REAKE: CONTRASTIVE MOLECULAR REPRESENTATION LEARNING WITH CHEMICAL SYNTHETIC KNOWLEDGE GRAPH

Anonymous authors
Paper under double-blind review

ABSTRACT

Molecular representation learning has demonstrated great promise in bridging machine learning and chemical science and in supporting novel chemical discoveries. State-of-the-art methods mostly employ graph neural networks (GNNs) with self-supervised learning (SSL) and extra chemical reaction knowledge to empower the learned embeddings. However, prior works ignore three major issues in modeling reaction data, that is abnormal energy flow, ambiguous embeddings, and sparse embedding space problems. To address these problems, we propose ReaKE, a chemical synthetic knowledge graph-driven pre-training framework for molecular representation learning. We first construct a large-scale chemical synthetic knowledge graph comprising reactants, products and reaction rules. We then propose triplet-level and graph-level contrastive learning strategies to jointly optimize the knowledge graph and molecular embeddings. Representations learned by ReaKE can capture intermolecular relationships reflected in the semantic knowledge graph and molecular structures. By comparing with other state-of-the-art methods, we show that ReaKE can achieve competitive performance on the reaction prediction pretext task and the learned representations transfer well to various downstream tasks, including reaction classification, yield prediction, and molecule property prediction. Further visualization shows that the learned representations can capture the fine-grained differences both between reactions and between molecules.

1 INTRODUCTION

Organic chemistry is rapidly developed with the growing interest in big data technology(Schwaller et al., 2021b). Among them, reaction prediction becomes a necessary component of retro-synthetic analysis or virtual library generation for drug design(Kayala & Baldi, 2011). However, the prediction of chemical reaction outcomes in terms of products, yields, or reaction rates with computational approaches remains a formidable undertaking. To handle this challenge, many deep learning methods are adaptively transferred to the chemistry domain and demonstrate competitive results in solving tedious reaction classification problems, revealing the mechanistic basis of chemical reactions, and accurately predicting reaction yields(Meuwly, 2021). These researches not only advance chemical research but also effectively cut down the cost of trial and error in industrial production and improve the yield of product synthesis(Engkvist et al., 2018).

Inspired by the great success in Natural Language Processing (NLP) domain, many recent molecular representation learning methods have been proposed to employ BERT-based(Devlin et al., 2018) or Transformer-based(Vaswani et al., 2017) models, such as the SMILES-BERT(Wang et al., 2019), Mol-BERT(Li & Jiang, 2021), Mol-Transformer(Schwaller et al., 2019), RXNFP(Schwaller et al., 2021a), K-BERT(Wu et al., 2022) and KV-PLM(Zeng et al., 2022). These methods treated SMILES sequences of molecules/reactions as text and generated efficient molecular embeddings by designing molecule-made-to-measure BERT or Transformer models. However, NLP-based methods cannot effectively model molecular structural information. Still, molecular biological activity largely depends on its structure, so graph neural network (GNN)-based supervised learning methods are introduced to model the topological information (Liu et al., 2019; Yang et al., 2019; Liu et al., 2022; Ma et al., 2022). In contrast, these methods require a large amount of labeled data to improve generalization whereas task-specific labels in the real world are often of limited size(Xie et al., 2022). Therefore, self-supervised learning (SSL) methods are actively proposed to maximize the use of unlabeled data, such as GraphCL(You et al., 2020), MICRO-Graph(Zhang et al., 2020), GraphLoG(Xu et al., 2021), MolCLR(Wang et al., 2022) and KPGT(Li et al., 2022). The above methods leverage contrastive learning on molecules and focus on extracting subgraph patterns for more comprehensive molecular representations.

However, SSL on molecular graph structures remains challenging as the current approaches mostly lack domain knowledge in chemical synthesis. Recent studies have pointed out that pre-trained GNNs with random node/edge masking gives limited improvements and often lead to negative transfer on downstream tasks Hu et al. (2020); Stärk et al. (2021), as the perturbations actions on graph structures can hurt the structural inductive bias of molecules. More recently, a few studies inject extra chemical reaction knowledge into SSL training to empower the learned embeddings. For example, RxnRep (Wen et al., 2022) leverages the chemical reaction data and makes the two augmented representations of a reaction similar to each other but distinct from different reactions. MolR(Wang et al., 2021) preserves the equivalence of molecules with respect to chemical reactions in the embedding space, i.e., forcing the sum of reactant embeddings and the sum of product embeddings to be equal for each chemical equation.

Albeit promising, the previous chemical reaction-aware self-supervised methods face the following three problems: (1)***Abnormal energy flow***: all chemical reactions are accompanied by changes in entropy, and changes in entropy require reaction conditions such as temperature and pressure to trigger. Under the equivalence assumption of the previous method, the reactants and products can flow with each other as long as the embedding is equal, which violates the principle of entropy increase in the second law of thermodynamics. For example, If one has a reaction of $A + B \to C$ and a reaction of $D + E \to C$, it will result in $A + B \to D + E$, but that reaction might not occur. (2) ***Ambiguous embeddings***: the previous method assumes that the embeddings of reactants and products are equal in embedding space, however, reactants and products are often similar but totally different in property, this assumption will lead to a lack of discrimination between reactants and products in the embedding space, for example, incorrectly predicting products as reactants 5. (3) ***Sparse embedding space***: since the amount of recorded chemical reactions is limited, the embedding spaces of reactants and products learned by the previous methods are sparse and lack smoothness, which may lead to a large offset of embeddings when making a small perturbation to the reaction.

To address these problems, we develop ReaKE, a novel deep learning framework that learns chemistry-meaningful molecular representations from graph-in-graph data architecture, i.e., a knowledge graph (KG) that connects molecules using reaction templates and individual graphs that represent 2D structure of molecules. First, to alleviate the energy flow and the ambiguous embedding problems, we construct a chemical synthetic knowledge graph and build explicit connections between molecules through *reaction template* information. This can introduce the changes in reaction sites as the trigger conditions of flow between molecules, but also establish the difference between reactants and products in the embedding space. Then, for solving the sparse embedding space, we further design a functional group-augmented SSL method for reaction triplet representation learning, which can help avoid large offsets of embeddings due to small chemical perturbations. Finally, we propose a reaction-aware contrastive learning strategy to improve the training efficiency of the knowledge graph representation.

Extensive experiments demonstrate that the representations learned by our proposed model can benefit a wide-range of downstream tasks that require extensive chemical synthesis priors. For example, ReaKE achieves a 6.8% absolute Hit@1 gain in pretext reaction prediction, an average of 9.4% absolute F_1 score gain in reaction classification tasks, and an average 4% R^2 improvement in yield predictions, respectively, over existing state-of-the-art methods. Further visualization studies indicate that our reaction representations can not only categorize reactions clearly but also capture discriminative properties of reaction templates.

2 METHODS

An illustrative overview of our proposed method of molecular pre-training with **Rea**ction **K**nowledge **E**mbedding (**ReaKE**) is presented in Fig. 1. In this section, we first introduce the definition of a chemical synthetic knowledge graph, as schematically shown in Fig. 2(a). Then we depict the joint learning of the molecular encoder at the node-level and the knowledge encoder at the graph-level (section 2.2), followed by the overall pre-training objects (section 2.3).

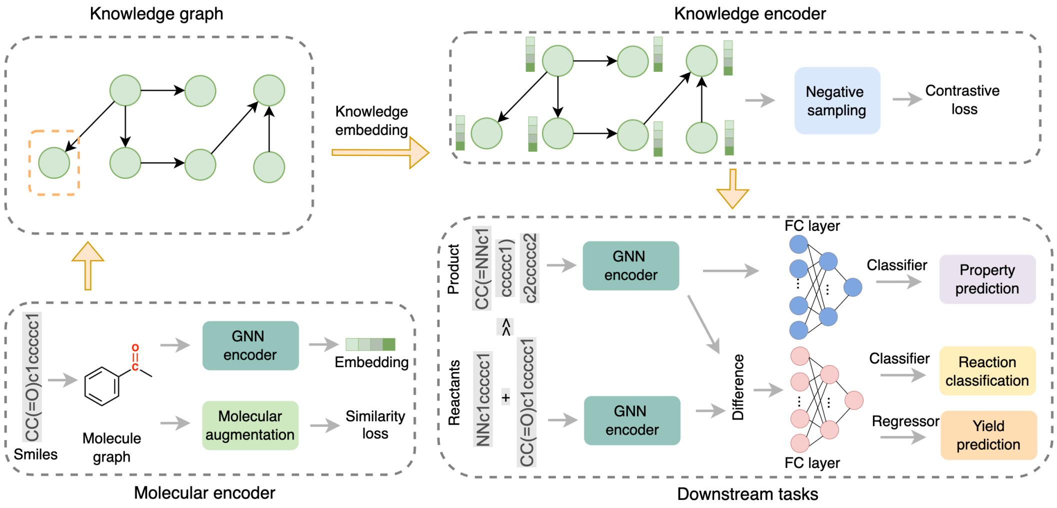


Figure 1: Overview of ReaKE. Our model joint learns molecular embeddings at the node-level and knowledge embeddings at the graph-level. The pre-trained model is evaluated on three representative tasks, including reaction classification, yield prediction and molecular property prediction.

2.1 DEFINITION OF CHEMICAL SYNTHETIC KNOWLEDGE GRAPH

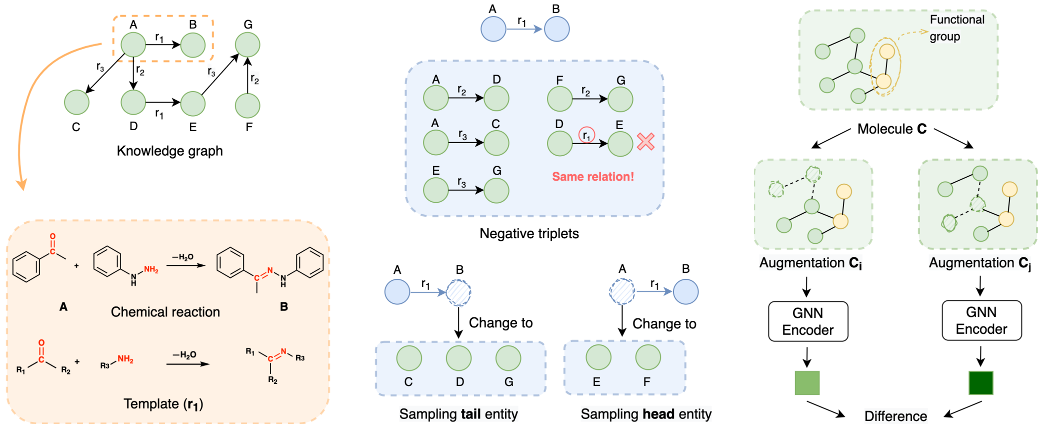
Chemically speaking, the flow between reactants and products needs to be triggered by environmental conditions such as temperature and pressure, and the reaction conditions are finally reflected on the reaction site, and the reaction site can be reflected by the reaction template, such that the introduction of template information can be used as the trigger for the reaction conditions to avoid the problem of abnormal energy flow. In this sense, we re-formulate the molecular SSL task as a multi-scale knowledge graph embedding (KGE) task Wang et al. (2017). We first introduce a chemical synthetic knowledge graph (KG) that is constructed by chemical reaction data. We define the reactant set $R = \{r_1, r_2, ...\}$ as head entities, the product set $P = \{p_1, p_2, ...\}$ as tail entities, and the reaction template set $T = \{t_1, t_2, ...\}$ as relations. Also, we define a triplet as $b_i = (r_i, t_i, p_i)$. Finally, we will have a set of triplets $B = \{b_1, b_2, \cdots\}$ that makes up the chemical synthetic knowledge graph. For example, as shown in Fig. 2(a), if we have a reaction $C_8H_8O + C_6H_8N_2 \rightarrow C_{14}H_{14}N_2$, we can extract its template $R_1 - CO - R_2 + R_3 - NH_2 \rightarrow R_1R_2 - CN - R_3$ as a relation and build a triplet $(C_8H_8O, template, C_{14}H_{14}N_2)$, $(C_6H_8N_2, template, C_{14}H_{14}N_2)$. To integrate the multi-scale representations of molecules and knowledge graphs, we utilize joint contrastive learning to fuse the heterogeneous information.

2.2 JOINT CONTRASTIVE LEARNING FOR MOLCULES AND KNOWLEDGE GRAPHS

In this section, we introduce our joint learning of triplet-level molecular representation and graph-level knowledge graph representation. The main purpose of triplet-level learning is to construct a smooth latent embedding space for reactants and products while graph-level learning aims to make triplet embeddings distributed on the embedding space more evenly. Next, we will elucidate these two parts in detail.

2.2.1 MOLECULAR REPRESENTATION LEARNING AT THE TRIPLET-LEVEL

Molecular Encoder with Graph Neural Network. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote a molecular graph with atoms $\mathcal{V} = \{v_1, v_2, \cdots\}$ and bonds $\mathcal{E} = \{d_1, d_2, \cdots\}$. Atom attributes are set as element type, total degree, atom is in ring or not, the number of connected hydrogen atoms, atom is aromatic,



(a) Chemical knowledge graph. (b) Negative sampling strategy. (c) Molecular augmentation.

Figure 2: Key points for ReaKE. (a) The construction of chemical synthetic knowledge graph, atoms with red color are reaction center. (b) The reaction-aware negative sampling strategy in a batch. (c) The self-supervised learning with functional group-based augmentation for molecules.

valence, radical electrons and so on. These atom properties are represented as a one-hot vector. GNNs utilize the graph connectivity as well as node features to learn representations of atoms and further the entire molecule. Generally, GNNs follow a message passing paradigm, in which each atom representation is iteratively updated by aggregating the representations of its neighbor atoms. At the l^{th} layer, GNN updates the atom embedding h_i of atom v_i as:

$$h_{v_i}^{(l)} = Update^{(l)}(h_{v_i}^{(l-1)}, Aggregate^{(l)}(\{h_{v_j}^{(l-1)} | v_j \in \mathcal{N}(v_i) \cup \{v_i\}\})) \tag{1}$$

where $\mathcal{N}(v_i)$ is the neighbor set of atom v_i. $Aggregate$ is an aggregation function depending on the architecture of different GNN. After L layers of message aggregating, the final representation of molecule $\mathcal{G}$ is defined as follows:

$$e_{\mathcal{G}} = Readout(\{h_{v_i}^{(L)} | v_i \in \mathcal{V}\}) \tag{2}$$

where $Readout$ function collects the representation of all atoms and obtains the final molecule representation by pooling operation.

Triplet-level Contrastive Learning. To alleviate the sparse embedding space issue, we propose a functional group-augmented SSL method to ensure that small shifts in the embeddings of reactants and products would not affect the correctness of triples. Specifically, we augment a molecule by keeping the reaction center functional group and drop atoms (crop from edge to center or randomly drop) outside the reaction center with a ratio β. An illustrative example is shown in Fig. 2(c).

In a triplet (r, t, p), the augmentation is applied twice on reactant r and product p separately, resulting in two molecular embeddings $e_{r(1)}$, $e_{r(2)}$ for e_r and two molecular embeddings $e_{p(1)}$, $e_{p(2)}$ for e_p. We minimize the agreement between the two molecules. The similarity loss is defined as follows:

$$L_{Sim} = s(e_{r(1)}, e_{r(2)}) + s(e_{p(1)}, e_{p(2)}) \tag{3}$$

where s is the distance function, $s(e_{r(1)}, e_{r(2)}) = \|e_{r(1)} - e_{r(2)}\|_2$ denotes the l_2 norm between two embeddings.

2.2.2 KNOWLEDGE REPRESENTATION LEARNING AT THE GRAPH-LEVEL

In addition to the training at the triplet level, we further leverage a contrastive knoledge embedding at the graph level to make the triplet embeddings more evenly distributed on the embedding space.

Template Encoder. To describe the inference relation from reactant molecule to product molecule, we utilize reaction templates to represent the changes that occur in the chemical process. An illustrative template example has been shown in Fig. 2(a). In particular, we leverage the toolkit RDChiral(Coley et al., 2019) to extract templates with radius 1. Given a reaction $r_1 + r_2 + \cdots + r_k \rightarrow p_1 + p_2 + \cdots + p_q$, $\{r_1, r_2, \cdots, r_k\}$ is the reactant set and $\{p_1, p_2, \cdots, p_q\}$ is the product set, we defined the template $t_{r_1} + t_{r_2} + \cdots + t_{r_k} \rightarrow t_{p_1} + t_{p_2} + \cdots + t_{p_q}$ and a GNN encoder $g(\cdot)$, the relation embedding is defined as follows:

$$e_t = \sum_{i=1}^{k} g(t_{r_i}) - \sum_{j=1}^{q} g(t_{p_j}) \tag{4}$$

where k denotes the number of reactants, n is the number of products. Note that the GNN encoder $g(\cdot)$ we used here is different from the molecule encoder.

Graph-Level Contrastive Learning. In addition to learning encoders of molecules and templates, we also propose contrastive strategies for knowledge graph embedding. In particular, we focus on the design of negative sampling Zhang et al. (2019). Negative sampling is a crucial part of the KGE training process, the generation of *hard* negative samples can greatly improve the efficiency and quality of training. However, previous approaches employ indiscriminate negative sampling in a batch that yields uninformative or wrong negatives easily. Thus, we propose a reaction-aware negative sampling which is shown in Fig. 2(b). The negative sampling strategy is detailed as follows:

Denote a mini-batch $B = \{b_1, b_2, \cdots, b_n\}$ of size n. For a triplet $b_i = (r_i, t_i, p_i) \in B$, we sample the head entity r'_i or tail entity p'_i in the rest triplets to construct negative triplets set B'_i. Head entities and tail entities in triples whose relation are t_i are excluded, that is we do not sample molecules of the same class as r_i or p_i. The set B' used to construct negative samples can be formulated as:

$$B' = \{b_k \in B | b_k = (r_k, t_k, p_k), t_k \neq t_i\} \tag{5}$$

The entity set in B' is denoted as $\{R', P'\}$ where the reactant entity set and the product entity set are represented as R' and P', respectively. The negative triplets set B'_i can then be defined as follows:

$$B'_i = \{(r'_i, t_i, p_i) | r'_i \in R'\} \cup \{(r_i, t_i, p'_i) | p'_i \in P'\} \tag{6}$$

By filtering out false triplets, the quality of negative sampling can thus be improved. We utilize TransE(Bordes et al., 2013) as the basic training objective because it is effective and simple in capturing asymmetry, inversion, and composition relations. Thus The overall KG loss is defined as follows:

$$L_{KG} = \frac{1}{n} \sum_{i=1}^{n} d_{t_i}(e_{r_i}, e_{p_i}) + \frac{1}{m} \sum_{j=1}^{m} \sigma(\gamma - d_{t_i}(e'_{r_j}, e'_{p_j})) \tag{7}$$

where m is the number of negative samples, it is up to $2(n-1)$. σ means the sigmoid function, which helps avoid learning easy triplets. (r'_j, t_i, p'_j) is the j^{th} negative triplet in B'_i. γ is the margin hyperparameter and d_t is the distance function.

$$d_t(e_r, e_p) = \|(e_r + e_t) - e_p\|_2 \tag{8}$$

2.3 OPTIMIZATION OBJECTIVES

Overall, the model is trained jointly with a weighted sum of knowledge embedding loss and molecule augmentation loss mentioned above, where λ stands for the trade-off parameter. It is described as follows:

$$L = L_{KG} + \lambda L_{Sim} \tag{9}$$

3 EXPERIMENTS

In this section, we conduct various experiments to demonstrate the generality of our model. We first use the embedding-based chemical reaction prediction task to validate whether the framework effectively solves the abnormal energy flow issue. Then, we introduce two reaction-related downstream tasks: reaction classification and yield prediction, to investigate whether our model could capture the changes between the before and after of a reaction (template information). Finally, we explore the generalization ability of ReaKE through molecular property prediction tasks.

3.1 DATASETS

Pre-training Dataset. The dataset we leverage for constructing the chemical synthetic knowledge graph here is USPTO-479k(Zheng et al., 2019). It contains reactions with up to five reactants and only one product. By removing reactions from which we cannot extract templates and others that contain the same reactants and products, we finally obtain 407,039 training reactions, 29,848 validation reactions and 39,802 testing reactions. We further convert the training reactions into 587,403 triplets with 103,339 reaction templates.

Downstream Task Datasets. For the reaction classification task, we use the Schneider dataset (Schneider et al., 2015). It is derived from the Schneider 50k dataset, which is a descendant of the USPTO dataset of patent reactions. After further cleaning, we obtain 38,800 reactions with 46 reaction types. We split it into a training set with 31,002 reactions, a validation set with 3,896 reactions and a test set with 3,902 reactions. For the yield prediction task, a palladium-catalyzed Buchwald-Hartwig C-N cross-coupling reactions dataset(Ahneman et al., 2018) is utilized to evaluate model performance in our experiment. It includes 3,955 reactions labeled with yield. These reactions are composed of 15 aryl halides, 1 methylaniline, 4 Buchwald ligands, 1 Pd catalyst, 3 bases and 23 additives. In addition, following drfp(Probst et al., 2022), we used four out-of-sample splits based on isoxazole additives and created a 70/10/20 train/valid/test split. For the molecular property prediction task, we use four Open Graph Benchmark (OGB) datasets with their standard scaffold splits and random splits. The datasets are BACE, Tox21, Clintox and HIV dataset(Wu et al., 2018).

3.2 CHEMICAL REACTION PREDICTION

Baselines. Following the evaluation protocol of MolR(Wang et al., 2021), we compare our models with several state-of-the-art molecular representation methods, including Mole2vec(Jaeger et al., 2018), MolBERT(Li & Jiang, 2021), and MolR. In the baselines, reactants and products are embedded by a molecule encoder, and the dot product of two embeddings is used for ranking product candidates. The pre-training setup can be found in Appendix A.

Evaluation Protocol. We use product ranking prediction to evaluate our pretrained model. In the test set, considering the direct use of templates or reaction types will lead to the problem of product data leakage, which is not conducive to our product ranking task, we split the template into reactant templates $\{t_{r_1}, t_{r_2}, \cdots, t_{r_k}\}$ and product template $\{t_{p_1}, t_{p_2}, \cdots, t_{p_q}\}$. All products and its t_p in the test set are treated as candidates. For reactants in a reaction, we calculate the embedding distance between $(e_r + \sum_{i=1}^{k} g(t_{r_i}))$ and all product candidates' $(e_p + \sum_{j=1}^{q} g(t_{p_j}))$, and rank the candidates by distance. Then, the true product's ranking is used to calculate mean reciprocal rank (MRR), mean rank (MR), and top-k hit ratio (Hit@k) which are standard evaluation metrics in KG models. Higher MRR, higher Hit@k and lower MR indicate that model achieves a better performance.

Table 1: Results of product ranking prediction.

Methods	MRR	MR	Hit@1	Hit@3	Hit@5	Hit@10
Mol2vec	0.681	483.7	0.614	0.725	0.759	0.798
Mol2vec-FT1	0.688 ± 0.000	417.6 ± 0.1	0.620 ± 0.000	0.734 ± 0.000	0.767 ± 0.000	0.806 ± 0.000
MolBert	0.708	460.7	0.623	0.768	0.811	0.858
MolBert-FT1	0.731 ± 0.000	457.9 ± 0.0	0.649 ± 0.000	0.790 ± 0.000	0.831 ± 0.000	0.873 ± 0.000
MolBERT-FT2	0.776 ± 0.000	459.6 ± 0.2	0.708 ± 0.000	0.827 ± 0.000	0.859 ± 0.000	0.891 ± 0.000
MolR	0.918 ± 0.000	27.4 ± 0.4	0.882 ± 0.000	0.949 ± 0.001	0.960 ± 0.001	0.970 ± 0.000
ReaKE-SAGE	0.953 ± 0.001	4.1 ± 0.2	0.930 ± 0.001	0.973 ± 0.001	0.980 ± 0.000	0.987 ± 0.000
ReaKE-GAT	0.965 ± 0.001	6.6 ± 0.4	0.946 ± 0.001	0.982 ± 0.001	0.986 ± 0.001	0.990 ± 0.000
ReaKE-GCN	0.966 ± 0.000	4.5 ± 0.1	0.948 ± 0.000	**0.983 ± 0.000**	0.987 ± 0.001	0.991 ± 0.000
ReaKE-TAG	**0.967 ± 0.000**	**2.9 ± 0.0**	**0.950 ± 0.000**	0.982 ± 0.000	**0.987 ± 0.000**	**0.992 ± 0.000**

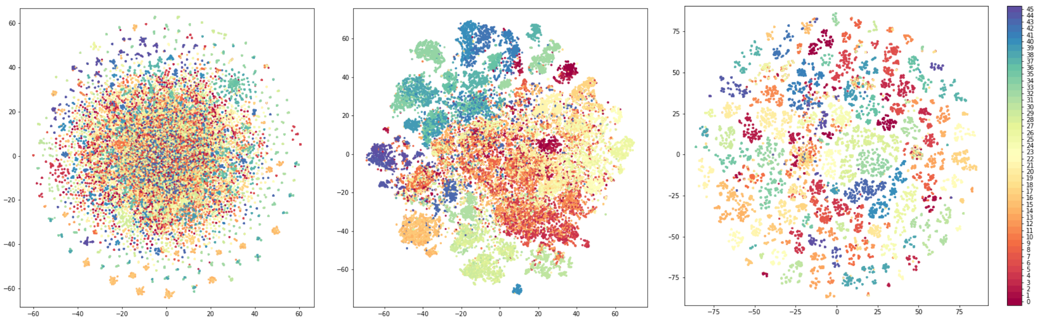
Results. As shown in Tab. 1, our ReaKE gains about 4.9% MRR and 6.8% Hit@1 performance enhancement against the baseline model MolR and outperforms all Bert-based models. In addition, ReaKE outperforms the baseline regardless of which GNN is combined, suggesting the challenges are better tackled with the external reaction template information. Appendix D shows the dice similarity between the predicted product and ground-truth product and cases where our method predicted correctly but other methods did not. Taken together, these results indicate that our method can avoid the situation of predicting products as reactants and can learn the changes between reactants and products, which also confirms the effectiveness of introducing templates.

3.3 REACTION CLASSIFICATION

Baselines and Evaluation Protocol. We consider several molecule fingerprints and reaction fingerprints as baselines, including AP3(Carhart et al., 1985), DRFP(Schneider et al., 2015), RXNFP(Schwaller et al., 2021a) and RxnRep(Wen et al., 2022). AP3 is an atom-pairs molecular fingerprint method with a maximum path length of three. DRFP creates a hash binary fingerprint based on the symmetric difference between substructures.

Table 2: Classification F_1 score on the Schneider dataset.

Methods	4 reactions per class	8 reactions per class	16 reactions per class	32 reactions per class	64 reactions per class	128 reactions per class
AP3	0.518 ± 0.004	0.620 ± 0.004	0.703 ± 0.006	0.761 ± 0.002	0.799 ± 0.004	0.828 ± 0.004
DRFP	0.100 ± 0.005	0.129 ± 0.004	0.199 ± 0.008	0.266 ± 0.007	0.338 ± 0.006	0.398 ± 0.002
RXNFP	0.322 ± 0.012	0.394 ± 0.013	0.471 ± 0.010	0.531 ± 0.006	0.575 ± 0.005	0.618 ± 0.004
RxnRep	0.441 ± 0.010	0.634 ± 0.003	0.767 ± 0.003	0.831 ± 0.002	0.875 ± 0.003	0.900 ± 0.002
MolR	0.629 ± 0.007	0.722 ± 0.009	0.803 ± 0.006	0.862 ± 0.006	0.901 ± 0.004	0.900 ± 0.027
ReaKE-SAGE	0.796 ± 0.011	0.860 ± 0.010	0.892 ± 0.004	0.908 ± 0.003	0.918 ± 0.003	0.933 ± 0.002
ReaKE-GAT	0.778 ± 0.012	0.844 ± 0.004	0.876 ± 0.005	0.897 ± 0.005	0.917 ± 0.003	0.923 ± 0.007
ReaKE-GCN	0.765 ± 0.008	0.842 ± 0.008	0.877 ± 0.009	0.898 ± 0.003	0.918 ± 0.004	0.923 ± 0.016
ReaKE-TAG	**0.821 ± 0.011**	**0.882 ± 0.003**	**0.901 ± 0.005**	**0.916 ± 0.004**	**0.928 ± 0.003**	**0.935 ± 0.004**



(a) DRFP. (b) RXNFP. (c) ReaKE.

Figure 3: t-SNE visualization of ReaKE, DRFP, RXNFP on schneider dataset. (a), (b)and (c)show the fingerprint distribution of the entire schneider dataset, with 46 colors representing 46 categories of reactions.

To evaluate the effectiveness of the learned representations, we use our pre-trained model as a feature extractor and obtain the final reaction representations by calculating the difference between reactants and products. Then, we train MLP for reaction classification. Following the few-shot setting of RxnRep(Wen et al., 2022), instead of using the entire training set, we sample 4, 8, 16, 32, 64, 128 reactions per class to simulate the situation of a small dataset. Every experiment is repeated five times with the resampling of training data. In addition, we visualize the initial embedding space directly encoded by the pre-trained model. We demonstrate the distribution of chemical reaction embeddings with different classes and the distribution of reactants and products within a type of reaction.

Results. The evaluation results under transfer feature extraction setting are illustrated in Tab. 2, ReaKE achieves 19.2%, 16%, 9.8%, 5.4%, 2.7%, 3.5% F_1 score gain over the state-of-art model with 4, 8, 16, 32, 64, 128 reactions per class as the training set. It demonstrates our learned embeddings can transfer well to downstream tasks with a small training set. The large margins between our method and MolR indicate that the addition of reaction template information can make the chemical representation more discriminative.

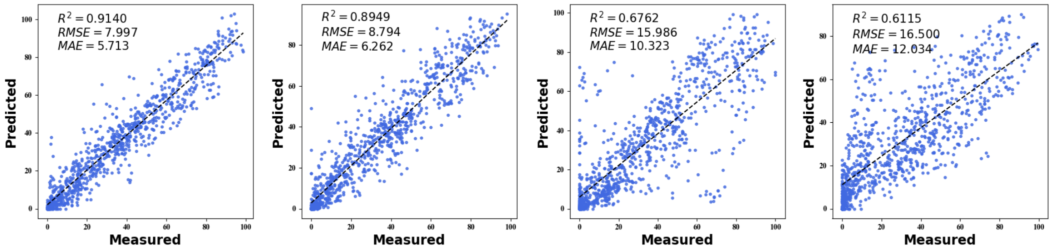
t-SNE visualizations of DRFP, RXNFP and our method on the Schneider dataset are shown in Fig. 6. We perform two kinds of visualizations: One is the visualization on the reaction fingerprint level as shown in Fig. 3(a), Fig. 3(b) and Fig. 3(c), that is obtain the embeddings of all reactions from pre-trained model, and then conduct t-SNE visualization to see the distribution of the initial embeddings in the feature space. As shown in Fig. 3(c), there is a clear boundary between each category and almost no overlapping parts. Furthermore, the reactions of the same category are clustered in multiple clusters, which demonstrates that our reaction representations can not only categorize reactions clearly but also capture discriminative properties about reaction templates. The other is the visualization on the molecular level which can be found in Appendix B.

3.4 YIELD PREDICTION

Baselines and Evaluation Protocol. For comparison, we choose the DFT-based method, DRFP(Schneider et al., 2015), Yield-BERT and its augmented version Yield-BERT (aug.) (Schwaller et al., 2020) as baselines. Yield-BERT is an extension of the learned RXNFP fingerprint with a regression layer.

Table 3: R^2 of yield prediction on Buchwald Hartwig reactions.

	Test 1	Test 2	Test 3	Test 4	Avg. 1-4
DFT	0.80	0.77	0.64	0.54	0.69
DRFP	0.81 ± 0.010	0.83 ± 0.003	0.71 ± 0.001	0.49 ± 0.004	0.71 ± 0.160
Yield-BERT	0.84 ± 0.010	0.84 ± 0.030	**0.75 ± 0.040**	0.49 ± 0.050	0.73
Yield-BERT(aug.)	0.80 ± 0.010	0.88 ± 0.020	0.56 ± 0.080	0.43 ± 0.040	0.58 ± 0.330
MolR	0.68 ± 0.003	0.84 ± 0.002	0.61 ± 0.006	0.51 ± 0.003	0.66 ± 0.120
ReaKE	**0.87 ± 0.002**	**0.89 ± 0.002**	0.67 ± 0.004	**0.57 ± 0.005**	**0.75 ± 0.140**



(a) Test 1. (b) Test 2. (c) Test 3. (d) Test 4.

Figure 4: Fine-tune performance R^2 scores (%) on Test1-4 of ReaKE.

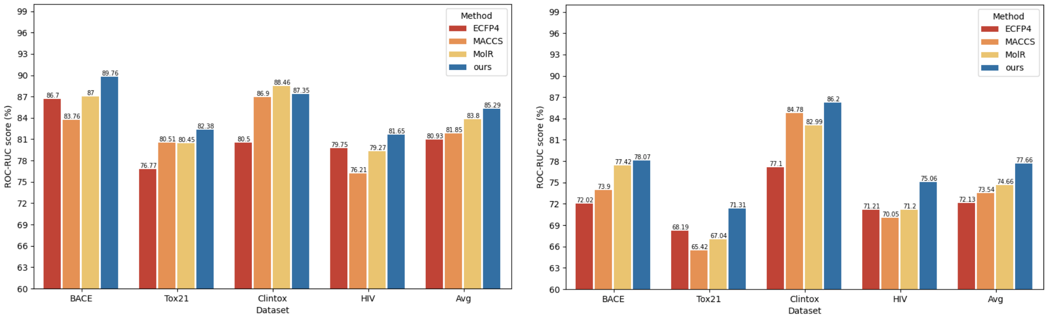
We obtain the reaction representations from our pre-trained model, and then feed it into xg-boost(Chen & Guestrin, 2016). We leveraged the four tests in DRFP(Schneider et al., 2015) to show the results under splits based on isoxazole additives with 5 times run. We report the mean and standard deviation of R^2 score.

Result. We apply feature extraction setting and fine-tune setting on the yield prediction task. The results under two settings are shown in Tab. 3 and Fig. 4 separately. For the feature extraction, our method gains about 2% average R^2 value enhancement on Test1-4. For the fine-tuning setting, we achieve a 4% average R^2 improvement on Test1-4. In addition, in Test1 and Test4, our fine-tune model gains about 7% R^2 improvement. In conclusion, our method can capture subtle changes in the reaction and improve the performance of prediction yield prediction, which also confirms that pretext template information is beneficial for downstream tasks.

3.5 MOLECULE PROPERTY PREDICTION

Baselines and Evaluation Protocol. To further confirm our pre-trained model's potential when addressing tasks not related to chemical synthesis, we apply ReaKE to molecular property predictions. we select the rule-based methods ECFP4(Rogers & Hahn, 2010), MACCCS(Heikamp & Bajorath, 2011), and the self-supervised learning method MolR as baselines. Due to the large gap between the pretext dataset and property datasets, we verify our method by adding an MLP layer under the fine-tuning setting. We demonstrate the average ROC-AUC score of the datasets under random splits and scaffold splits, we apply 5-fold cross-validation to random splits and run experiments 5 times with different random seeds under scaffold splits.

Results. The evaluation results under fine-tune setting are illustrated in Fig. 5. On average, we achieve a 1.5% and 3.0% gain on the random splits and scaffold splits. It suggests our method is effective for training at the node level.



(a) Random split. (b) Scaffold split.

Figure 5: Fine-tune performance ROC-AUC scores (%) on property prediction benchmarks.

3.6 ABLATION STUDY

We further analyze the contribution of different components under two kinds of augmentation in our ReaKE. The drop atom mode randomly crops atoms outside the reaction center, and the subgraph mode crops atoms from edge to center. The variants are as follows: (1) **w/o neg**: w/o negative sampling; (2) **w/o aug**: w/o molecular augmentation; (3) **w/o temp**: w/o adding reaction templates information.

The ablation results on chemical reaction prediction are reported in Appendix C. In specific, the removal of the template component leads to the most significant performance drop, which is align with our assumption that template information can benefit molecular modeling. Note that molecular augmentation can bring obvious improvements. We show two kinds of augmentations: drop atom mode and drop subgraph mode. The exclusion of both strategies will decrease the results, showing the importance of augmentation. We also try to iterate the estimation w/o negative sampling but observe a performance drop given more iterations.

4 CONCLUSION

In this work, we propose a simple yet effective chemical synthetic knowledge graph to tackle the challenges in modeling reaction data. We introduce the changes in reaction sites as the trigger conditions of flow between molecules and build explicit connections between molecules through reaction template information. Besides, we design a functional group-augmented SSL method for reaction triplet representation learning, which can help avoid large offsets of embeddings due to small chemical perturbations. Finally, we propose a reaction-aware contrastive learning strategy to improve the training efficiency of the knowledge graph representation. Comprehensive experiments over multiple benchmark downstream tasks consistently demonstrate that our pre-trained model can be generalized to either reactions or molecular-related tasks and outperforms previous baselines under both feature extractor mode and fine-tune mode.

Further directions to explore can be described as the following points: First, we can take into account the 3D structure of molecules and reactions to improve the expressiveness of the embedding. Second, it is interesting to consider external factors related to the reaction (i.e., temperature, reaction conditions, etc.), which would probably improve the performance of tasks such as yield prediction. Finally, interpreting the embeddings learned by reaction data also tends to be meaningful.

REFERENCES

Derek T Ahneman, Jesús G Estrada, Shishi Lin, Spencer D Dreher, and Abigail G Doyle. Predicting reaction performance in c–n cross-coupling using machine learning. *Science*, 360(6385):186–190, 2018.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26, 2013.

Raymond E Carhart, Dennis H Smith, and R Venkataraghavan. Atom pairs as molecular features in structure-activity studies: definition and applications. *Journal of Chemical Information and Computer Sciences*, 25(2):64–73, 1985.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Connor W Coley, William H Green, and Klavs F Jensen. Rdchiral: An rdkit wrapper for handling stereochemistry in retrosynthetic template extraction and application. *Journal of chemical information and modeling*, 59(6):2529–2537, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Ola Engkvist, Per-Ola Norrby, Nidhal Selmi, Yu-hong Lam, Zhengwei Peng, Edward C Sherer, Willi Amberg, Thomas Erhard, and Lynette A Smyth. Computational prediction of chemical reactions: current status and outlook. *Drug discovery today*, 23(6):1203–1218, 2018.

Kathrin Heikamp and Jürgen Bajorath. How do 2d fingerprints detect structurally diverse active compounds? revealing compound subset-specific fingerprint features through systematic selection. *Journal of chemical information and modeling*, 51(9):2254–2265, 2011.

Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay S. Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. In *8th International Conference on Learning Representations, ICLR*. OpenReview.net, 2020.

Sabrina Jaeger, Simone Fulle, and Samo Turk. Mol2vec: unsupervised machine learning approach with chemical intuition. *Journal of chemical information and modeling*, 58(1):27–35, 2018.

Matthew Kayala and Pierre Baldi. A machine learning approach to predict chemical reactions. *Advances in Neural Information Processing Systems*, 24, 2011.

Han Li, Dan Zhao, and Jianyang Zeng. Kpgt: Knowledge-guided pre-training of graph transformer for molecular property prediction. *arXiv preprint arXiv:2206.03364*, 2022.

Juncai Li and Xiaofei Jiang. Mol-bert: An effective molecular representation with bert for molecular property prediction. *Wireless Communications and Mobile Computing*, 2021, 2021.

Shengchao Liu, Mehmet F Demirel, and Yingyu Liang. N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. *Advances in neural information processing systems*, 32, 2019.

Songtao Liu, Rex Ying, Hanze Dong, Lanqing Li, Tingyang Xu, Yu Rong, Peilin Zhao, Junzhou Huang, and Dinghao Wu. Local augmentation for graph neural networks. In *International Conference on Machine Learning*, pp. 14054–14072. PMLR, 2022.

Hehuan Ma, Yatao Bian, Yu Rong, Wenbing Huang, Tingyang Xu, Weiyang Xie, Geyan Ye, and Junzhou Huang. Cross-dependent graph neural networks for molecular property prediction. *Bioinformatics*, 38(7):2003–2009, 2022.

Markus Meuwly. Machine learning for chemical reactions. *Chemical Reviews*, 121(16):10218–10239, 2021.

Daniel Probst, Philippe Schwaller, and Jean-Louis Reymond. Reaction classification and yield prediction using the differential reaction fingerprint drfp. *Digital discovery*, 1(2):91–97, 2022.

David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

Nadine Schneider, Daniel M Lowe, Roger A Sayle, and Gregory A Landrum. Development of a novel fingerprint for chemical reactions and its application to large-scale reaction classification and similarity. *Journal of chemical information and modeling*, 55(1):39–53, 2015.

Philippe Schwaller, Teodoro Laino, Théophile Gaudin, Peter Bolgar, Christopher A Hunter, Costas Bekas, and Alpha A Lee. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS central science*, 5(9):1572–1583, 2019.

Philippe Schwaller, Alain C Vaucher, Teodoro Laino, and Jean-Louis Reymond. Data augmentation strategies to improve reaction yield predictions and estimate uncertainty. 2020.

Philippe Schwaller, Daniel Probst, Alain C Vaucher, Vishnu H Nair, David Kreutter, Teodoro Laino, and Jean-Louis Reymond. Mapping the space of chemical reactions using attention-based neural networks. *Nature Machine Intelligence*, 3(2):144–152, 2021a.

Philippe Schwaller, Alain C Vaucher, Teodoro Laino, and Jean-Louis Reymond. Prediction of chemical reaction yields using deep learning. *Machine learning: science and technology*, 2(1):015016, 2021b.

Hannes Stärk, Dominique Beaini, Gabriele Corso, Prudencio Tossou, Christian Dallago, Stephan Günnemann, and Pietro Liò. 3d infomax improves gnns for molecular property prediction. *arXiv preprint arXiv:2110.04126*, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Hongwei Wang, Weijiang Li, Xiaomeng Jin, Kyunghyun Cho, Heng Ji, Jiawei Han, and Martin D Burke. Chemical-reaction-aware molecule representation learning. *arXiv preprint arXiv:2109.09888*, 2021.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12): 2724–2743, 2017.

Sheng Wang, Yuzhi Guo, Yuhong Wang, Hongmao Sun, and Junzhou Huang. Smiles-bert: large scale unsupervised pre-training for molecular property prediction. In *Proceedings of the 10th ACM international conference on bioinformatics, computational biology and health informatics*, pp. 429–436, 2019.

Yuyang Wang, Jianren Wang, Zhonglin Cao, and Amir Barati Farimani. Molecular contrastive learning of representations via graph neural networks. *Nature Machine Intelligence*, 4(3):279–287, 2022.

Mingjian Wen, Samuel M Blau, Xiaowei Xie, Shyam Dwaraknath, and Kristin A Persson. Improving machine learning performance on small chemical reaction data with unsupervised contrastive pretraining. *Chemical science*, 13(5):1446–1458, 2022.

Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018.

Zhenxing Wu, Dejun Jiang, Jike Wang, Xujun Zhang, Hongyan Du, Lurong Pan, Chang-Yu Hsieh, Dongsheng Cao, and Tingjun Hou. Knowledge-based bert: a method to extract molecular features like computational chemists. *Briefings in Bioinformatics*, 23(3):bbac131, 2022.

Yaochen Xie, Zhao Xu, Jingtun Zhang, Zhengyang Wang, and Shuiwang Ji. Self-supervised learning of graph neural networks: A unified review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

Minghao Xu, Hang Wang, Bingbing Ni, Hongyu Guo, and Jian Tang. Self-supervised graph-level representation learning with local and global structure. In *International Conference on Machine Learning*, pp. 11548–11558. PMLR, 2021.

Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian P Kelley, Andrew Palmer, Volker Settels, et al. Are learned molecular representations ready for prime time? 2019.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems*, 33:5812–5823, 2020.

Zheni Zeng, Yuan Yao, Zhiyuan Liu, and Maosong Sun. A deep-learning system bridging molecule structure and biomedical text with comprehension comparable to human professionals. *Nature communications*, 13(1):1–11, 2022.

Shichang Zhang, Ziniu Hu, Arjun Subramonian, and Yizhou Sun. Motif-driven contrastive learning of graph representations. *arXiv preprint arXiv:2012.12533*, 2020.

Yongqi Zhang, Quanming Yao, Yingxia Shao, and Lei Chen. Nscaching: simple and efficient negative sampling for knowledge graph embedding. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, pp. 614–625. IEEE, 2019.

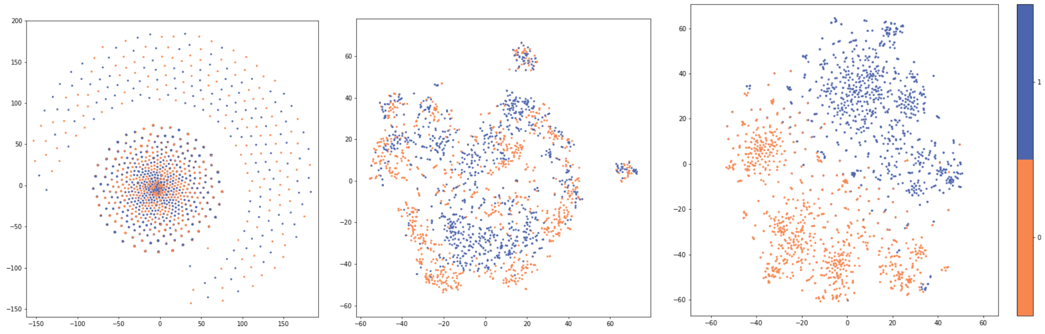
Shuangjia Zheng, Jiahua Rao, Zhongyue Zhang, Jun Xu, and Yuedong Yang. Predicting retrosynthetic reactions using self-corrected transformer neural networks. *Journal of chemical information and modeling*, 60(1):47–55, 2019.

A PRE-TRAINING SETUP

We consider four GNN models as molecule encoders, they are GCN, GAT, SAGE, and TAG. In all models, we use a 2-layer GNN with a sum pooling Readout function and project the representation to a 1024-dimensional latent space for both molecule encoder and template encoder. Besides, we optimize our model using Adam optimizer with a learning rate of 5e-5. As a default setting, we use a margin value γ of 4.0, a molecule augmentation drop ratio β of 0.7, and a trade-off parameter α of 1.0. Furthermore, we train at batch-size 1024 for 30 epochs.

B VISUALIZATION ON SCHNEIDER DATASET

We further demonstrate the distribution of reactants and products within a type of reaction in Schneider dataset.



(a) DRFP. (b) RXNFP. (c) ReaKE.

Figure 6: t-SNE visualization of ReaKE, DRFP, RXNFP on schneider dataset. (a), (b)and (c) demonstrate the distribution of reactant and product embeddings within the reactions of the same class, label **0** stands for reactants and label **1** stands for products.

The visualization on the molecular level is shown in Fig. 6(a), Fig. 6(b) and Fig. 6(c), which is gaining the embeddings of reactants and products in a class of Schneider reactions, and then using t-SNE visualization to explore the distribution. As shown in Fig. 6(c), the reactants and products form clusters separately, which suggests the ***Ambiguous embeddings*** problem is alleviated, and there exist differences between the embeddings of the reactant and the product. Besides, our model can aggregate molecules with the same functional group.

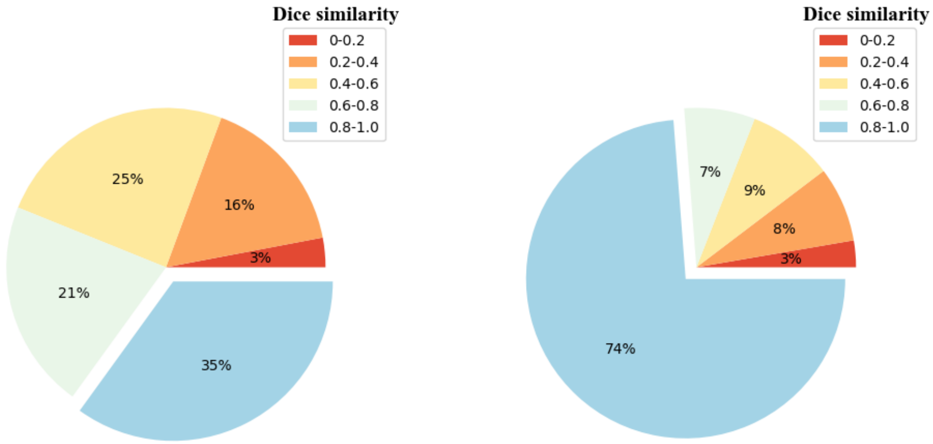
C ABLATION STUDY

Table 4: Ablation study results on chemical reaction prediction.

Method	**MRR**	**MR**	**Hit@1**	**Hit@10**
ReaKE (drop atom)	**0.967**	**2.8**	**0.950**	**0.992**
-w/o neg	0.925	11.6	0.900	0.967
-w/o aug	0.953	3.7	0.930	0.987
-w/o temp	0.917	33.9	0.880	0.970
ReaKE (subgraph)	**0.967**	**3.3**	**0.950**	**0.991**
-w/o neg	0.922	18.0	0.896	0.966
-w/o aug	0.953	3.7	0.930	0.987
-w/o temp	0.914	40.4	0.880	0.965

D CASE STUDY

To explore whether our method can effectively solve the ***Abnormal energy flow*** problem in the pretext task, we statistics the distribution of dice FP similarities between predicted products and real products. We except the cases where both our method and MolR predict the true product for demonstrating the differences more explicitly.



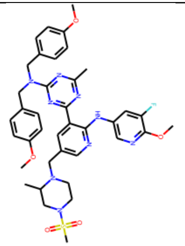
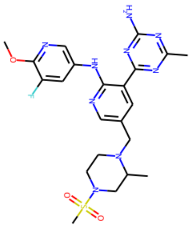
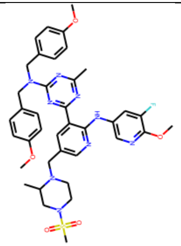
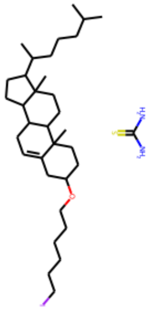
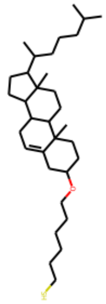
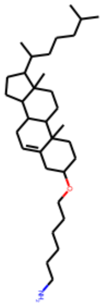
(a) MolR.

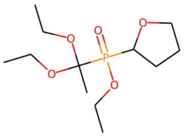
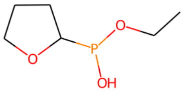
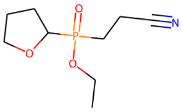
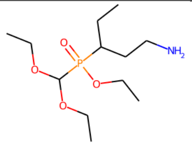
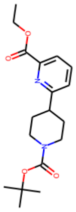
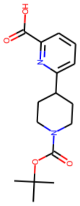
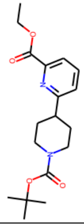
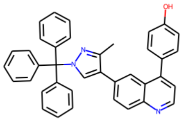
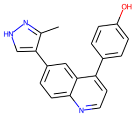
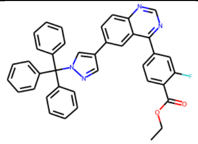
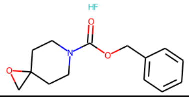
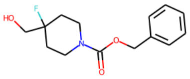
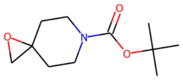
(b) ReaKE.

Figure 7: Pie chart of dice FP similarities distribution of predicted and real products.

We also conduct a case study on USPTO-479k and select the first 100 reactions on the test dataset to see if our optimization is effective compared with MolR. The detailed results are demonstrated on Table 5, we remove examples where all methods predict correctly.

Table 5: Case study on the first 100 reactions of USPTO-479k test set.

Index	Reactant(s)	Ground-truth product	Predicted product by ReaKE	Predicted product by MolR
5			same as ground-truth	
10			same as ground-truth	

66				
72			same as ground-truth	
79			same as ground-truth	
82			same as ground-truth	

As shown in Table 5, our method models chemical reaction data in detail and avoids some cases where the predicted product value is consistent with the reactant (such as the No.5 reaction and the No.72 reaction). This phenomenon also confirms that our initial analysis of the problem and optimization measures are effective. Besides, there are reactions (such as the No. of 66 reaction) that cannot be predicted correctly by both methods, which is due to the fact that only the main product is retained in USPTO-479k, and the by-product is omitted. This deficiency leads to inaccurate prediction, but the correct product ranking predicted by our method is still better than previous methods.