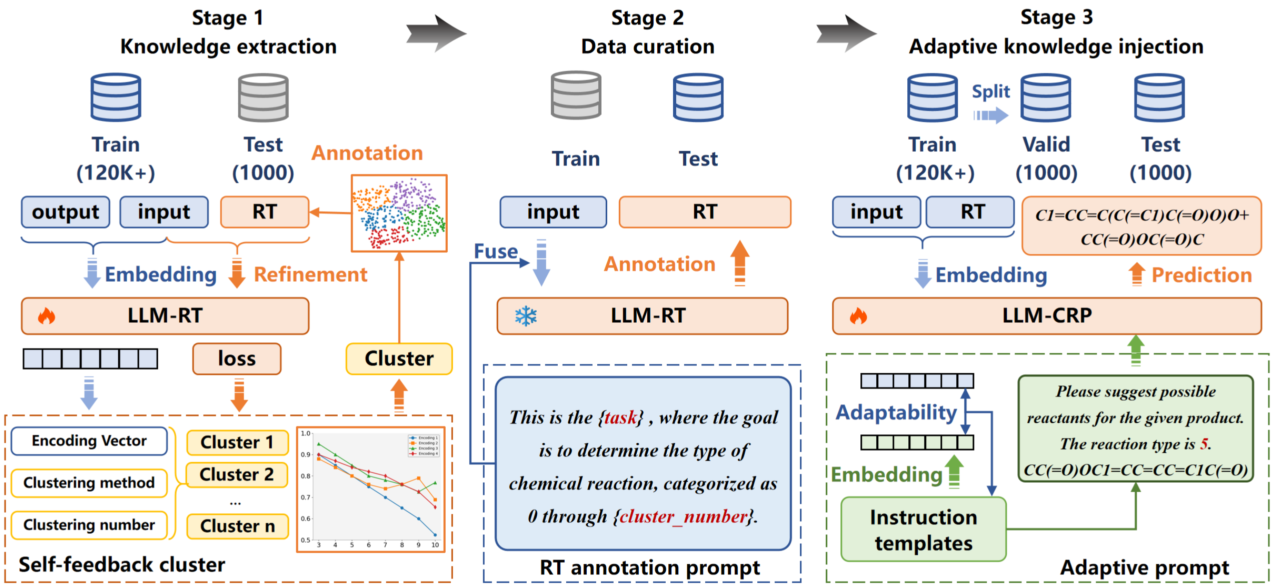

Fig. 2. **Three-stage training scheme of prompt-based knowledge elicitation**. **Knowledge extraction**, the datasets are divided into train, valid, and test sets. The training dataset's inputs and outputs are clustered using LLM-RT embeddings, leading to RT annotations. The annotation accuracy of LLM-RT is refined by iteratively tuning cluster parameters and training with input and RT, aiming to improve precision and identify the best cluster. **Data curation**, the trained LLM-RT annotates the RTs for the validation and testing datasets based on their inputs. **Adaptive knowledge injection**, adaptability is calculated based on the embeddings of inputs and instructions, leading to the selection of adaptive instructions. It is followed by fine-tuning the LLM with prompts that are enhanced with prior knowledge.

[39] extends training with International Union of Pure and Applied Chemistry (IUPAC) [40] names of molecules and broadens its application to additional tasks.

Knowledge priors refer to pre-existing, domain-specific knowledge that can be integrated into AI models to enhance their understanding and prediction capabilities [41]. The knowledge can be fused with the model architecture, training frameworks, or training data to enrich the model's insights and improve its outcomes. In molecular science, knowledge priors, such as details on molecular structures, chemical properties, and reaction mechanisms, significantly boost AI models' predictive accuracy. The KPGT [42] framework exemplifies this by utilizing a graph transformer for molecular graphs with a knowledge-guided pre-training strategy, aiming to understand molecules' structural and semantic knowledge. Similarly, PGMG [43], a pharmacophore-guided deep learning method, innovatively tackles the mapping challenges between pharmacophores and molecules, thereby increasing the diversity of biologically active molecules generated. For prompt-enhanced method, the KANO [44] method introduces a chemical element-oriented knowledge graph to encapsulate fundamental knowledge of elements and functional groups. It employs this graph in a novel molecular contrastive learning approach with functional prompts, effectively leveraging deep domain knowledge throughout the pre-training and fine-tuning stages.

Prompt-based knowledge priors represent an evolution in leveraging domain-specific knowledge within AI models. By crafting prompts that encapsulate molecular knowledge, researchers can direct the focus of LLMs toward SLM problems. This strategy not only heightens the accuracy and pertinence of model predictions but also streamlines the integration of intricate scientific knowledge, reducing the dependency on voluminous training data and augmenting the model's interpretability.

3 Methodology

In this chapter, we present an overview of our approach, which leverages chemical knowledge through prompt learning to enhance the accuracy of CRPs. First, we introduce the foundational data structure and the preparatory steps. Next, we delve into the self-feedback knowledge elicitation process, a pivotal mechanism to unearth knowledge patterns. Finally, we describe how we train our LLM on specific reaction prediction tasks.

3.1 Overview

The approach depicted in Figure 2 unfolds through a three-stage training strategy. Initially, the dataset is divided into training, validation, and testing sets, with an emphasis on knowledge extraction to facilitate RT annotation. This stage involves iteratively refining the selection of the optimal clustering approach and training the LLM-RT, leading to the formation of a self-feedback clustering mechanism. In the data curation phase, the frozen LLM-RT employs prompts alongside inputs to perform RT annotation on the validation and testing sets. Finally, the method incorporates the $prompt_{enhanced}$ for fine-tuning the LLM.

The CRP tasks are classified into three primary categories: forward reactions, reverse reactions, and reagent predictions. Each category can be represented mathematically as follows:

- **Forward Reaction Prediction** aims to predict the products ($\mathcal{P}$) for a given set of reactants ($\mathcal{R}$), formulated as $f_{forward} : \mathcal{R} \rightarrow \mathcal{P}$.

- **Retrosynthesis Prediction** seeks to identify potential reactants ($\mathcal{R}$) from known products ($\mathcal{P}$), represented as $f_{retrosynthesis} : \mathcal{P} \to \mathcal{R}$.
- **Reagent Prediction** focuses on determining the reagents ($\mathcal{G}$) required for the conversion of reactants to products, expressed as $f_{reagent} : (\mathcal{R}, \mathcal{P}) \to \mathcal{G}$.

This scenario can be represented by a more generalized function:

$$f_{general} : (\mathcal{R}, \mathcal{P}) \to (\mathcal{R}', \mathcal{P}', \mathcal{G}) \tag{1}$$

where $\mathcal{R}$ and $\mathcal{P}$ are the sets of reactants and products, respectively, and the function aims to predict $\mathcal{R}'$ (reactants), $\mathcal{P}'$ (products), and $\mathcal{G}$ (reagents) for any given chemical reaction. Let us define the knowledge-driven prompt, denoted as $prompt_{enhanced}$, as the combination of adaptive instructions (IA) and knowledge priors RTs ($\mathcal{K}$):

$$prompt_{enhanced} = IA + \mathcal{K} \tag{2}$$

Incorporating prior knowledge by $prompt_{enhanced}$ enhances the specificity and accuracy of the predictions, modifying the general function to:

$$f_{enhanced} : (\mathcal{R}, \mathcal{P}, prompt_{enhanced}) \to (\mathcal{R}', \mathcal{P}', \mathcal{G}) \tag{3}$$

In this function, the $prompt_{enhanced}$ serves to guide the prediction process by incorporating both adaptive instructions and the specified knowledge priors, leading to the prediction of reactants $\mathcal{R}'$, products $\mathcal{P}'$, and the necessary reagents $\mathcal{G}$.

Distinct from existing methods, our approach integrates RT knowledge priors with LLMs through adaptive prompt learning and self-feedback knowledge elicitation techniques. It addresses the scarcity of RT information in real-world datasets, and the dynamic prompts prevent rigid pattern guiding in LLMs, offering a solution to enhancing prediction accuracy.

3.2 Data

In this study, we utilize the 'Molecule-oriented instructions' from the Mol-Instructions [15] dataset, which encompasses three chemical reaction tasks. Our data preparation involves converting the 'input' into SMILES format, resulting in dataset D. This dataset's 'instruction' components are sorted by different task types and integrated into our instruction template library, aiding the infusion of domain-specific knowledge into our models.

Data preprocessing: Before conducting our experiments, we engage in a thorough data preprocessing regimen to safeguard the dataset's integrity and uniformity. The primary steps encompass transforming all molecular representations into SMILES format to standardize the molecular data, thereby ensuring compatibility across our experiments. We rigorously clean the data by removing entries that fail the SMILES conversion process, thus mitigating potential inconsistencies and errors in later analysis stages.

Data split: We follow the original test set partitioning scheme of the dataset. However, in the 'knowledge extraction' stage in Figure 2, the training subset D_{train} is randomly divided in a 98:1:1 ratio to obtain D'_{train}, D'_{valid}, and D'_{test}. This division facilitates the training of the LLM-RT predictor, which subsequently annotates the testing set D_{test}. In the 'Adaptive knowledge injection' stage, the validation set D_{valid} is split from D_{train}.

3.3 Knowledge Extraction and Data Curation

This subsection explains the process of self-feedback knowledge elicitation in CRP problems and describes the method of data curation for instruction-tuning dataset.

Knowledge annotation Knowledge extraction from the LLM-RT begins by embedding inputs and outputs of the training dataset D_{train}, followed by clustering into corresponding clusters, which are annotated as the RTs. For the vector encoding of input and output embeddings, we consider four alternative methods: directly using the output vector ($output_{vec}$), subtracting input vector from output vector ($output_{vec}$ - $input_{vec}$), concatenating input and output into a vector ($concat(input_{vec}, output_{vec})$, and the dot product of output and input vectors ($output_{vec} \cdot input_{vec}$).

The selection of our clustering method and number is crucial to balance the accuracy and diversity of RT annotations. As the number of clusters increases, accurately annotating D_{test} becomes more challenging. There are several common fundamental types of reactions, including but not limited to synthesis reactions, decomposition reactions, single-replacement reactions, and double-replacement reactions. While these basic categories can be subdivided, over-detailed classification may not be conducive to generalization and efficiency for an effective prediction model. Moreover, as the number of clusters increases, the accuracy of unsupervised labeling is likely to decrease. Furthermore, a high number of clusters may complicate the interpretability of the model, making it harder for users to understand and trust the model's predictions. Considering these factors, a recommended number of clusters would be between 3 to 12 [45]. This range should adequately cover the main types of chemical reactions while avoiding the pitfalls of over-segmentation, such as increased model complexity or reduced generalizability. Furthermore, we opt for the k-means [46] algorithm as our clustering method, given its efficiency and effectiveness in grouping data into cohesive, distinct clusters that reflect underlying patterns within the RTs.

$$E_n = Embedding(input_{D_{train}}, output_{D_{train}}) \tag{4}$$

$$R_n = Cluster(E_n) \tag{5}$$

$$Acc_n = ST(R_n, input_{D'_{train}}, input_{D'_{valid}}, input_{D'_{test}}) \tag{6}$$

The annotation results R_n in the n^{th} iteration, where the training dataset $D_{train,n}$ is encoded and clustered. E_n denotes the encoding function by LLM-RT, and $Cluster$ represents the clustering operation. The annotation accuracy Acc_n achieved after conducting the supervised training (ST) using the annotation results R_n and the training dataset $D_{train,n}$ from the n^{th} iteration.

Self-feedback cluster: After a round of RT annotation, we conduct supervised training of the LLM-RT model against inputs and RTs, adjusting the model weights to refine annotation accuracy continually. After a training round, RTs for D_{train} are re-annotated, thus iteratively optimizing

Algorithm 1 Self-Feedback Knowledge Elicitation Process

```
Initialize the LLM-RT model for encoding methods $encodings$, set cluster method and number range, and prepare instruction templates library.
for each encoding $e$ in $encodings$ do
    for $number = 3$ to $12$ do
        Set cluster number $N = number$, encoding method $E = e$.
        Embed inputs and outputs from $D_{train}$ using encoding method $E$.
        Perform clustering on the embedded data to identify RTs.
        Annotate $D_{train}$ with the identified RTs.
        Split $D_{train}$ to $D'_{train}$, $D'_{valid}$ and $D'_{test}$
        Train the LLM-RT model on $D'$ using the RT annotations.
        Evaluate and record the annotation accuracy.
    end for
    Evaluate and record the overall best annotation accuracy for encoding $e$.
end for
Freeze the LLM-RT model after identifying the optimal $E$ and $N$.
Use the optimized LLM-RT model to annotate RTs in $D_{\text{test}}$.
```

the LLM-RT training through a self-feedback mechanism, forming an effective self-feedback clustering.

$$Optimal(E_{best}, N_{best}) = SF(Acc_n, E_n, Cluster, N) \quad (7)$$

$$RT_{D_{train}} = Cluster_{N_{best}}(E_{best}) \quad (8)$$

The self-feedback (SF) process for selecting the optimal encoding method E and the number of clusters N to maximize the annotation accuracy Acc_n, and $Optimal$ indicates the chosen optimal encoding method and cluster number after the iteration process concludes. Then, the $RT_{D_{train}}$ annotated for the training dataset D_{train}, derived from clustering the dataset with the best embedding method E_{best} into N_{best} clusters through $Cluster_{N_{best}}$. The selection process aims to balance the annotation accuracy and the number of RTs, ensuring optimal categorization.

> **RT annotation prompt:**
> This is the $\{task\}$ reaction prediction task, where the goal is to determine the type of chemical reaction based on the given compounds, categorized as 0 through $\{cluster\_number\}$.
> input: ...

Data curation: The RT annotation prompt is employed to guide the LLM-RT model in annotating the RT during the ST and Annotate process. The term 'task' in the prompt can be dynamically substituted with specific tasks such as forward reaction prediction, reverse reaction synthesis, or reagent prediction, adapting the prompt to various contexts of CRPs. Meanwhile, 'cluster_number' corresponds to the number of clusters N identified in the Self-feedback Cluster process. Utilizing the trained and frozen LLM-RT, we annotate the RTs for the validation and testing datasets using RT annotation prompts and input information. The RT performs annotation on the input data of D_{test} using the $Annotate$ process facilitated by the LLM-RT.

$$RT_{D_{test}} = Annotate(input_{D_{test}}) \quad (9)$$

3.4 Adaptive Knowledge Injection

The application of prompt learning to infuse extracted knowledge priors into our models demonstrates how this approach boosts the predictive accuracy of chemical reactions.

> **Instruction Templates:**
> *forward:*
>
> - "Please suggest a potential product based on the given reactants and reagents."
> - "Please provide a feasible product that could be formed using the given reactants and reagents."
> - "Based on the given reactants and reagents, what product could potentially be produced?"
> - . . .
>
> *retrosynthesis:*
>
> - "Provided the product below, propose some possible reactants that could have been used in the reaction."
> - "Please suggest potential reactants used in the synthesis of the provided product."
> - "Given these product, can you propose the corresponding reactants?"
> - . . .
>
> *reagent:*
>
> - "Based on the given chemical reaction, can you propose some reagents that might have been utilized?"
> - "Can you provide potential reagents for the following chemical reaction?"
> - "Please suggest some possible reagents that could have been used in the following chemical reaction."
> - . . .

Adaptive instruction: To address the constraints of static templates and improve model generalization, we introduce an adaptive selector for template selection. We establish an instruction template library, allocating 12 distinct templates per task, for a total of 36 templates. During the embedding process, the $input_i$ from dataset D and all corresponding templates from the task-specific list in the instruction template library are embedded via the LLM-CRP model. The adaptive selector then evaluates adaptability through vector differences between the input embedding and each template embedding within the library. For each batch, this process entails matching a single input with multiple instructions to determine the best fit based on adaptability scores. The

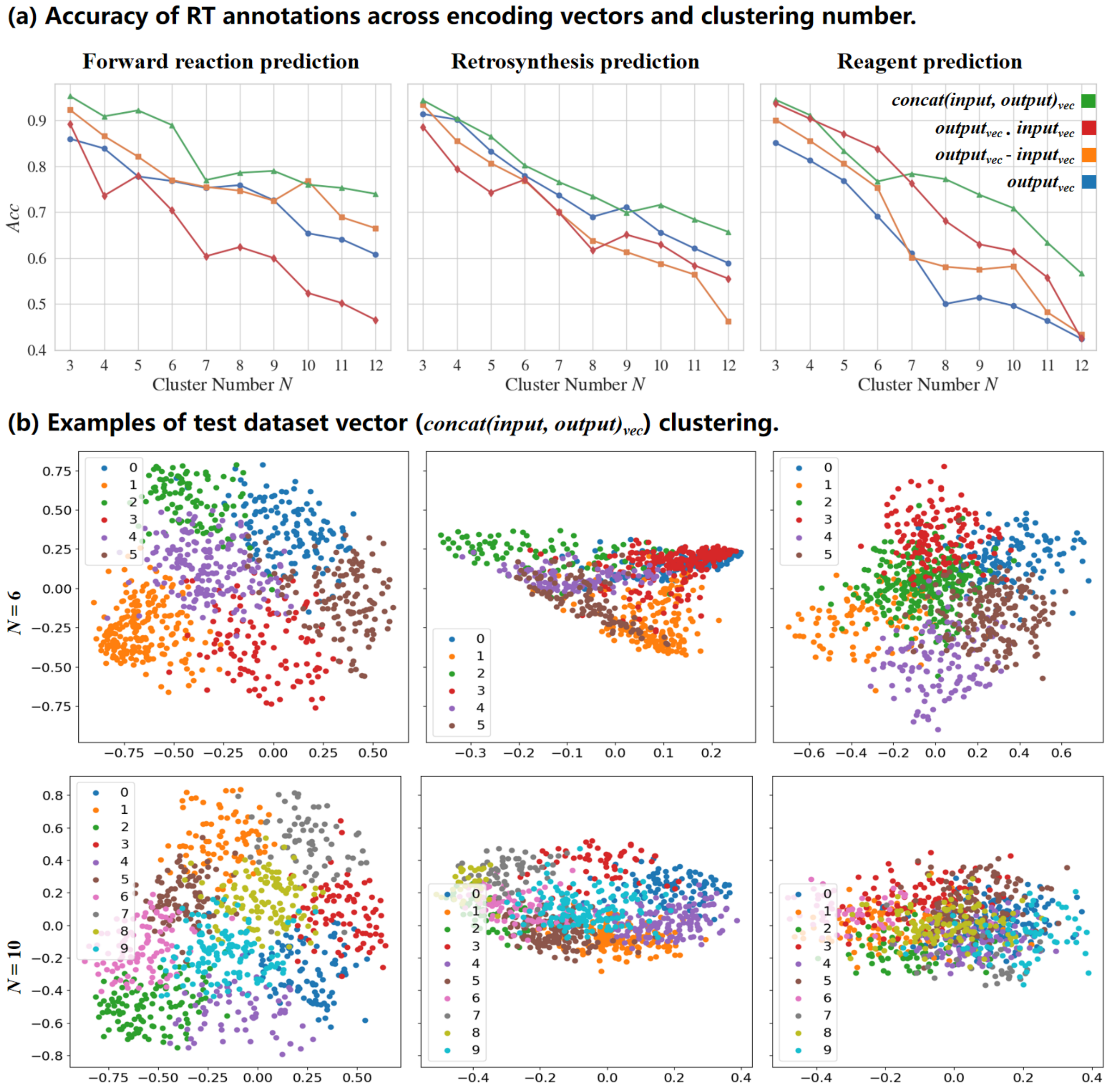

Fig. 3. **Performance of encoding vector self-feedback annotation and clustering**. **(a) Accuracy of RT annotations across encoding vectors and clustering number**, we compare the annotation accuracy Acc among four encoding methods alongside reasonable Cluster Numbers N. The results indicate that the encoding method using $(concat(input, output)_{vec})$ yields the best performance. **(b) The test dataset vector ($concat(input, output)_{vec}$) clustering**, with the N set to 6 and N set to 10, test dataset vectors are reduced to two dimensions via a linear layer to display the clustering outcome.

template that exhibits the highest adaptability with the input is chosen to facilitate precise knowledge injection.

$$Adaptability_{i,j} = -\|Emb_{input} - Emb_{instruction}\|_2 \tag{10}$$

$$IA_i = \arg\min_j(Adaptability_{i,j}) \tag{11}$$

The Emb_{input} is the embedding of the i^{th} input, transforming it into a vector representation that captures its semantic properties within the model's learned feature space. Similarly, $Emb_{instruction}$ calculates the embedding of the j^{th} instruction template. The $Adaptability_{i,j}$ quantifies the similarity score between the i^{th} input and the j^{th} instruction template by maximizing the negative Euclidean distance, thus indicating higher relevance when the value is larger. The adaptive selector then determines the most suitable instruction for the i^{th} input by selecting the template that maximizes the $Adaptability_{i,j}$ score. This process effectively identifies the instruction that best aligns with the input, optimizing the knowledge injection based on the adaptive selector's analysis.

Enhanced prompt: The selected adaptive instruction is combined with the corresponding RT for the current input to form an $prompt_{enhanced}$. This enhanced prompt is then amalgamated with the $input$ and injected into LLM-CRP, aiming to refine the model's response to the input based on

TABLE 1

Performance comparison of various models on the reactions task. The results for Text+Chem T5 and nach0 are taken from their respective publications, whereas T5, MolT5, and Text+Chem T5 (finetune) represent performance after finetuning on a specific dataset. The abbreviation RT stands for reaction type, N denotes the number of clusters, and IA signifies Instruction-adaptive. Using Text+Chem T5 (finetune) as a baseline, ours(RT+IA, N=10) shows improved EM_{score} performance, particularly in retrosynthesis and reagent prediction tasks.

Task	Model	$Bleu_{score}$	$Meteor_{score}$	EM_{score}	$Similarity_{score}$	$Validity_{score}$	$improve$
Forward	Text+Chem T5	—	—	0.594	—	—	—
	nach0	—	—	0.890	—	—	—
	T5	0.986	0.989	0.926	0.984	0.992	—
	MolT5	0.986	0.988	0.897	0.978	0.992	—
	Text+Chem T5 (finetune)	0.988	0.991	0.932	**0.985**	**0.999**	—
	ours(RT,N=10)	**0.991**	0.991	**0.937**	0.984	**0.997**	0.5%
	ours(RT+IA, N=10)	**0.991**	**0.993**	**0.945**	**0.986**	**0.997**	1.4%
	ours(RT+IA,N=6)	**0.989**	**0.992**	0.930	0.984	0.995	-0.2%
Retrosynthesis	Text+Chem T5	—	—	0.372	—	—	—
	nach0	—	—	0.390	—	—	—
	T5	0.920	0.921	0.649	0.855	0.989	—
	MolT5	0.918	0.920	0.637	0.846	0.987	—
	Text+Chem T5 (finetune)	0.926	0.929	0.663	0.858	**0.997**	—
	ours(RT,N=10)	**0.941**	**0.947**	**0.749**	**0.895**	0.996	13.0%
	ours(RT+IA, N=10)	**0.944**	**0.950**	**0.757**	**0.905**	0.994	14.2%
	ours(RT+IA,N=6)	0.920	0.921	0.654	0.848	**0.998**	-1.4%
Reagent	nach0	—	—	0.140	—	—	—
	T5	0.506	0.654	0.168	0.548	0.998	—
	MolT5	0.515	0.660	0.178	0.559	0.997	—
	Text+Chem T5 (finetune)	0.482	0.657	0.163	0.571	0.996	—
	ours(RT,N=10)	**0.589**	**0.728**	**0.273**	**0.640**	**0.999**	67.4%
	ours(RT+IA, N=10)	**0.617**	**0.744**	**0.284**	**0.649**	**1.000**	74.2%
	ours(RT+IA,N=6)	0.499	0.665	0.175	0.587	**0.999**	7.4%

the tailored guidance.

$$prompt_{enhanced\_i} = fuse(IA_i, RT) \tag{12}$$

$$(input_i, prompt_{enhanced\_i}) \rightarrow (\mathcal{R}', \mathcal{P}', \mathcal{G}) \tag{13}$$

The $prompt_{enhanced\_i}$ represents the process of creating an enhanced prompt for the i^{th} input by fusing the adaptively selected instruction $instruction_{adaptive\_i}$ with the corresponding RT. This fusion process ($fuse$) generates a $prompt_{enhanced}$ that is specifically tailored to both the context of the input and the instructional guidance deemed most appropriate by the adaptive selection mechanism. Subsequently, the pair $(input_i, prompt_{enhanced\_i})$ is fed into LLM-CRP. The model's output, represented as $(\mathcal{R}', \mathcal{P}', \mathcal{G})$, encompasses the predictions of reactants $\mathcal{R}'$, products $\mathcal{P}'$, and the necessary reagents $\mathcal{G}$.

4 Evaluation and Results

In this chapter, we subject our proposed methodologies to a rigorous evaluation aimed at addressing three fundamental research questions that guide our investigation. For KI1, we set the number of clusters to range from 3 to 12 and tested four different embedding techniques, selecting the optimal clustering number and encoding method based on annotation accuracy. Regarding KI2, we infuse the RT into the LLM using prompts and assess the effectiveness of adaptive prompt learning. Finally, for KI3, we integrate chemical reaction data to perform comprehensive fine-tuning, achieving results from multi-task training.

- KI1: How can we balance annotation accuracy and number of RTs in knowledge elicitation by LLMs?
- KI2: Can LLMs perform better through prompt-based knowledge infusion?
- KI3: Can a multi-task collaborative approach improve the performance of LLMs?

4.1 Experimental Setup

This subsection provides an overview of the training configurations and evaluation metrics. We detail the model settings and hyperparameter values in model training. In the knowledge extraction phase of our RT annotation experiments, we employ multi-class accuracy (Acc) as the primary evaluation metric. This metric measures the proportion of correctly identified RTs among all predictions, providing a straightforward assessment of the model's performance in categorizing chemical reactions into their correct types.

Training settings: Training employs Tesla V100-SXM2-32GB GPUs with CUDA 11.7 and PyTorch 2.0.0, leveraging AdamW for optimization. Batch sizes vary from 24 to 48, with the patience of 2 epochs for early stopping to curb overfitting. Training spans up to 40 epochs, using adaptive learning rates between 1e-4 and 1e-3 to finetune speed and stability.

Evaluation metrics: During the knowledge injection phase for CRPs, our evaluation strategy is more comprehensive, incorporating a blend of NLP evaluation metrics and compound generation assessment metrics. We include BLEU scores ($Bleu_{score}$), gauging the linguistic similarity between the generated text and reference sequences, and METEOR scores ($Meteor_{score}$), offering a more nuanced evaluation by considering sentence structure. Furthermore, to assess the chemical relevance and accuracy of the generated compounds, we introduce a similarity

TABLE 2
Multi-task performance summary. T5, MolT5, and Text+Chem T5 models are finetuned with aggregated data from three tasks. The average single-task results of Text+Chem T5 are compiled as Text+Chem T5 (avg-tasks), serving as the baseline. The integrated model with $prompt_{enhanced}$ exhibits a **14.9%** improvement over this baseline. Joint training on multiple tasks tends to decrease performance, yet there is a **17.8%** enhancement over Text+Chem T5 (finetune), highlighting the role of RT and IA in multi-task collaboration.

Model	$Bleu_{score}$	$Meteor_{score}$	EM_{score}	$Similarity_{score}$	$Validity_{score}$	$improve$
T5	0.825	0.854	0.556	0.790	0.992	—
MolT5	**0.837**	**0.859**	**0.586**	0.797	0.996	—
Text+Chem T5 (finetune)	0.822	0.857	0.572	0.797	0.995	—
Text+Chem T5 (avg-tasks)	0.799	**0.859**	**0.586**	**0.805**	**0.997**	—
ours(RT+IA,N=10)	**0.879**	**0.901**	**0.674**	**0.854**	**0.998**	14.9%

metric ($Similarity_{score}$), quantifying the resemblance between generated and target compounds. The validity metric ($Validity_{score}$) ensures that every generated compound is chemically valid. The exact match score (EM_{score}) disregards the sequence in which compounds are generated, focusing on the presence of correct chemical entities. These metrics provide a comprehensive view of the model's capability in CRPs.

4.2 Knowledge Elicitation

To address KI1, this section starts by outlining the objectives of the analysis, emphasizing the importance of selecting optimal encoding vectors and the number of clusters N for RT annotation accuracy. We introduce four encoding methods evaluated in the study: direct output vector, output minus input vector, concatenated input-output vector, and the dot product of input and output vectors. These vectors are encoded using LLM-RT Text+Chem T5. Then, we explain the rationale behind exploring different cluster numbers, highlighting the hypothesis that the choice of encoding and clustering can significantly impact the annotation accuracy Acc.

Encoding methods: We delve into a comparative analysis of Acc using various encoding vectors across different task types, with a focus on a range of N. As illustrated in Figure 3 (a), Acc gradually declines within the adaptable range of N from 3 to 12. The findings underscore that the concatenated input-output vector ($concat(input, output)_{vec}$) consistently achieves the highest annotation accuracy among different tasks, maintaining over 70% accuracy even when N is set to 10. It highlights the concatenated vector's capability to capture the nuances of chemical RTs.

Clustering visualization and implications: We examine the clustering outcomes for the test dataset, focusing on the optimal encoding method ($concat(input, output)_{vec}$) for selected cluster numbers, $N = 6$ and $N = 10$, with visualizations depicted in Figure 3 (b). The process involves reducing the high-dimensional encoded vectors to two dimensions for visualization, enabling us to observe the cluster distributions. These visualizations reveal that chemical reactions for each task display distinct knowledge patterns. Although the direct implications of these tasks remain unspecified, the identified patterns serve as crucial prompt information for generating reaction content.

4.3 Knowledge Injection

This subsection assesses the effectiveness of integrating knowledge priors via prompt learning, directly responding to KI2 by comparing the predictive advantages gained through our knowledge-infused model against baseline approaches. In our study, we utilize Text+Chem T5 as the LLM-CRP. The adaptive selector chooses appropriate adaptive instructions, which are then fused with RT to create a prompt-enhanced input. Subsequently, the input is fed into the LLM-CRP for training and testing. The results are shown in Table 1.

RT integration: Selecting an optimal N and integrating RT leads to significant performance enhancement, especially in retrosynthesis and reagent prediction, with improvements of 14.2% and 74.2%.

Instruction adaptation: This effect is especially pronounced in retrosynthesis and reagent prediction tasks, with almost a 10% additional increase (13.0% → 14.2% and 67.1% → 74.2%)

Cluster number: When the cluster number N is set too low, despite achieving high standard accuracy rates, the potential benefits of RT knowledge injection may not be fully realized, and could inadvertently result in a decline in model performance.

4.4 Multi-Task Reaction Prediction

To tackle KI3, we evaluate the performance of our model on multi-task reaction prediction, highlighting how prompt-based knowledge injection influences the model's ability to accurately predict various chemical reaction tasks. In our study, we amalgamate datasets from three distinct tasks for fine-tuning models, including T5, MolT5, and Text+Chem T5. Assuming each task is trained separately, we can establish an optimal performance baseline for each, denoted as Text+Chem T5 (avg-tasks). The outcomes of this experiment are detailed in Table 2.

The side effects of multi-tasking: Based on the experimental results of Text+Chem T5 (avg-tasks) and Text+Chem T5 (finetune), we observe that direct integration and training across multiple tasks can lead to conflicting gradient updates and task interference, ultimately degrading the model's performance. This phenomenon, often referred to as negative transfer, occurs when the optimization for one task adversely affects the learning of another, resulting in suboptimal performance across the board.

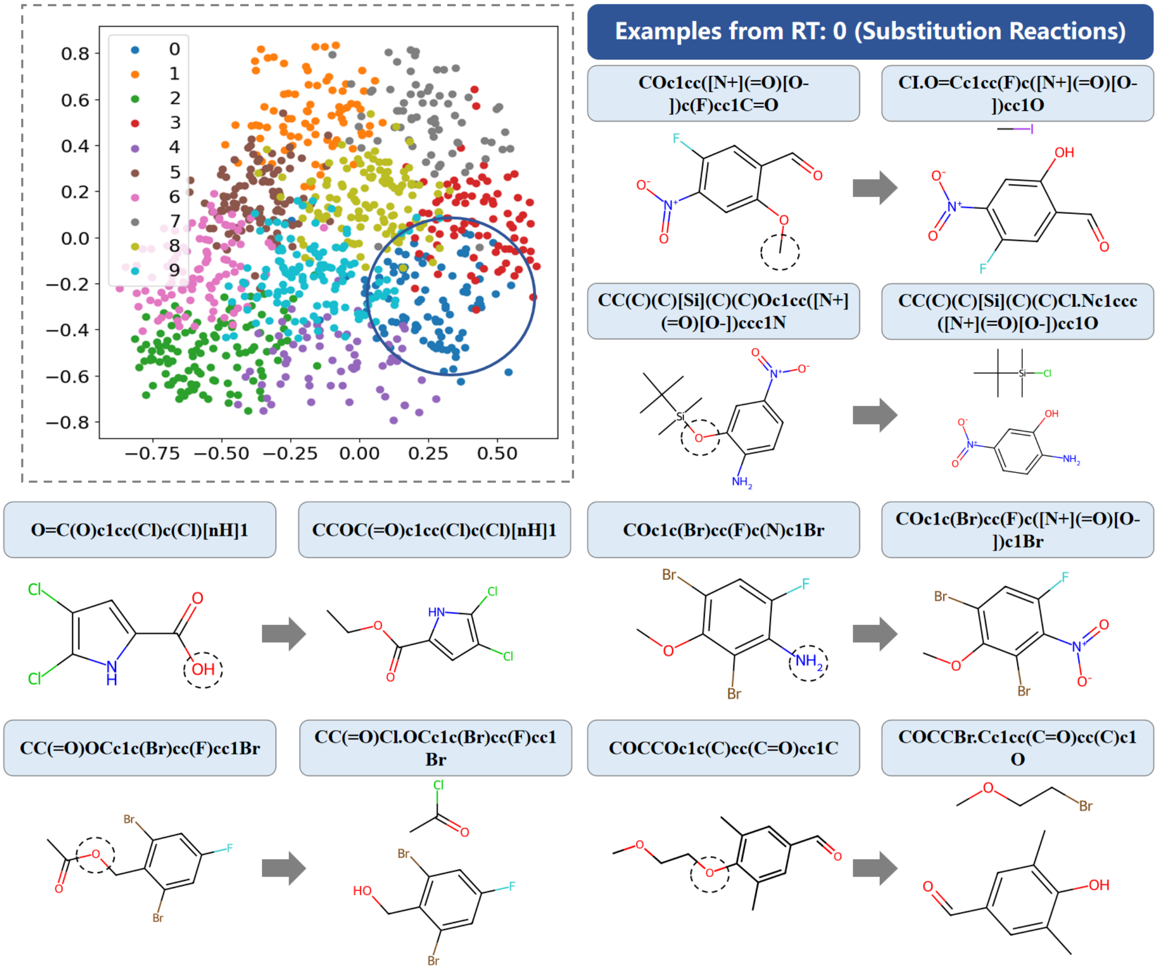
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−0.75 −0.50 −0.25 0.00 0.25 0.50
0
1
2
3
4
5
6
7
8
9
Examples from RT: 0 (Substitution Reactions)
COc1cc([N+](=O)[O-])c(F)cc1C=O
CI.O=Cc1cc(F)c([N+](=O)[O-])cc1O
CC(C)(C)[Si](C)(C)Oc1cc([N+](=O)[O-])ccc1N
CC(C)(C)[Si](C)(C)Cl.Nc1ccc([N+](=O)[O-])cc1O
O=C(O)c1cc(Cl)c(Cl)[nH]1
CCOC(=O)c1cc(Cl)c(Cl)[nH]1
COc1c(Br)cc(F)c(N)c1Br
COc1c(Br)cc(F)c([N+](=O)[O-])c1Br
CC(=O)OCc1c(Br)cc(F)cc1Br
CC(=O)Cl.OCc1c(Br)cc(F)cc1Br
COCCOc1c(C)cc(C=O)cc1C
COCCBr.Cc1cc(C=O)cc(C)c1O

Fig. 4. **Case studies of RT annotation**. To validate the practical significance of RT annotation, we filter through the $concat(input, output)_{vec}$ vector with $N = 10$ labeled results, focusing on samples with an RT label of 0. The molecules in these instances transform simple atomic substitutions. This analysis verifies the predominance of substitution reactions within these cases, demonstrating the real-world relevance of our RT annotation method.

Synergy of prompt-enhanced learning: The Prompt-Enhanced approach demonstrates a synergistic effect across multiple tasks, not only counteracting the side effects but also fostering synergy. This approach results in a significant performance boost of 14.9% over Text+Chem T5 (avg-tasks) due to improved task-specific adaptation and focused learning. By guiding the model with task-specific prompts, we effectively mitigate the issues of task interference, enabling the model to leverage shared knowledge bases while honing in on the nuances of each task. The enhanced task formulation provided by the prompts leads to more effective learning strategies and superior overall performance.

5 Discussion

In our exploration, we've highlighted the innovative approach to RT annotation as a solution to the pervasive challenge of data scarcity in real-world scenarios. Our methodology showcases the LLM's inherent ability to internalize and utilize latent knowledge, asserting the necessity of precise guidance to unlock its full potential. Further, our analysis extends to the multi-task benefits derived from this guided learning process. We aim to illuminate the critical insights and limitations encountered throughout our study.

RT annotation significance: From Table 1 in Chapter 4, the practical effect of RT Annotation injection into language models is evident. The model's performance can be improved with higher cluster numbers N, but this also introduces challenges with annotation accuracy. Thus, finding a balance between annotation accuracy and the quantity of N emerges as a focal point of this part. More importantly, we scrutinize whether RTs annotated through self-feedback knowledge elicitation correspond to actual chemical RTs. This alignment between automated annotations and human-understandable concepts can significantly propel the advancement of interpretability in LLMs. Figure 4 presents randomly selected instances of reactions with RT annotations labeled as 0, with the reaction sites indicated. Most of these reactions are identified as substitution reactions, which confirms the practical significance of the knowledge patterns that our knowledge elicitation methodology extracts.

Knowledge learning in LLMs: LLMs might intrinsically possess the capability to predict RTs. The slight improvements in forward reaction predictions illuminate this ability of LLMs. LLMs can also grasp an understanding of reaction mechanisms without explicit instruction. By deconstructing the problem, we allow the models to gain prior knowledge, facilitating the planning of reaction pathways and simplifying the text generation task into more straightforward classification issues and less complex text generation tasks. Consequently, this can enhance the overall task performance while also increasing the complexity of the process. This approach to problem decomposition is also potentially applicable across various scientific domains, such as molecule design and lead optimization, suggesting a broad utility of this methodology in advancing research and understanding in multiple fields.

Knowledge-enhanced multi-task synergy: We delve into the mechanisms underlying the significant uptick in model accuracy for multi-task learning facilitated by the use of enhanced prompts and knowledge injection. The integration of contextually rich prompts and targeted knowledge snippets acts as a catalyst, fine-tuning the model's focus and understanding of each task. This approach not only amplifies task-specific performance but also harmonizes the learning process across disparate tasks. The prior knowledge acts as an anchor, grounding the model's learning process in real-world phenomena and relationships, thereby reducing the ambiguity inherent in complex tasks. For example, understanding the relationship between molecular structure and pharmacological activity in one task can enhance the model's ability to predict drug toxicity in another, as both tasks share underlying chemical knowledge. Furthermore, this synergy underscores the potential of structured knowledge and task-specific prompts in augmenting the intrinsic multi-tasking capabilities of LLMs.

In future research, the adaptive algorithms can accurately determine the optimal number and encoding strategies for knowledge partners, thus avoiding the inefficient trial-and-error approach. Exploring the potential of dynamic prompts that not only derive from a broader, more randomized pool but also retain the ability to guide specific tasks promises to improve model performance. While the extraction and injection of RTs have enhanced interpretability to a degree, they fall short of revealing the model's exploration within the chemical space. The development of tools for knowledge visualization and tracking would enable the pinpointing of how RTs guide the text generation process, underutilizing the potential of LLMs.

6 Conclusion

In this study, we reconceptualize the task at hand as SLM and pioneer a data-curated, self-feedback knowledge elicitation method to identify knowledge partners, specifically RTs. We then employ dynamic prompt learning to integrate this prior knowledge into LLMs, thereby enhancing accuracy in CRPs and across multiple-task CRPs. This research sets a novel paradigm for knowledge elicitation within scientific domains and for the integration of knowledge priors, laying foundational groundwork for the advancement of SLM.

Acknowledgments

This work is supported by grants from the National Natural Science Foundation of China (62172456 and 62372484), the Major Key Project of PCL PCL2021A13 and Peng Cheng Cloud-Brain.

References

[1] S. Shilpa, G. Kashyap, and R. B. Sunoj, "Recent applications of machine learning in molecular property and chemical reaction outcome predictions," *The Journal of Physical Chemistry A*, vol. 127, no. 40, pp. 8253–8271, 2023.

[2] W. L. Jorgensen, E. R. Laird, A. J. Gushurst, J. M. Fleischer, S. A. Gothe, H. E. Helson, G. D. Paderes, and S. Sinclair, "Cameo: a program for the logical prediction of the products of organic reactions," *Pure and Applied Chemistry*, vol. 62, no. 10, pp. 1921–1932, 1990.

[3] R. Höllering, J. Gasteiger, L. Steinhauer, K.-P. Schulz, and A. Herwig, "Simulation of organic reactions: from the degradation of chemicals to combinatorial synthesis," *Journal of chemical information and computer sciences*, vol. 40, no. 2, pp. 482–494, 2000.

[4] M. A. Kayala and P. Baldi, "Reactionpredictor: prediction of complex chemical reactions at the mechanistic level using machine learning," *Journal of chemical information and modeling*, vol. 52, no. 10, pp. 2526–2540, 2012.

[5] J. N. Wei, D. Duvenaud, and A. Aspuru-Guzik, "Neural networks for the prediction of organic chemistry reactions," *ACS central science*, vol. 2, no. 10, pp. 725–732, 2016.

[6] K. Do, T. Tran, and S. Venkatesh, "Graph transformation policy network for chemical reaction prediction," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 750–760.

[7] J. Schulman, B. Zoph, C. Kim, J. Hilton, J. Menick, J. Weng, J. F. C. Uribe, L. Fedus, L. Metz, M. Pokorny *et al.*, "Chatgpt: Optimizing language models for dialogue," *OpenAI blog*, 2022.

[8] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[9] P. Liu, J. Tao, and Z. Ren, "Scientific language modeling: A quantitative review of large language models in molecular science," *arXiv preprint arXiv:2402.04119*, 2024.

[10] R. Taylor, M. Kardas, G. Cucurull, T. Scialom, A. Hartshorn, E. Saravia, A. Poulton, V. Kerkez, and R. Stojnic, "Galactica: A large language model for science," *arXiv preprint arXiv:2211.09085*, 2022.

[11] C. Edwards, T. Lai, K. Ros, G. Honke, K. Cho, and H. Ji, "Translation between molecules and natural language," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022, pp. 375–413.

[12] Q. Pei, W. Zhang, J. Zhu, K. Wu, K. Gao, L. Wu, Y. Xia, and R. Yan, "Biot5: Enriching cross-modal integration in biology with chemical knowledge and natural language associations," in *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[13] D. Christofidellis, G. Giannone, J. Born, O. Winther, T. Laino, and M. Manica, "Unifying molecular and textual representations via multi-task language modelling," in *Proceedings of the 40th International Conference on Machine Learning*, 2023, pp. 6140–6157.

[14] H. Cao, Z. Liu, X. Lu, Y. Yao, and Y. Li, "Instructmol: Multi-modal integration for building a versatile and reliable molecular assistant in drug discovery," 2023.

[15] Y. Fang, X. Liang, N. Zhang, K. Liu, R. Huang, Z. Chen, X. Fan, and H. Chen, "Mol-instructions: A large-scale biomolecular instruction dataset for large language models," in *ICLR*. OpenReview.net, 2024. [Online]. Available: https://openreview.net/pdf?id=Tlsdsb6l9n

[16] X. Xu, M. Li, C. Tao, T. Shen, R. Cheng, J. Li, C. Xu, D. Tao, and T. Zhou, "A survey on knowledge distillation of large language models," *arXiv preprint arXiv:2402.13116*, 2024.

[17] L. Braun, C. C. J. Dominé, J. E. Fitzgerald, and A. M. Saxe, "Exact learning dynamics of deep linear networks with prior knowledge," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022. [Online]. Available: https://openreview.net/forum?id=lJx2vng-KiC

[18] Y. Wang, Z. Li, and A. Barati Farimani, "Graph neural networks for molecules," in *Machine Learning in Molecular Sciences*. Springer, 2023, pp. 21–66.

[19] P. Schwaller, T. Laino, T. Gaudin, P. Bolgar, C. A. Hunter, C. Bekas, and A. A. Lee, "Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction," *ACS central science*, vol. 5, no. 9, pp. 1572–1583, 2019.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[21] D. Weininger, "Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules," *Journal of Chemical Information and Computer Sciences*, vol. 28, no. 1, pp. 31–36, 1988.

[22] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[23] T. Choudhary, V. Mishra, A. Goswami, and J. Sarangapani, "A comprehensive survey on model compression and acceleration," *Artificial Intelligence Review*, vol. 53, pp. 5113–5155, 2020.

[24] P. Liu, X. Wang, C. Xiang, and W. Meng, "A survey of text data augmentation," in *2020 International Conference on Computer Communication and Network Security (CCNS)*. IEEE, 2020, pp. 191–195.

[25] L. Bonifacio, H. Abonizio, M. Fadaee, and R. Nogueira, "Inpars: Data augmentation for information retrieval using large language models," *arXiv preprint arXiv:2202.05144*, 2022.

[26] J. Ye, J. Gao, Q. Li, H. Xu, J. Feng, Z. Wu, T. Yu, and L. Kong, "Zerogen: Efficient zero-shot learning via dataset generation," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022, pp. 11 653–11 669.

[27] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[28] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[29] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[30] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark *et al.*, "Training compute-optimal large language models," *arXiv preprint arXiv:2203.15556*, 2022.

[31] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[32] A. Zeng, X. Liu, Z. Du, Z. Wang, H. Lai, M. Ding, Z. Yang, Y. Xu, W. Zheng, X. Xia *et al.*, "Glm-130b: An open bilingual pre-trained model," in *The Eleventh International Conference on Learning Representations*, 2022.

[33] P. Liu, Y. Ren, J. Tao, and Z. Ren, "Git-mol: A multi-modal large language model for molecular science with graph, image, and text," *Computers in Biology and Medicine*, vol. 171, p. 108073, 2024.

[34] Y. Fang, N. Zhang, Z. Chen, L. Guo, X. Fan, and H. Chen, "Domain-agnostic molecular generation with self-feedback," in *The Twelfth International Conference on Learning Representations*, 2023.

[35] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7871–7880.

[36] M. Krenn, F. Häse, A. Nigam, P. Friederich, and A. Aspuru-Guzik, "Self-referencing embedded strings (selfies): A 100% robust molecular string representation," *Machine Learning: Science and Technology*, vol. 1, no. 4, p. 045024, 2020.

[37] J.-M. Wei, X.-J. Yuan, Q.-H. Hu, and S.-Q. Wang, "A novel measure for evaluating classifiers," *Expert Systems with Applications*, vol. 37, no. 5, pp. 3799–3809, 2010.

[38] J. Lu and Y. Zhang, "Unified deep learning model for multitask reaction predictions with explanation," *Journal of Chemical Information and Modeling*, vol. 62, no. 6, pp. 1376–1387, 2022.

[39] Q. Pei, L. Wu, K. Gao, X. Liang, Y. Fang, J. Zhu, S. Xie, T. Qin, and R. Yan, "Biot5+: Towards generalized biological understanding with iupac integration and multi-task tuning," *arXiv preprint arXiv:2402.17810*, 2024.

[40] G. L. Long and J. D. Winefordner, "Limit of detection. a closer look at the iupac definition," *Analytical chemistry*, vol. 55, no. 7, pp. 712A–724A, 1983.

[41] T. Kuang, P. Liu, and Z. Ren, "The impact of domain knowledge and multi-modality on intelligent molecular property prediction: A systematic survey," *arXiv preprint arXiv:2402.07249*, 2024.

[42] H. Li, R. Zhang, Y. Min, D. Ma, D. Zhao, and J. Zeng, "A knowledge-guided pre-training framework for improving molecular representation learning," *Nature Communications*, vol. 14, no. 1, p. 7568, 2023.

[43] H. Zhu, R. Zhou, D. Cao, J. Tang, and M. Li, "A pharmacophore-guided deep learning approach for bioactive molecular generation," *Nature Communications*, vol. 14, no. 1, p. 6234, 2023.

[44] Y. Fang, Q. Zhang, N. Zhang, Z. Chen, X. Zhuang, X. Shao, X. Fan, and H. Chen, "Knowledge graph-enhanced molecular contrastive learning with functional prompt," *Nature Machine Intelligence*, pp. 1–12, 2023.

[45] "Types of chemical reactions," 2019, [Online; accessed 2024-04-09]. [Online]. Available: https://chem.libretexts.org/@go/page/79224

[46] M. Ahmed, R. Seraj, and S. M. S. Islam, "The k-means algorithm: A comprehensive survey and performance evaluation," *Electronics*, vol. 9, no. 8, p. 1295, 2020.

Pengfei Liu is a Ph.D. candidate in the School of Computer Science and Engineering at Sun Yat-sen University, with a concurrent affiliation to Peng Cheng Laboratory. His research interests include large language models, multi-modal learning, molecular modeling and design, and visual analytics.

Jun Tao is an associate professor of computer science at Sun Yat-sen University and National Supercomputer Center in Guangzhou. He received a Ph.D. degree in computer science from Michigan Technological University in 2015. His research interest lies at the intersection of visualization, learning approaches, and science discoveries, with a focus on developing novel interactive techniques that combines human and machine intelligence to explore flow and scalar fields.

Zhixiang Ren is currently an Associate Research Scientist at Peng Cheng Laboratory and a PhD Supervisor at Southern University of Science and Technology in Shenzhen, China. He received his PhD from the University of New Mexico (Albuquerque, USA) in 2018. His research interests include AI for science and multi-modal large AI models, especially ”AI+bioinformatics” and AI-aided drug design. He has published over 50 papers with more than 7,000 citations. He has been granted several patents and has led the development of one international standard and two industry standards in the field of intelligent computing. He is also an associated editor of ”Frontiers in Big Data”, ”Big Data Mining and Analytics” and ”CAAI Artificial Intelligence Research”, as well as an experienced reviewer for multiple renowned journals such as Neural Networks and Pattern Recognition.