2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) | 978-1-6654-0126-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/BIBM52615.2021.9669794

MolCloze: A Unified Cloze-style Self-supervised Molecular Structure Learning Model for Chemical Property Prediction

Yingheng Wang*
Department of Electronic Engineering
Tsinghua University
Beijing, China
wangyh20@mails.tsinghua.edu.cn

Xin Chen*
Technology and Engineering Group
Tencent
Shenzhen, China
marcuschen@tencent.com

Yaosen Min
Institute of Interdisciplinary Information Sciences
Tsinghua University
Beijing, China
minys18@mails.tsinghua.edu.cn

Ji Wu
Department of Electronic Engineering
Tsinghua University
Beijing, China
wuji_ee@mail.tsinghua.edu.cn

***Abstract*—Machine Learning approaches are required to predict accurately on test samples that are distributionally different from training ones in the fields of drug discovery, computational biology, and cheminformatics. However, (i) labeled task-specific molecule data are often scarce, and (ii) poor generalization due to test molecules that are structurally different from those seen during training. To alleviate the problems, we propose a cloze-style self-supervised learning model (MolCloze) to obtain universal informative representations for molecular property prediction tasks. With carefully designed self-supervised tasks unifying generative- and discriminative-paradigm, MolCloze can learn rich structural and semantic information of molecules from enormous unlabelled molecular data. To capture such complex information, we design two novel strategies - Structural Fingerprint Tokenization (SFT) for better tokenizing molecule graphs, and Normalized Graph Raw Shortcut-connection (NGRS) for better latent representations by training a deeper model. We pre-train the MolCloze model via three tasks, which are Unordered Masked Language Modeling (UMLM) , Replaced Masked Token Detection (RMTD), and Contrastive Energy-based Unmasked Token Clozing (CE-UTC). Then, we transfer the pre-trained model to a broad range of downstream molecular property prediction tasks via minor architecture modification. Extensive experiments demonstrate the generalizability of MolCloze by predicting a broad range of chemical properties which are related to drug discovery. We also observe significant performance boost on different downstream molecular property prediction datasets, achieving higher performance than the state-of-the-art baseline approaches and previous pre-training techniques developed for molecule data.**

***Index Terms*—Self-supervised learning, molecule modeling, pre-training, chemical property prediction**

I. Introduction

The large intersection between chemistry and machine learning enables the domain of cheminformatics to apply machine learning methods in chemical modeling for decades. These applications help to improve the capability of accurate prediction of molecular properties in the chemical and pharmaceutical industries. It benefits various academic areas and industrial domains such as improvement to rational chemical design, reducing R&D cost, decreasing the failure rate in potential drug screening trials, as well as speeding the process of new drug discovery. However, there are still essential issues regarding traditional machine learning approaches that they **mainly rely on intensive manual feature engineering** and **strong domain knowledge**. Besides, the designed molecular fingerprints **are highly task-dependent, not general enough** for other property prediction task.

In the past few years, with the renaissance of neural networks and the prevalence of deep learning methods, sophisticated deep learning methods have been adopted in many chemical and biological applications [1]. In many such applications, deep learning is verified to match or even exceed conventional manual experimental methods which are accurate and valid but costly and slow, demonstrating that deep learning is a powerful tool in learning feature for data and good at task-related prediction. However, for molecular property prediction tasks, there still exist two problems - **unavailability of sufficient labeled task-specific data** and **evaluation discrepancy from real-world application scenarios**. The former problem prevents state-of-the-art task-specific models from being sufficiently optimized [2]. Massive experiments conducted in MoleculeNet benchmark datasets [3] indicate that when the dataset on which the model is trained is very small, the optimized data-driven model may under-perform the conventional methods. The latter problem refers to the fact that most molecular property prediction models are evaluated using the random splitting. The conventional random splitting deviates from the real-world application scenarios where testing substances are

*The first two authors made equal contribution to this work.

structurally different from those in training set substantially. Therefore, we inspect that the models evaluated under the random splitting may be over-optimistic, and the predictions may be unreliable. For instance, when one pharmaceutical scientist wants to predict the molecular properties of a newly-synthesized molecule, the models which are trained on an extremely limited task-specific dataset are not likely to give a reliable prediction.

However, the existing molecular property prediction methods cannot deal with the two problems well. The current mainstreams can be classified into three categories in terms of input molecule representation. The three kinds of input molecule representations chemical fingerprint [4], SMILES string [5], and molecular graph [6]. The first category is the models [7] with chemical fingerprint(s) as input. On the one hand, the chemical fingerprint encodes pre-defined domain knowledge, which is beneficial for convergence. For another, developers are unable to identify whether the used chemical fingerprint is relevant to the task at hand. Therefore, it's hard for these methods to achieve excellent performance across multiple downstream tasks. The second category is the Natural Language Processing(NLP) approaches with SMILES string as input. SMILES string is a widely used sequential notation for molecule data. There is an analogous algorithm to Word2vec [8], Mol2vec [9] in the field of molecular property prediction. Mol2vec can also provide good initialization and fast convergence, but not contextualized due to its shallow architecture. Besides, merely viewing molecule data with sophisticated internal connectivity as sequence data is not reasonable and hard to interpret. The third category is graph neural networks (GNNs) [6] with molecular graphs as input. Molecular graph data can be directly handled by GNNs, and there are many GNNs which can be used to encode molecular graph data. Though different GNNs have demonstrated considerable performance on several datasets, they need to be trained via supervised learning on sufficient task-specific labeled data from scratch, which is not very suitable for the setting of molecular property prediction tasks where labeled task-specific data is scarce, and out-of-distribution substances are universal.

From our viewpoints, the emerging pre-trained language models based on Self-Supervised Learning (SSL) can provide us with the desired representations for molecule data, especially the cloze task [10], which has proven highly effective for representation learning over text by predicting the identity of a token given its surrounding context. Although generative SSL models implementing cloze tasks like BERT [11] achieve good performance by replacing input tokens with [MASK], it still incurs drawbacks in efficiency and introduces a pre-train/fine-tune mismatch where BERT sees [MASK] tokens in training but not in fine-tuning. Discriminative SSL models like ELECTRA [12] uses a different pre-training task that alleviates these disadvantages. Instead of masking tokens, ELECTRA replaces some input tokens with fakes sampled from a small generator network, and then distinguishes the original vs. replaced tokens. Therefore, we look forward to introducing a unified cloze-based model that combines advantages of both generative and discriminative models. We design three cloze-based SSL tasks which includes **U**nordered **M**asked **L**anguage **M**odeling (**UMLM**) , **R**eplaced **M**asked **T**oken **D**etection (**RMTD**), and **C**ontrastive **E**nergy-based **U**nmasked **T**oken **C**lozing (**CE-UTC**). We will describe them in Section III-B in detail. Meanwhile, to develop an analogous language model for molecule graphs, we also need to solve two problems, which is **What the "token" is** and **How to design the "Transformers" for molecules**. The tokens for the language modeling are sub-words or words, whereas the concept of the tokens is vague in the molecule setting. Thus, we devise the **S**tructural **F**ingerprint **T**okenization (**SFT**) strategy, which has almost the same expressive power as Weisfeiler-Leman (WL) Test [13] and follows the same distribution as word frequency in natural language. We will prove our claims and state detailed information in Section III-A. For the latter problem, since the standard Transformer block is designed to process test data, it may not be suitable for the processing of molecular graph data. In light of the insights shown in graph convolution [14], we adjust the simple residual connection in the Transformer block [15] into a special shortcut connection to calibrate the hidden representations learned in deeper Transformer blocks using the local structural feature encoded in a shortcut connection with renormalization trick. We name the special short cut connection as **N**ormalized **G**raph **R**aw **S**hortcut-connection (**NGRS**). We will illustrate the effectiveness of the two strategies in the later experimental section. The frequency distribution of circular identifiers follows the Zipf's law as the word frequency in natural language, which can better support the repositioning of the Transformer architecture to molecule graph data.

To demonstrate the generalizability and superiority of our MolCloze model, we apply it on various downstream molecular property prediction tasks with the limited parameters and pre-trained dataset size, finding that our MolCloze model outperforms the state-of-the-art baselines on most downstream tasks. Our contributions of this paper could be summarized as:

- We devise two novel strategies **SFT** and **NGRS**, aiming to adapt transformer architecture for molecule data.
- We elaborately design three cloze-based SSL tasks **UMLM** , **RMTD** and **CE-UTC** to pre-train our model with massive unlabeled molecule data.
- We conduct extensive experiments, showing that MolCloze has better performance compared with a series of state-of-the-art methods.

II. Preliminary

1) Transformer architecture and Attention mechanism: The attention mechanism is the main building block of Transformer. We focus on multi-head attention, which stacks several scaled dot-product attention layers together and allows parallel running. One scaled dot-product attention layer takes a set of queries, keys, values $(\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v})$ as inputs. Then it computes the dot products of the query with all keys, and applies a softmax function to obtain the weights on

the values. By stacking the set of $(\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v})$s into matrices $(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V})$, it admits highly optimized matrix multiplication operations. Specifically, the outputs can be arranged as a matrix $Att(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = softmax(\boldsymbol{Q}\boldsymbol{K}^{\boldsymbol{T}}/\sqrt{d})V$, where d is the dimension of $\boldsymbol{q}$ and $\boldsymbol{k}$. Suppose we arrange k attention layers into the multi-head attention, then its output matrix can be written as $MultiH(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = Concat(H_1, \ldots, H_k)\boldsymbol{W}$, where $H_i = Att(\boldsymbol{Q}\boldsymbol{W}_i^{\boldsymbol{Q}}, \boldsymbol{K}\boldsymbol{W}_i^{\boldsymbol{K}}, \boldsymbol{V}\boldsymbol{W}_i^{\boldsymbol{V}})$, $\boldsymbol{W}_i^{\boldsymbol{Q}}$, $\boldsymbol{W}_i^{\boldsymbol{K}}$, and $\boldsymbol{W}_i^{\boldsymbol{V}}$ are the projection matrices of head i.

2) Circular fingerprints: Circular fingerprints [16] are a refinement of the Morgan algorithm [4], designed to encode which substructures are present in a molecule in a way that is invariant to atom-relabeling. Circular fingerprints (1) generate each layer's features by applying a fixed hash function to the concatenated features of the neighborhood in the previous layer, and (2) then treat the results of these hashes as integer indices, where one bit is written to the fingerprint vector at the index given by the feature vector at each node in the graph. Ignoring collisions, each index of the fingerprint denotes the presence of a particular substructure. The size of the substructures represented by each index depends on the depth of the network. Thus the number of layers is referred to as the "radius" of the fingerprints.

3) Weisfeiler-Lehman test: The Weisfeiler-Lehman (WL) test of graph isomorphism [13] is an effective and computationally efficient test that distinguishes a broad class of graphs. Testing graph isomorphism refers to asking whether two graphs are topologically identical. This is a challenging problem because no polynomial-time algorithm is discovered yet [17]. The WL test iteratively (1) aggregates the labels of nodes and their neighborhoods, and (2) hashes the aggregates the aggregated labels into *unique* new labels. The algorithm decides that two graphs are non-isomorphic if at some iteration the labels of the nodes between the two graphs differ.

III. Methodology

In this section, we firstly introduce our proposed two strategies **SFT** and **NGRS** and prove our claims by bridging the gap between circular fingerprints and the WL test, which is the upper bound of the expressive power of general message-passing GNNs. Then, we describe the detailed information of our three well-designed cloze-based SSL tasks for molecular structure pre-training step by step.

A. Adapting Transformer for Molecule Data: Strategies

Though the original transformer-style architecture has been tailored and utilized widely for various NLP tasks and operates over sequence data, the core component - Transformer block is in essence a powerful encoder for unordered data. From the viewpoint of message passing paradigm [6], the Transformer architecture is a special variant of GNNs with multi-head dot-product attention mechanism as the aggregator and position-wise fully-connected neural network as the inter-layer updater. Molecule data are in essence graph-structured data with intricate connectivity patterns, which can be processed with the Transformer block without position embedding. Thus, it is applicable and valuable to exploit the Transformer architecture for developing an analogous pre-trained model for molecule.

However, there still remains two issues, which are - How to **define the token** and **design the Transformer block** for molecule. To tackle these problems, we develop the **S**tructural **F**ingerprint **T**okenization (**SFT**) and **N**ormalized **G**raph **R**aw **S**hortcut-connection (**NGRS**), respectively. We will expand on the detailed information about the two strategies in the following parts.

1) Structural Fingerprint Tokenization (SFT) Strategy: The Transformer architecture is firstly proposed to operate over sequence data. The token sequences must be mapped into different embedding vector spaces in terms of token, position and segment embedding before fed into the model. However, in out setting, we would like our MolCloze model to be trained over molecule graph data. With this purpose, we need to clarify what the input tokens are. A straightforward approach is to define discrete tokens as atoms. This approach will bring a vocabulary with a very limited size because the total number of atoms is only 118 nowadays. More concretely, as for organic molecules, the vocabulary size will be smaller than 118. Due to the scarce atom vocabulary, the model can not capture sufficient useful information within molecule data in this setting. We must define tokens with a vocabulary of moderate size.

[18] suggests that the structural fingerprints can be a powerful feature to calculate attention distribution involved in Graph Attention Networks (GAT) [19], which exploits the same multi-head self-attention mechanism to capture useful information as the Transformer block. Their experiments show improvement of model performance with attention mechanism incorporating structural fingerprints. This finding motivate us to define tokens as structural fingerprints.

Cheminformatics community has defined many molecular structural fingerprints. In our study, we choose the intermediate atom-relabeling results of circular fingerprints, which is one of 2D-descriptors, as the input tokens parameterized by a pre-defined radius. The radius hyper-parameter defines the breadth of chemical localization. We name these defined tokens as circular identifiers (CI). Our main motivation to use circular identifiers can be attributed to their powerful expressive ability, which is as same as the WL test. This viewpoint is intuitive and can be proved straightforwardly.

a) Relationship between WL and CI:

Definition 1. (WL node coloring) Let (G, l) be a labeled graph, $\{\{\ldots\}\}$ denote a multiset, Σ be arbitrary codomain, and $HASH\{\ldots\}$ be a bijective mapping. In each iteration, $t \geq 0$, the WL computes a node coloring $c_l^{(t)} : V(G) \to \Sigma$, which depend on the coloring from the previous iteration. Formally, in iteration 0, $c_l^{(0)}$ is set to l. Now in iteration $t > 0$, the node coloring by the WL is defined as: $c_l^{(t)}(v) = HASH\left(\left(c_l^{(t-1)}(v), \{\{c_l^{(t-1)}(u) | u \in N(v)\}\}\right)\right)$ [20].

Definition 2. (CI atom relabeling) Let (M, f) be a molecular graph with atom features and radius R, $Concat\{\ldots\}$ denote

the concatenation operator, Σ be arbitrary codomain, and $HASH\{\ldots\}$ be a bijective mapping. In each iteration, $0 \leq t \leq R-1$, the CI computes an atom relabeling $h_f^{(t)} : A(M) \rightarrow \Sigma$, which depend on the labels from the previous iteration. Formally, in iteration 0, $h_f^{(1)}$ is set to f. Now in iteration $1 < t \leq R-1$, the atom relabeling by the CI is defined as: $h_f^{(t)}(a) = HASH\left(\left(h_f^{(t-1)}(a), Concat\{h_f^{(t-1)}(r)|r \in N(a)\}\right)\right)$ [21].

As shown in Definition 1 and 2, the main difference between the WL node coloring and the CI atom relabeling is the concatenation operator over the multiset labeled by neighboring nodes(atoms). However, this operation will not influence the bijective mapping $Hash\{\ldots\}$, which means that the WL coloring and the CI atom relabeling perform essentially the same computations. Therefore, we have our first theoretical result which shows that the CI has the same expressive power as the WL algorithm.

Theorem 1. Let $\equiv$ denote two equivalent operations, (M, f, l) be a labeled molecular graph with atom features. If all choices of initial colorings $c_l^{(0)} := l$ are consistent with f, we have $c_l^{(t)} \equiv h_f^{(t)}$.

In many previous research works [22], the expressive power of GNNs have been proved to be upper-bounded by the WL algorithm. A basic message-passing GNN framework can be formulated as follows:

$$g^{(t)}(v) = g_{merge}^{W_1}\left(g^{(t-1)}(v), g_{aggr}^{W_2}\left(\{\{g^{(t-1)}(w)|w \in N(v)\}\}\right)\right) \quad (1)$$

where $g_{aggr}^{W_2}$ aggregates over the set of neighborhood features and $g_{merge}^{W_1}$ merges the node's representations from $(t-1)_{th}$ step with the computed neighborhood features. Both $g_{merge}^{W_1}$ and $g_{aggr}^{W_2}$ may be arbitrary differentiable, permutation-invariant functions (e.g., neural networks). Formally, for every encoding of the labels $l(v)$ as vectors $g^{(0)}(v)$, and for every choice of $W_1^{(t)}$ and $W_2^{(t)}$, we have that the node coloring $c_l^{(t)}$ of the WL algorithm always refines the coloring $g^{(t)}$ induced by a GNN model parameterized by $W_1^{(t)}$ and $W_2^{(t)}$.

Lemma 2. Let (G, l) be a labeled graph, and $A \sqsubseteq B$ denote A refines B. For all $t \geq 0$ and all choices of initial node features $g^{(0)}$ consistent with l, and parameters $W_1^{(t)}$ and $W_2^{(t)}$, we have $c_l^{(t)} \sqsubseteq g^{(t)}$ [20].

According to Theorem 1 and Lemma 2, we can conclude that, with the same expressive ability as the WL algorithm, the CI performs powerful expression than GNNs in general cases.

Corollary 3. Let (M, f, l) be a labeled molecular graph with atom features, and $A \sqsubseteq B$ denote A refines B. If all choices of initial node colorings $c_l^{(0)} := l$ are consistent with f and $g^{(0)}$, we have $h_f^{(t)} \sqsubseteq g^{(t)}$.

However, GNNs have parameterized pooling and update operation, which can help to obtain better molecule-level representations on large-scale pre-train datasets and fine-tune more easily for downstream tasks. Therefore, with highly expressive but unparameterized CI, there must be a powerful neural network model to fit them as fully as possible. Obviously, Transformer architecture can play this role perfectly.

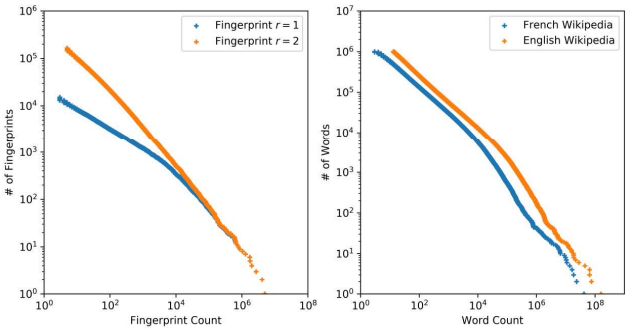

Figure 1. Frequency distribution of circular identifiers with different radius and that of English and French Wikipedia. They both obey Zipf's law, implying that the recent language pre-training techniques can be repositioning to the pre-training techniques for molecule data.

Apart from the benefit for the powerful expressive power and the multi-head attention mechanism, the circular identifiers also take the following advantages: **(1)** the frequency distribution of CI follows the Zipf's law as the word frequency in natural language (shown in Figure 1), which can better support the repositioning of the Transformer architecture to molecule graph data; **(2)** it has an easy implementation in pre-defined libraries like RDKit [23]; **(3)** the variants of circular fingerprints built on top of CI have been widely used with machine learning algorithms, obtaining considerable performance in some cheminformatics tasks.

Owing to the benefits listed above, we exploit the circular identifiers parameterized by a pre-defined radius as the tokens. All of the CI appearing in the pre-training dataset will be hashed to a vector with a fixed length. Note that we use token and CI interchangeably in the former part, we will use token to refer to CI in the later section.

2) Normalized Graph Raw Shortcut-connection (NGRS) Strategy: Transformer is proposed to handle unordered data essentially [15]. In order to adapt it to molecule data, a naive approach is to simply add attention masking information to the multi-head attention module according to the adjacency matrices of the molecular graphs and keep other parts unchanged as follows:

$$\boldsymbol{H}_{att} = Att(\boldsymbol{H}^{(l)}) = softmax(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}} \odot \boldsymbol{A})\boldsymbol{V} \quad (2)$$

However, this naive approach will be at disadvantage due to the over-smoothing phenomenon same as in deep GNNs [14]. And it will also introduce more risks of incorporating noise information in deeper Transformer blocks. Thus, we devise a special shortcut-connection to calibrate hidden representations in deeper Transformer blocks, aiming at incorporating broader neighborhood information and decrease the risk of being corrupted by noisy information. We name the special shortcut-connection as **N**ormalized **G**raph **R**aw **S**hortcut-connection (**NGRS**). Also, The **NGRS** strategy can be seen as an ad-

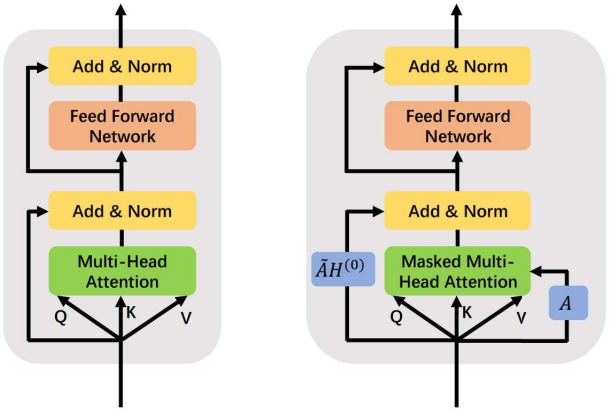

Figure 2. Comparison between original (L) and our (R) Transformer block.

justment of the original residual connection adopted in the standard Transformer blocks:

$$\boldsymbol{H}_{mha} = \tilde{\boldsymbol{A}}\boldsymbol{H}^{(0)} + LAYERNORM(MHA(\boldsymbol{H}^{(l)})) \quad (3)$$

where $\tilde{\boldsymbol{A}} = \boldsymbol{I} + \boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{D}^{-\frac{1}{2}}$, $\boldsymbol{A}$ denotes the adjacency matrix of the input molecular graph, $\boldsymbol{H}^{(0)}$ is the hidden representation before the first Transformer block, namely, the raw attribute information of the token. This strategy can be benefit for the stability of feed forward computation [14]. We compare the original one with ours in Figure 2.

B. Self-supervised Tasks for Pre-training

The success of the pre-training model crucially depends on the design of self-supervision tasks. In recent research works [11], [12], [24], the cloze task achieves excellent performance in language pre-training models. It helps these models to learn text representations more effectively by predicting the identity of a token given its surrounding context. Therefore, from the unifying standpoint, we propose three new cloze-based self-supervised tasks for molecule pre-training: **U**nordered **M**asked **L**anguage **M**odeling (**UMLM**) , **R**eplaced **M**asked **T**oken **D**etection (**RMTD**), and **C**ontrastive **E**nergy-based **U**nmasked **T**oken **C**lozing (**CE-UTC**), which are sketched in Figure 3.

1) Unordered Masked Language Modeling (UMLM): Inspired by generative models like the most famous BERT [11], we adapt the original masked language modeling tasks for molecule data, namely, UMLM. The UMLM task uses the same setting as the original version of BERT. We also mask out the approximate 15% tokens to train the deep bidirectional Transformer architecture. Of the masked tokens, 80% are replaced with the special $[MASK]$ token, 10% are replaced with a random token and the remaining 10% tokens are kept unchanged. The goal of the masked language modeling tasks is to predict the masked tokens according to the observed ones. The final layer of the masked language modeling task will generate a probability distribution over the discrete tokens, and the self-supervised tasks is optimized the cross-entropy between the predicted probability distribution and the true distribution.

Considering a generator $\mathcal{G}$, which is primarily established with an encoder (e.g. Transformer) that maps input tokens into contextualized vector representations, the probability output for generating a particular token x_t with a softmax layer is:

$$p_{\mathcal{G}}(x_t|\boldsymbol{x}) = exp(e(x_t)^T h_{\mathcal{G}}(\boldsymbol{x})_t) / \sum_{x' \in \mathcal{V}} exp(e(x')^T h_{\mathcal{G}}(\boldsymbol{x})_t) \quad (4)$$

where e denotes token embeddings, $\boldsymbol{x}$ is a sequence of input tokens $[x_1, \ldots, x_n]$, $h(\ldots)$ refers to the output vector representations from the encoder and $\mathcal{V}$ represents the vocabulary. Then, given the masked positions $\boldsymbol{m} = [m_1, \ldots, m_k]$ and masked sequence $\boldsymbol{x}_m$, the loss function of UMLM can be formulated as:

$$\mathcal{L}_{UMLM}(\boldsymbol{x}, \theta_{\mathcal{G}}) = \mathbb{E}\left(\sum_{i \in \boldsymbol{m}} -\log p_{\mathcal{G}}(x_i|\boldsymbol{x}_m)\right) \quad (5)$$

2) Replaced Masked Token Detection (RMTD): Recently, many researches [12], [24] demonstrate discriminative self-supervised tasks can help to pre-train the models more fully so that they can achieve higher accuracy on downstream tasks. Following this idea, we design a RMTD task to detect which masked token is generated correctly or replaced. Considering another neural networks, a discriminator $\mathcal{D}$, which is also primarily established with an encoder, it can predict whether the token x_t is "real" (only the masked positions in our settings), i.e., that it comes from the data rather than the generator distribution: $\mathcal{D}(\boldsymbol{x}, t) = sigmoid(w^T h_{\mathcal{D}}(\boldsymbol{x})_t)$, where $sigmoid(\cdot)$ denotes a sigmoid layer. The discriminator is trained to distinguish tokens in the data from tokens that have been replaced by generator samples. More specifically, with a corrupted sequence $\boldsymbol{x}_c$ in which we replace the masked-out tokens with generated samples and ignore the unmasked tokens, the discriminator is trained to predict which tokens in $\boldsymbol{x}_c$ match the original input $\boldsymbol{x}$. Therefore, formally the loss function of RMTD is:

$$\begin{aligned}\mathcal{L}_{RMTD}(x, \theta_{\mathcal{D}}) = \mathbb{E}\Bigg(&\sum_{t=1}^{n} -\mathbb{1}(x_{c,t} = x_t)\log \mathcal{D}(\boldsymbol{x}_c, t) \\ &- \mathbb{1}(x_{c,t} \neq x_t)\log(1 - \mathcal{D}(\boldsymbol{x}_c, t))\Bigg)\end{aligned} \quad (6)$$

It should be noted that the Transformer blocks of the discriminator share parameters with the other of the generator, which improves pre-training efficiency [12].

3) Contrastive Energy-based Unmasked Token Clozing (CE-UTC): For the large amount of unmasked tokens in sequences, we hope there can be another auxiliary self-supervised task which may improve the pre-training efficiency and meanwhile accelerate the convergence of UMLM and RMTD. Thus, we propose to design an energy-based approach according to its computational simplification [25]. On the top of the weights-sharing Transformer blocks, we assign each given position t an energy score $E(\boldsymbol{x})_t = \boldsymbol{w}^T \boldsymbol{h}(\boldsymbol{x})_t$ using a learned weight vector $\boldsymbol{w}$. Then the un-normalized output can be defined as $\hat{p}_{\mathcal{E}}(x_t|\boldsymbol{x}_{\setminus t}) = exp(-E(\boldsymbol{x})_t)$. An

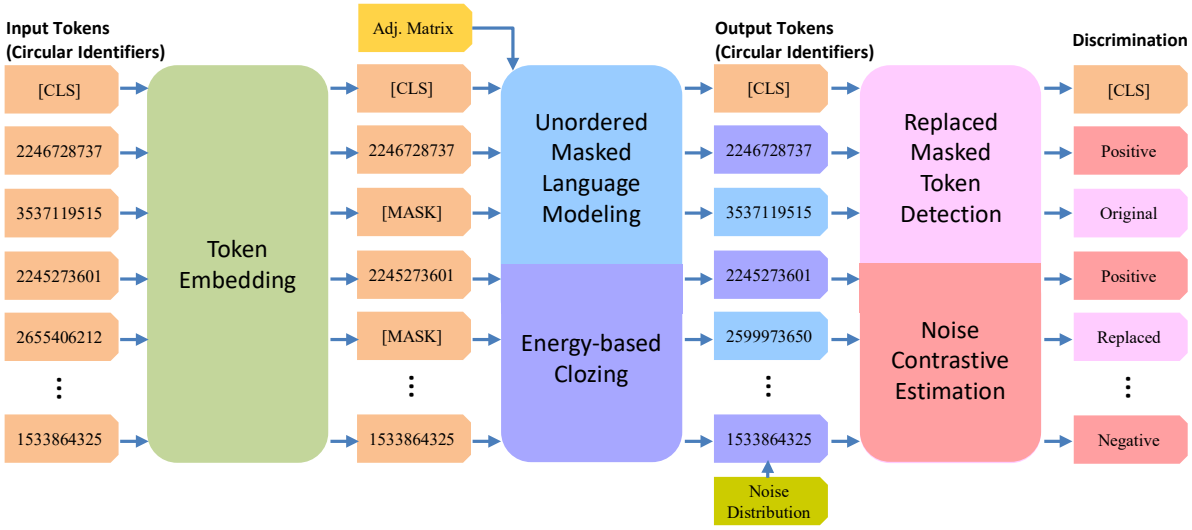

Figure 3. Overview of the designed self-supervised tasks of MolCloze. (Noted that the Contrastive Energy-based Unmasked Token Clozing (CE-UTC) is grouped by the Energy-based Clozing and Noise Contrastive Estimation.)

efficient way to train this un-normalized model $\mathcal{E}$ is Noise-Contrastive Estimation (NCE) [26]. NCE learns the parameters of a model by defining a binary classification task where samples from the data distribution have to be distinguished from samples generated by a noise distribution. Hence, we use a neural network $\mathcal{Q}$ also on the top of the weights-sharing Transformer blocks to produce the noise distribution $q(x_t|\boldsymbol{x}_{\setminus t}) = softmax(\boldsymbol{W}\boldsymbol{h}(\boldsymbol{x})_t)$. We replace x_t with a noise token $\hat{x}_t$ sampled from q in a sequence to form a negative sample. Then, with k negatives for every n positives, we train a classifier to distinguish them through the NCE objective:

$$\begin{aligned}&\mathcal{L}_{CE-UTC}(\boldsymbol{x},\theta_{\mathcal{E}},\theta_{\mathcal{Q}})\\&= n\cdot\mathop{\mathbb{E}}_{\boldsymbol{x},t}\left[-\log\frac{n\cdot\hat{p}_{\mathcal{E}}(x_t|\boldsymbol{x}_{\setminus t})}{n\cdot\hat{p}_{\mathcal{E}}(x_t|\boldsymbol{x}_{\setminus t}+k\cdot q(x_t|\boldsymbol{x}_{\setminus t})}\right]\\&+k\cdot\mathop{\mathbb{E}}_{\substack{\boldsymbol{x},t\\ \hat{x}_t\sim q}}\left[-\log\frac{k\cdot q(x_t|\boldsymbol{x}_{\setminus t})}{n\cdot\hat{p}_{\mathcal{E}}(x_t|\boldsymbol{x}_{\setminus t}+k\cdot q(x_t|\boldsymbol{x}_{\setminus t})}\right]\end{aligned}\tag{7}$$

This loss is optimized to minimization when $\hat{p}_{\mathcal{E}}$ matches the data distribution p_{data} [26]. Besides, this property can be also seen as a self-normalization (reaching to 1) of $\sum_{x'\in\mathcal{V}} exp(-E(R(\boldsymbol{x},t,x'))_t)$ along with optimization of the model, where $R(\boldsymbol{x},t,x')$ denotes replacing the token at position t with x'.

Overall, we minimize the combined loss over a large molecule corpus $\mathcal{X}$:

$$\begin{aligned}\min_{\theta_{\mathcal{G}},\theta_{\mathcal{D}},\theta_{\mathcal{E}},\theta_{\mathcal{E}}}\sum_{\boldsymbol{x}\in\mathcal{X}}&\mathcal{L}_{UMLM}(\boldsymbol{x},\theta_{\mathcal{G}})+\alpha\mathcal{L}_{RMTD}(x,\theta_{\mathcal{D}})\\&+\beta\mathcal{L}_{CE-UTC}(\boldsymbol{x},\theta_{\mathcal{E}},\theta_{\mathcal{Q}})\end{aligned}\tag{8}$$

IV. Experiments

A. Datasets

1) Self-training dataset: To train our model, we need a large-scale unlabeled dataset to perform self-supervised training. In this study, we construct a dataset with 1 million molecule samples from ZINC. Detailed hyper-parameter configuration is in Table I.

Table I
Hyper-parameter configuration of MolCloze

Parameter	Meaning	Value
d	hidden size of Transformer representations	768
L	number of Transformer blocks	4
m	number of attention heads	12
r	token (circular identifier) radius	2
E	number of epochs	5
B	batch size	32
α	coefficient of RMTD loss	1
β	coefficient of CE-UTC loss	20

2) Fine-tuning tasks and datasets: In order to prove the effectiveness of our MolCloze model, we select six molecule property prediction tasks from the MoleculeNet benchmark [3]. Similar to their implementation, we also use scaffold-split to avoid obtaining overly optimistic results.

B. Baselines

We comprehensively evaluate our MolCloze model against baselines which have gained increasing popularity in the domain of molecular property prediction. These baselines include 8 popular baselines from MoleculeNet [3] and several state-of-the-arts (SOTAs) approaches. Among them, TF_Robust is a DNN-based multi-task framework taking the molecular fingerprints as the input. GraphConv, Weave and SchNet are three graph convolutional models. MPNN and its variants DMPNN and MGCN are models considering the edge features during message passing. AttFP is an extension of the graph attention network. Specifically, to demonstrate the power of our well-designed self-supervised strategies and tasks, we also compare MolCloze with three pre-trained models: Mol2vec [9], N-Gram [27], Pre-GIN [28] and GROVER [29].

Table II
THE PERFORMANCE RESULTS COMPARED TO BASELINES (ROC-AUC(%)).

Dataset	BBBP	BACE	Tox21	ToxCast	ClinTox	Sider
#Molecules	2039	1513	7831	8575	1478	1427
TF_Robust	86.0	82.4	69.8	56.4	76.5	60.7
GraphConv	87.7	85.4	77.2	66.7	84.5	54.3
Weave	83.7	79.1	74.1	65.5	82.3	54.3
SchNet	84.7	75.0	76.7	67.1	71.7	59.5
MPNN	91.3	81.5	80.8	66.5	87.9	59.5
DMPNN	91.9	85.2	81.6	67.8	89.7	63.2
MGCN	85.0	73.4	70.7	65.8	63.4	55.2
AttFP	90.8	86.3	80.7	59.1	93.3	60.5
Mol2vec	89.5	83.3	75.9	66.5	82.6	60.1
N-GRAM	91.2	87.6	76.9	-[1]	85.5	63.2
Pre-GIN	91.5	85.1	81.1	67.4	76.2	61.4
GROVER[2]	93.5	88.6	**81.8**	**68.9**	90.2	63.1
MolCloze	**94.0**	**90.9**	**81.8**	**68.6**	**93.5**	**67.5**

C. Results on Downstream Tasks

Table II demonstrates model performance (averaged across 10 runs) across different molecular property prediction tasks. The experimental results indicate: (1) Overall, our MolCloze model achieves superior performance than other baseline approaches based on GNNs by a considerable margin. (2) Traditional supervised learning methods that learn molecular representations from scratch for each task under-perform our MolCloze model and other baseline approaches with pre-training strategies in the tasks with limited dataset size (BBBP. BACE, ClinTox, and Sider). (3) The TF_Robust model takes on dramatically varying predictive performances on different downstream tasks. This phenomenon has a straightforward interpretation that the chemical fingerprints directly applied as molecular representations is only useful for a small subset of downstream tasks such as BBBP and BACE. For the remaining datasets, they cannot capture the related features, leading to inferior performance, which is the very disadvantages of the molecular descriptor-based methods. (4) Pre-trained models achieve relatively better performance on all datasets, demonstrating the priority of self-supervised learning for molecule data. Among them, Mol2vec and Pre-GIN slightly underperform GROVER because of their shallow architecture with less number of parameters. As shown in Table II, GROVER is strong. However, our MolCloze outperforms it on most of the datasets, especially those with limited labeled data (BBBP. BACE, ClinTox, and Sider), because we use the circular identifiers, which have more powerful expressive power than their GNNs-based framework, to encode molecular substructures.

In order to probe the relationship between the pre-trained embeddings and some common phenomena in organic chemistry, we elaborately selected 18 typical compounds as examples. We computed the pairwise cosine similarity based on pre-trained embeddings between these 18 substances, and showed the visualization results in Figure 4. The heat-map in Figure 4 can indicate the following chemical phenomena:

- **Structural similarity of derivatives of the same series.** The similarity between derivates of the same series (consecutive three compounds) is significantly higher than those across different series.
- **Interchangeability of halogen atoms.** Compounds 1-3 are more similar and the main structural difference lies in the type of halogen atom. This structural difference do not influence the pairwise similarity.
- **Polarity.** Ortho-dichlorobenzene (compound 4) is more similar to meta-dichlorobenzene (compound 5) than para-dichlorobenzene (compound 6) for the latter is non-polar.
- **Aromaticity.** Compared with other compounds, the similarity between compounds 1-6, 10-12 is higher, because these compounds are all aromatic. For compounds 7,8,9 and 16-18, although they also have a ring structure where single and double bonds occur simultaneously, they do not satisfy the Huckel rule. Therefore, they are not aromatic and quite different with those aromatic substances.
- **Conjugation of functional groups.** The similarity between compounds 16-18 and compounds 13-15 is low because the ring structure in compounds 13-15 does not satisfy the conjugated system. The opposite example is: the similarity between compounds 16-18 and substance 7-9 is high, because these six compounds have a conjugated system, so the chemical properties are similar.

[1]This result is too time-consuming to be finished in time.

[2]We pre-train it with 1 million unlabeled molecules as same as MolCloze.

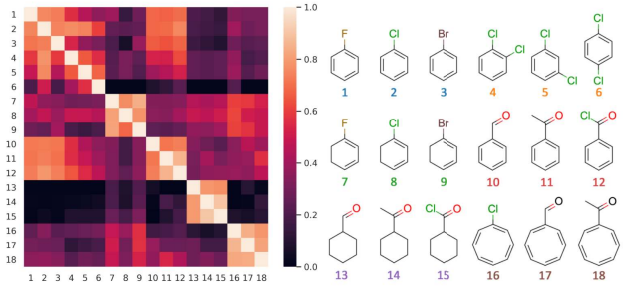

Figure 4. The pairwise similarity heat-map of the specified molecules. The molecule-level embedding vectors are provided by our MolCloze model. The comparison results reflects some empirical phenomena in organic chemistry.

Table III
ABLATION STUDY RESULTS. ("W/" AND "W/O" ARE THE ABBREVIATIONS FOR "WITH" AND "WITHOUT" RESPECTIVELY.)

Dataset	BBBP	BACE	Tox21	ToxCast	ClinTox	Sider
w/o SFT	83.6	69.4	69.7	66.6	56.2	55.7
w/o NGRS	89.2	84.0	75.2	64.7	84.2	55.9
w/ UMLM	90.4	85.9	78.3	67.1	85.3	62.7
w/o CE-UTC	92.2	87.1	79.9	67.8	88.4	65.5
MolCloze	**94.0**	**90.9**	**81.8**	**68.6**	**93.5**	**67.5**

D. Ablation Studies on Design Choices of MolCloze

1) How useful are the proposed strategies?: To investigate the contribution of the two proposed strategies to adapt Transformer architecture for molecule data, we conduct ablation studies for them with the same experimental settings.

a) Analysis of SFT: Table III compares the fine-tuned performance results on downstream datasets conducted by our MolCloze model and its naive version without our SFT strategy. We observe that out SFT strategy boost the fine-tune

performance on molecular property prediction tasks by a large margin. The experimental results shown in Table III verifies the correctness of our speculation mentioned in Section III-A as well as the effectiveness of our SFT strategy. The easy pre-training induced by the elimination of our SFT strategy cannot make the naive version of our MolCloze model learn sufficient useful semantic information for a wide array of molecular property prediction tasks.

b) Analysis of NGRS: Table III also compares our MolCloze model with its naive version with the same residual connection strategy as the standard Transformer blocks. We observe that our NGRS strategy incorporating normalized graph adjacency information as well as raw attribute information can effectively boost the fine-tune performance on molecular property prediction tasks by a considerable margin, which proves that our NGRS strategy is effective.

2) How powerful are the self-supervised learning tasks?: To verify the effectiveness of our three well-designed self-supervised learning tasks, we conduct ablation studies across them, all of which follow the same experimental settings. As shown in Table III, the model has achieved fairly good performance only with the UMLM task, demonstrating the priority of the cloze-style task. Besides, the model improves a lot by strengthening itself with a discriminative task RMTD. The table also shows that our MolCloze model is boosted by a large margin due to the added CE-UTC task, which sufficiently uses unmasked tokens to optimize and pre-train our MolCloze model better. We also find RMTD and CE-UTC can help the UMLM loss to converge better.

V. Conclusion

We develop a unified cloze-style pre-training model for molecule data. With strategies adapting Transformer for molecule data, well-designed self-supervised tasks, and largely-expressive architecture, our MolCloze model can learn rich implicit information from the enormous unlabelled graphs. Experiments on multiple datasets, diverse downstream tasks and various baselines show that the new pre-training strategy generalizes better than other supervised or pre-trained models.

References

[1] I. Wallach, M. Dzamba, and A. Heifets, "Atomnet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery," *arXiv preprint arXiv:1510.02855*, 2015.

[2] K. Yang, K. Swanson, W. Jin, C. Coley, P. Eiden, H. Gao, A. Guzman-Perez, T. Hopper, B. Kelley, M. Mathea *et al.*, "Analyzing learned molecular representations for property prediction," *Journal of chemical information and modeling*, vol. 59, no. 8, pp. 3370–3388, 2019.

[3] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande, "Moleculenet: a benchmark for molecular machine learning," *Chemical science*, vol. 9, no. 2, pp. 513–530, 2018.

[4] H. L. Morgan, "The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service." *Journal of Chemical Documentation*, vol. 5, no. 2, pp. 107–113, 1965.

[5] D. Weininger, A. Weininger, and J. L. Weininger, "Smiles. 2. algorithm for generation of unique smiles notation," *Journal of chemical information and computer sciences*, vol. 29, no. 2, pp. 97–101, 1989.

[6] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1263–1272.

[7] J. Ma, R. P. Sheridan, A. Liaw, G. E. Dahl, and V. Svetnik, "Deep neural nets as a method for quantitative structure–activity relationships," *Journal of chemical information and modeling*, vol. 55, no. 2, pp. 263–274, 2015.

[8] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *arXiv preprint arXiv:1310.4546*, 2013.

[9] S. Jaeger, S. Fulle, and S. Turk, "Mol2vec: unsupervised machine learning approach with chemical intuition," *Journal of chemical information and modeling*, vol. 58, no. 1, pp. 27–35, 2018.

[10] W. L. Taylor, ""cloze procedure": A new tool for measuring readability," *Journalism quarterly*, vol. 30, no. 4, pp. 415–433, 1953.

[11] 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.

[12] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," *arXiv preprint arXiv:2003.10555*, 2020.

[13] B. Weisfeiler and A. Leman, "The reduction of a graph to canonical form and the algebra which appears therein," *NTI, Series*, vol. 2, no. 9, pp. 12–16, 1968.

[14] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[16] R. C. Glen, A. Bender, C. H. Arnby, L. Carlsson, S. Boyer, and J. Smith, "Circular fingerprints: flexible molecular descriptors with applications from physical chemistry to adme," *IDrugs*, vol. 9, no. 3, p. 199, 2006.

[17] L. Babai, "Graph isomorphism in quasipolynomial time," in *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, 2016, pp. 684–697.

[18] K. Zhang, Y. Zhu, J. Wang, and J. Zhang, "Adaptive structural fingerprints for graph attention networks," in *International Conference on Learning Representations*, 2019.

[19] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[20] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe, "Weisfeiler and leman go neural: Higher-order graph neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 4602–4609.

[21] D. Duvenaud, D. Maclaurin, J. Aguilera-Iparraguirre, R. Gómez-Bombarelli, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," *arXiv preprint arXiv:1509.09292*, 2015.

[22] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" *arXiv preprint arXiv:1810.00826*, 2018.

[23] G. Landrum, "Rdkit documentation," *Release*, vol. 1, no. 1-79, p. 4, 2013.

[24] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Pre-training transformers as energy-based cloze models," *arXiv preprint arXiv:2012.08561*, 2020.

[25] Z. Ma and M. Collins, "Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency," *arXiv preprint arXiv:1809.01812*, 2018.

[26] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 297–304.

[27] S. Liu, M. F. Demirel, and Y. Liang, "N-gram graph: Simple unsupervised representation for graphs, with applications to molecules," *arXiv preprint arXiv:1806.09206*, 2018.

[28] W. Hu, B. Liu, J. Gomes, M. Zitnik, P. Liang, V. Pande, and J. Leskovec, "Strategies for pre-training graph neural networks," *arXiv preprint arXiv:1905.12265*, 2019.

[29] Y. Rong, Y. Bian, T. Xu, W. Xie, Y. Wei, W. Huang, and J. Huang, "Self-supervised graph transformer on large-scale molecular data," *Advances in Neural Information Processing Systems*, vol. 33, 2020.