
MOTIF-AWARE ATTRIBUTE MASKING FOR MOLECULAR GRAPH PRE-TRAINING

Anonymous authors

Paper under double-blind review

ABSTRACT

1 Attribute reconstruction is used to predict node or edge features in the pre-training
2 of graph neural networks. Given a large number of molecules, they learn to capture
3 structural knowledge, which is transferable for various downstream property
4 prediction tasks and vital in chemistry, biomedicine, and material science. Previous
5 strategies that randomly select nodes to do attribute masking leverage the
6 information of local neighbors. However, the over-reliance of these neighbors
7 inhibits the model’s ability to learn long-range dependencies from higher-level
8 substructures. For example, the model would learn little from predicting three
9 carbon atoms in a benzene ring based on the other three but could learn more from
10 the inter-connections between the functional groups, or called chemical motifs. To
11 explicitly determine inter-motif knowledge transfer of pre-trained model, we define
12 inter-motif node influence measures. Then, we propose and investigate motif-
13 aware attribute masking strategies to capture long-range inter-motif structures by
14 leveraging the information of atoms in neighboring motifs. Once each graph is
15 decomposed into disjoint motifs, the features for every node within a sample motif
16 are masked. The graph decoder then predicts the masked features of each node
17 within the motif for reconstruction. We evaluate our approach on eight molecular
18 property prediction datasets and demonstrate its advantages.

19 1 INTRODUCTION

20 Molecular property prediction has been an important topic of study in fields such as physical chemistry,
21 physiology, and biophysics (Wu et al., 2017). It can be defined as a graph label prediction problem
22 and addressed by machine learning. However, graph learning models such as graph neural networks
23 (GNNs) must overcome issues in data scarcity, as the creation and testing of real-world molecules is
24 an expensive endeavor (Chang et al., 2022). To address labeled data scarcity, model pre-training has
25 been utilized as a fruitful strategy for improving a model’s predictive performance on downstream
26 tasks, as pre-training allows for the transfer of knowledge from large amounts of unlabeled data. The
27 selection of pre-training strategy is still an open question, with contrastive tasks (Zhu et al., 2021)
28 and predictive/generative tasks (Hu et al., 2020a) being the most popular methods.

29 Attribute reconstruction is one predictive method for graphs that utilizes masked autoencoders to
30 predict node or edge features (Hu et al., 2020a; Kipf & Welling, 2016; Xia et al., 2022). Masked
31 autoencoders have found success in vision and language domains (He et al., 2022; Devlin et al., 2018)
32 and have been adopted as a pre-training objective for graphs as the reconstruction task is able to
33 transfer structural pattern knowledge (Hu et al., 2020a), which is vital for learning specific domain
34 knowledge such as valency in material science. Additional domain knowledge which is important for
35 molecular property prediction is that of functional groups, also called chemical motifs (Pope et al.,
36 2019). *The presence and interactions between chemical motifs directly influence molecular properties,*
37 *such as reactivity and solubility* (Frechet, 1994; Plaza et al., 2014). Prior work in message passing for
38 quantum chemistry has shown that long-range dependencies are important for downstream prediction
39 in chemical domains (Gilmer et al., 2017). Therefore, to capture the interaction information between
40 motifs, it is important to transfer inter-motif structural knowledge and other long-range dependencies
41 during the pre-training of graph neural networks.

42 Unfortunately, the random attribute masking strategies used in previous work for graph pre-training
43 were not able to capture the long-range dependencies inherent in inter-motif knowledge (Kipf &

Random Attribute Masking

e.g., AttrPred (Hu et al. 2020), GraphMAE (Hou et al. 2022)

Motif-aware Attribute Masking

ours, named MoAMa

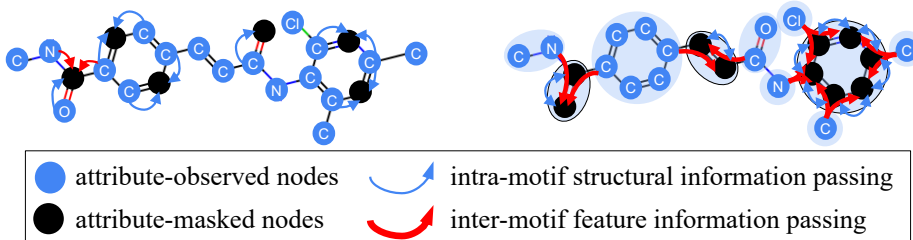


Figure 1: Our MoAMa masks every node in sampled motifs to pre-train GNNs. The full masking of a motif forces the GNNs to learn to (1) pass feature information across motifs and (2) pass local structural information within the motif. Compared to the traditional random attribute masking strategies, the motif-aware masking captures the most essential information to learn graph embeddings. Random masking would put most of the pre-training effort on passing the feature information within a motif, e.g., predicting two carbon nodes in a benzene ring based on the other four.

44 Welling, 2016; Hu et al., 2020b; Pan et al., 2019). That is because they rely on neighboring node
45 feature information for reconstruction (Hu et al., 2020a; Hou et al., 2022). Notably, leveraging
46 the features of local neighbors can contribute to learning important local information, including
47 valency and atomic bonding. However, GNNs heavily rely on the neighboring node’s features rather
48 than graph structure (Yun et al., 2021), and this over-reliance inhibits the model’s ability to learn
49 from motif structures as message aggregation will prioritize local node feature information due to
50 the propagation bottleneck (Alon & Yahav, 2021). For example, as shown on the left-hand side of
51 Figure 1, if only a (small) partial set of nodes were masked in several motifs, the pre-trained GNNs
52 would learn to predict the node types (i.e., carbon) of two atoms in the benzene ring based on the
53 features and structure of the other four carbon atoms in the ring, limiting the knowledge transfer
54 of long-range dependencies. To measure the inter-motif knowledge transfer of graph pre-training
55 strategies, we define five inter-motif influence measurements and report our findings in Sec. 6.

56 Recent successes in vision and language domains have shown the utility of masking semantically
57 related regions, such as pixel batches (Li et al., 2022; Xie et al., 2022; He et al., 2021) and multi-token
58 spans (Levine et al., 2020; Sun et al., 2019; Joshi et al., 2020), and have demonstrated that a random
59 masking strategy is not guaranteed to transfer necessary inter-part relations and intra-part patterns
60 (Li et al., 2022). To better enable the transfer of long-range inter-part relations downstream, we
61 propose a novel semantically-guided masking strategy based on chemical motifs. In Figure 1, we
62 visually demonstrate our method for motif-aware attribute masking, where each molecular graph
63 is decomposed into disjoint motifs. Then the node features for each node within the motif will be
64 masked by a mask token. A graph decoder will predict the masked features of each node within the
65 motif as the reconstruction task. The benefits of this strategy are twofold. First, because all features
66 of the nodes within the motif are masked, our strategy reduces the amount of feature information
67 being passed within the motif and relieves the propagation bottleneck, allowing for the greater
68 transfer of inter-motif feature and structural information. Second, the masking of all intra-motif node
69 features explicitly forces the decoder to transfer intra-motif structural information. A novel graph
70 pre-training solution based on the **Motif-aware Attribute Masking** strategy, called **MoAMa**, is able
71 to learn long-range inter-motif dependencies with knowledge of intra-motif structure. We evaluate
72 our strategy on eight molecular property prediction datasets and demonstrate its improvement to
73 inter-motif knowledge transfer as compared to previous strategies.

74 2 RELATED WORK

75 **Molecular graph pre-training** The prediction of molecular properties based on graphs is important
76 (Wu et al., 2017). Molecules are scientific data that are time- and computation-intensive to
77 collect and annotate for different property prediction tasks (Liu et al., 2023). Many self-supervised
78 learning methods (Hu et al., 2020a; Hou et al., 2022; Zhang et al., 2021; Kim et al., 2022; Xia et al.,
79 2023) were proposed to capture the transferable knowledge from another large scale of molecules
80 without annotations. For example, AttrMask (Hu et al., 2020a) randomly masked atom attributes for

81 prediction. GraphMAE (Hou et al., 2022) pre-trained the prediction model with generative tasks to
82 reconstruct node and edge attributes. D-SLA (Kim et al., 2022) used contrastive learning based on
83 graph edit distance. These pre-training tasks could not well capture useful knowledge for various
84 domain-specific tasks since they fail to incorporate important domain knowledge in pre-training. A
85 great line of prior work (Zhang et al., 2021; Rong et al., 2020; Sun et al., 2021) used graph motifs
86 which are the recurrent and statistically significant subgraphs to characterize the domain knowledge
87 contained in molecular graph structures, e.g., functional groups. However, their solutions were
88 tailored to specific frameworks for either generation-based or contrast-based molecular pre-training.
89 Additionally, explicit motif type generation/prediction inherently does not transfer intra-motif struc-
90 tural information and is computationally expensive due to the large number of prediction classes. In
91 this work, we study on the strategies of attribute masking with the awareness of domain knowledge
92 (i.e., motifs), which plays an essential role in self-supervised learning frameworks (Xia et al., 2023).

93 **Masking strategies on molecules** Attribute masking of atom nodes is a popular method in graph
94 pre-training given its broad usage in predictive, generative, and contrastive self-supervised tasks (Hu
95 et al., 2020a;b; Hou et al., 2022; You et al., 2020; 2021). For example, predictive and generative
96 pre-training tasks (Hu et al., 2020a; Hou et al., 2022; Xia et al., 2023) mask atom attributes for
97 prediction and reconstruction. Contrastive pre-training tasks (You et al., 2020; 2021) mask nodes to
98 create another data view for alignment. Despite the widespread use of attribute masking in molecular
99 pre-training, there is a notable absence of comprehensive research on its strategy and effectiveness.
100 Previous studies have largely adopted strategies from the vision and language domains (He et al.,
101 2022; Devlin et al., 2018), where atom attributes are randomly masked with a predetermined ratio.
102 Since molecules are atoms held together by strict chemical rules, the data modality of molecular
103 graphs is essentially different from natural images and languages. For molecular graphs, random
104 attribute masking results in either over-reliance on intra-motif neighbors (Dwivedi et al., 2023)
105 or breaking the inter-motif connections via random edge masking. In this work, we introduce a
106 novel strategy of attribute masking, which turns out to capture and transfer useful knowledge from
107 intra-motif structures and long-range inter-motif node features.

108 3 PRELIMINARIES

109 **Graph property prediction** Given a graph $G = (\mathcal{V}, \mathcal{E}) \in \mathcal{G}$ with the node set \mathcal{V} for atoms and
110 the edge set $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ for bonds, we have a d -dimensional node attribute matrix $\mathbf{X} \in \mathbb{R}^{|\mathcal{V}| \times d}$ that
111 represents atom features such as atom type and chirality. We use $y \in \mathcal{Y}$ as the graph-level property
112 label for G , where \mathcal{Y} represents the label space. For graph property prediction, a predictor with
113 the encoder-decoder architecture is trained to encode G into a representation vector in the latent
114 space and decode the representation to predict \hat{y} . The training process optimizes the parameters to
115 make \hat{y} to be the same as the true label value y . A GNN is a commonly used encoder that generates
116 k -dimensional node representation vectors, denoted as $\mathbf{h}_v \in \mathbb{R}^k$, for any node $v \in \mathcal{V}$:

$$\mathbf{H} = \{\mathbf{h}_v : v \in \mathcal{V}\} = \text{GNN}(G) \in \mathbb{R}^{|\mathcal{V}| \times k}. \quad (1)$$

117 Here \mathbf{H} is the node representation matrix for the graph G . Without loss of generality, we implement
118 Graph Isomorphism Networks (GIN) (Xu et al., 2019) as the choice of GNN in accordance with
119 previous work (Hu et al., 2020a). Once the set of node representations are created, a READOUT(\cdot)
120 function (such as max, mean, or sum) is used to summarize the node-level representation into
121 graph-level representation \mathbf{h}_G for any G :

$$\mathbf{h}_G = \text{READOUT}(\mathbf{H}) \in \mathbb{R}^k. \quad (2)$$

122 The graph-level representation vector \mathbf{h}_G is subsequently passed through a multi-layer perceptron
123 (MLP) to generate the label prediction \hat{y} , which exists in the label space \mathcal{Y} :

$$\hat{y} = \text{MLP}(\mathbf{h}_G) \in \mathcal{Y}. \quad (3)$$

124 **GNN pre-training** Random initialization of the predictor’s parameters would easily result in
125 suboptimal solutions for graph property prediction. This is because the number of labeled graphs
126 is usually small. It prevents a proper coverage of task-specific graph and label spaces (Hu et al.,
127 2020a; Liu et al., 2023). To improve generalization, GNN pre-training is often used to warm-up the
128 model parameters based on a much larger set of molecules without labels. In this work, we focus on
129 the attribute masking strategy for GNN pre-training that aims to predict the masked values of node
130 attributes given the unlabeled graphs.

131 4 INTER-MOTIF INFLUENCE

132 To measure the influence generally from (either intra-motif or inter-motif) source nodes on a target
 133 node v , we design a measure that quantifies the influence from any source node u in the same
 134 graph G , denoted by $s(u, v)$. \mathbf{h}_v was learned by Eq. (1) and was influenced by node u . When the
 135 embedding of u is eliminated from GNN initialization, i.e., set $\mathbf{h}_u^{(0)} = \vec{0}$, Eq. (1) would produce a
 136 new representation vector of node v , denoted by $\mathbf{h}_{v, w/o u}$. We use the L^2 -norm to define the influence:

$$s(u, v) = \|\mathbf{h}_v - \mathbf{h}_{v, w/o u}\|_2. \quad (4)$$

137 The collective influence from a group of nodes in a motif $M = (\mathcal{V}_M, \mathcal{E}_M)$ is measured as follows:

$$s_{\text{motif}}(v, M) = \frac{1}{|\mathcal{V}_M \setminus \{v\}|} \sum_{u \in \mathcal{V}_M \setminus \{v\}} s(u, v). \quad (5)$$

138 Suppose the target node v is in the motif $M_v = (\mathcal{V}_{M_v}, \mathcal{E}_{M_v})$. Using M_v as the target motif, the
 139 influence from intra-motif and inter-motif nodes can be calculated as:

$$s_{\text{intra}}(v) = s_{\text{motif}}(v, M_v); \quad s_{\text{inter}}(v) = \frac{\sum_{M \in \mathcal{M} \setminus \{M_v\}} |\mathcal{V}_M| \times s_{\text{motif}}(v, M)}{|\mathcal{V} \setminus \mathcal{V}_{M_v}|}. \quad (6)$$

140 Usually the number of inter-motif nodes is significantly bigger than the number of intra-motif nodes,
 141 i.e., $|\mathcal{V}| \gg |\mathcal{V}_{M_v}|$, which reveals two issues in the influence measurements. First, when the target
 142 motif is too small (e.g., has only one or two nodes), the intra-motif influence cannot be defined or
 143 is defined on the interaction with only one neighbor node. Second, most inter-motif nodes are not
 144 expected to have any influence, so the average function in Eq. (5) would lead comparisons to be
 145 biased to intra-motif influence. To address the two issues, we constrain the influence summation
 146 to be on the *same number* of nodes (i.e., top- k) from the intra-motif and inter-motif node groups.
 147 Explicitly, this means $u \in \mathcal{V}_M / \{v\}$ in Eq. (5) is sampled from the top- k most influential nodes
 148 (top-3). The ratio of inter- to intra-motif influence over the graph dataset \mathcal{G} is then defined as:

$$\text{InfRatio}_{\text{node}} = \frac{1}{\sum_{(\mathcal{V}, \mathcal{E}) \in \mathcal{G}} |\mathcal{V}|} \sum_{(\mathcal{V}, \mathcal{E}) \in \mathcal{G}} \sum_{v \in \mathcal{V}} \frac{s_{\text{inter}}(v)}{s_{\text{intra}}(v)}, \quad (7)$$

$$\text{InfRatio}_{\text{graph}} = \frac{1}{|\mathcal{G}|} \sum_{G=(\mathcal{V}, \mathcal{E}) \in \mathcal{G}} \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \frac{s_{\text{inter}}(v)}{s_{\text{intra}}(v)}, \quad (8)$$

149 where the average function is performed at the node level and graph level, respectively. Eq. (7)
 150 directly measures the influence ratios of all nodes v within the dataset \mathcal{G} . However, this measure may
 151 include bias due to the distribution of nodes within each graph. We alleviate this bias in Eq. (8) by
 152 averaging influence ratios across each graph first.

153 While the InfRatio measurements are able to compare general inter- and intra-motif influences, these
 154 measures combine all inter-motif nodes into one set and do not consider the number of motifs in each
 155 graph. We further define rank-based measures that consider the distribution of motif counts across \mathcal{G} .

156 Let $\{M_1, \dots, M_i, \dots, M_n\}$ be an ordered set, where $M_i \in \mathcal{M}$ and $s_{\text{motif}}(v, M_i) \geq s_{\text{motif}}(v, M_j)$ if
 157 $i < j$. If $M_i = M_v$, we define $\text{rank}_v = i$. Note that graphs with only one motif are excluded as the
 158 distinction between inter and intra-motif nodes loses meaning. From this ranking, we define our score
 159 for inter-motif node influence averaged at the node, motif, and graph levels, derived from a similar
 160 score measurement used in information retrieval, Mean Reciprocal Rank (MRR) (Craswell, 2009):

$$\text{MRR}_{\text{node}} = \frac{1}{\sum_{(\mathcal{V}, \mathcal{E}) \in \mathcal{G}} |\mathcal{V}|} \sum_{(\mathcal{V}, \mathcal{E}) \in \mathcal{G}} \sum_{v \in \mathcal{V}} \frac{1}{\text{rank}_v}, \quad (9)$$

$$\text{MRR}_{\text{graph}} = \frac{1}{|\mathcal{G}|} \sum_{(\mathcal{V}, \mathcal{E}) \in \mathcal{G}} \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \frac{1}{\text{rank}_v} \quad (10)$$

$$\text{MRR}_{\text{motif}} = \sum_{n=2}^N \frac{|\mathcal{G}^{(n)}|}{|\mathcal{G}| \sum_{(\mathcal{V}, \mathcal{E}) \in \mathcal{G}^{(n)}} |\mathcal{V}|} \sum_{(\mathcal{V}, \mathcal{E}) \in \mathcal{G}^{(n)}} \sum_{v \in \mathcal{V}} \frac{1}{\text{rank}_v}, \quad (11)$$

161 where $\mathcal{G}^{(n)} \subset \mathcal{G}$ is the set of graphs that contain $n \in [2, \dots, N]$ motifs.

162 Similar to the InfRatio measurements, MRR_{node} directly captures the impact of the influence ranks
163 for each node within the full graph set, whereas $\text{MRR}_{\text{graph}}$ alleviates bias on the number of nodes
164 within a graph by averaging across individual graphs first. Because these rank-based measurements
165 are intrinsically dependent on the number of motifs within each graph, we additionally define
166 $\text{MRR}_{\text{motif}}$ which weights the measurement towards popular motif counts within the data distribution.
167 In information retrieval, MRR scores are used to quantify how well a system can return the most
168 relevant item for a given query. Higher MRR scores indicate that relevant items were returned at
169 higher ranks for each query. However, as opposed to traditional MRR measurements, where a higher
170 rank for the most relevant item indicates better performance, lower scores are preferred for our MRR
171 measurements as lower intra-motif influence rank indicate greater inter-motif node influence.

172 In Sec 6, we show the inter-motif node influence measurements of previous pre-trained models.

173 5 PROPOSED SOLUTION

174 In this section, we present our novel solution named MoAMa for effectively pre-training graph neural
175 networks on molecular data. We will give details about the strategy of motif-aware attribute masking
176 and reconstruction. Each molecule G will have some portion of their node masked according to
177 domain knowledge based motifs. We replace the node attributes of all masked nodes with a special
178 mask token. Then, the GNN in Eq. (1) encodes the masked graph to the node representation space,
179 and an MLP reconstructs the atom types for the attribute masked molecule.

180 5.1 KNOWLEDGE-BASED MOTIF EXTRACTION

181 To leverage the expertise from the chemistry domain, we extract motifs for molecules using the
182 BRICS (Breaking of Retrosynthetically Interesting Chemical Substructures) algorithm (Degen et al.,
183 2008). This algorithm leverages chemical domain knowledge by creating 16 rules for decomposition,
184 the rules of which define the bonds that should be cleaved from the molecule in order to create a
185 multi-set of disjoint subgraphs. Two key strengths of the BRICS algorithm over a motif-mining
186 strategy (Geng et al., 2023) is that no training is required and important structural features, such as
187 rings, are inherently preserved.

188 For each graph G , the BRICS algorithm decomposes the full graph into separate motifs. We denote
189 the decomposition result as $\mathcal{M}_G = \{M_1, M_2, \dots, M_n\}$, which is a set of n motifs. Each motif
190 $M_i = (\mathcal{V}_i, \mathcal{E}_i)$, for $i \in \{1, 2, \dots, n\}$, is a disjoint subgraph of G such that $\mathcal{V}_i \subset \mathcal{V}$ and $\mathcal{E}_i \subset \mathcal{E}$. For
191 each motif multi-set \mathcal{M}_G , the union of all motifs $M_i \in \mathcal{M}_G$ should equal G . Formally, this means
192 $\mathcal{V} = \bigcup_i \mathcal{V}_i$ and $\mathcal{E} = (\bigcup_i \mathcal{E}_i) \cup E_x$, where E_x represents all the edges removed between motifs
193 during the BRICS decomposition. Within the ZINC15 dataset (Sterling & Irwin, 2015), used for
194 pre-training, each molecule has an average of 9.8 motifs, each of which have an average of 2.4 atoms.

195 5.2 MOTIF-AWARE ATTRIBUTE MASKING AND RECONSTRUCTION

196 To perform motif-aware attribute masking, m motifs are sampled to form the multi-set $\mathcal{M}'_G \subset \mathcal{M}_G$
197 such that $(\sum_{(\mathcal{V}_i, \mathcal{E}_i) \in \mathcal{M}'_G} |\mathcal{V}_i|) / |\mathcal{V}| = \alpha$, for α is a chosen ratio value. The motifs sampled for \mathcal{M}'_G
198 must adhere to two criteria: (1) each node within the motif must be within a k -hop neighborhood (k
199 equals number of GNN layers) of an inter-motif node, and (2) sampled motifs may not be adjacent.
200 These two criteria guarantee inter-motif knowledge access for each masked node. To adhere to the
201 above criteria and account for variable motif sizes, we allow for some flexibility in the value for α .
202 We choose the bounds $0.15 < \alpha < 0.25$ in accordance to those used in previous works ($\alpha = 0.15$
203 (Hu et al., 2020a) and $\alpha = 0.25$ (Hou et al., 2022)).

204 Given a selected motif $M \in \mathcal{M}'_G$, nodes within M have their attributes masked by replacing them
205 with a mask token [MASK], which is a vector $\mathbf{m} \in \mathbb{R}^d$. Each element in \mathbf{m} is a special value that is not
206 present within the attribute space for that particular dimension. For example, we may set the attribute
207 for the atom type dimension in \mathbf{m} to the value 119, as we totally have 118 atom types (Hu et al.,
208 2020a). We use $\mathcal{V}_{[\text{MASK}]} = \{v \in \mathcal{V}_i : M_i = (\mathcal{V}_i, \mathcal{E}_i) \in \mathcal{M}'_G\}$ to denote the set of all the masked
209 nodes. We then define the input node features in the masked attribute matrix $\mathbf{X}_{[\text{MASK}]} \in \mathbb{R}^{|\mathcal{V}| \times d}$ for
210 any $v \in \mathcal{V}$ using the following equation:

$$(\mathbf{X}_{[\text{MASK}]})_v = \begin{cases} \mathbf{X}_v, & v \notin \mathcal{V}_{[\text{MASK}]}, \\ \mathbf{m}, & v \in \mathcal{V}_{[\text{MASK}]}, \end{cases} \quad (12)$$

211 where $(\mathbf{X}_{[\text{MASK}]})_v$ and \mathbf{X}_v denote the row of the node v in $\mathbf{X}_{[\text{MASK}]}$ and \mathbf{X} , respectively. With a
212 GNN encoder, all nodes with attributes $\mathbf{X}_{[\text{MASK}]}$ for the masked graph $G_{[\text{MASK}]}$ are encoded to the
213 latent representation space according to Eq. (1): $\mathbf{H} = \text{GNN}(G_{[\text{MASK}]})$. \mathbf{H} is then used to define the
214 reconstruction loss of the node attributes:

$$\mathcal{L}_{\text{rec}} = \mathbb{E}_{v \in \mathcal{V}_{[\text{MASK}]}}[\log p(\mathbf{X}|\mathbf{H})], \quad (13)$$

215 where $p(\mathbf{X}|\mathbf{H})$ for the reconstruction attribute value is inferred by a decoder. In practice, recon-
216 struction loss is measured using the scaled cosine error (SCE) (Hou et al., 2022), which calculates
217 the difference between the probability distribution for the reconstruction attributes and the one-hot
218 encoded target label vector. This choice of reconstruction loss is further discussed in later sections.

219 5.3 DESIGN SPACE OF THE ATTRIBUTE MASKING STRATEGY

220 The design space of the motif-aware node attribute masking includes the following four parts:

221 **Masking distribution** We investigate the influence of masking distribution to the masking strategy
222 using two factors to control the distribution of masked attributes:

- 223 • Percentage of nodes within a motif selected for masking: we propose to mask nodes from the
224 selected motifs at different percentages. The percentage indicates the strength of the masked
225 domain knowledge, which affects the hardness of the pre-training task of the attribute
226 reconstruction.
- 227 • Dimension of the attributes: We propose to conduct either node-wise or element-wise
228 (dimension-wise) masking. Element-wise masking selects different nodes for masking in
229 different dimensions according to the percentage, while node-wise masking selects different
230 nodes for all-dimensional attribute masking in different motifs.

231 **Reconstruction target** Existing molecular graph pre-training methods heavily rely on two atom
232 attributes: atom type and chirality. Therefore, the reconstructive task could include one or both
233 attributes using one or two different decoders. Experiments will find the most effective task definition.

234 **Reconstruction loss** We study different implementations of reconstruction loss functions for \mathcal{L}_{rec} .
235 They include cross entropy (CE), scaled cosine error (SCE) (Hou et al., 2022), and mean square error
236 (MSE). GraphMAE (Hou et al., 2022) suggested that SCE was the best loss function, however, it is
237 worth investigating the effect of the loss function choices in the motif-based study.

238 Additionally, attribute masking focuses on local graph structures and suffers from representation
239 collapse (Hu et al., 2020a; Hou et al., 2022). To address this issue, we use a knowledge-enhanced
240 auxiliary loss \mathcal{L}_{aux} to complement \mathcal{L}_{rec} . Given any two graphs G_i and G_j from the graph-based
241 chemical space \mathcal{G} , \mathcal{L}_{aux} first calculates the Tanimoto similarity (Bajusz et al., 2015) between G_i and
242 G_j as $\text{Tanimoto}(G_i, G_j)$ based on the bit-wise fingerprints, which characterizes frequent fragments
243 in the molecular graphs. Then \mathcal{L}_{aux} aligns the latent representations with the Tanimoto similarity
244 using the cosine similarity, inspired by previous work (Atsango et al., 2022). Formally, we define:

$$\mathcal{L}_{\text{aux}} = \sum_{i,j} (\text{Tanimoto}(G_i, G_j) - \text{cosine}(\mathbf{h}_{G_i}, \mathbf{h}_{G_j})), 1 \leq i, j \leq |\mathcal{G}|, i \neq j, \quad (14)$$

245 where \mathbf{h}_{G_i} and \mathbf{h}_{G_j} are the graph representation of G_i and G_j , respectively. The full pre-training loss
246 is $\mathcal{L} = \beta\mathcal{L}_{\text{rec}} + (1 - \beta)\mathcal{L}_{\text{aux}}$, where β is a hyperparameter to balance these two loss terms ($\beta = 0.5$).

247 **Decoder model** The decoder trained via Eq. (13) could be a GNN or a MLP. Although the GNN
248 decoder might be powerful (Hou et al., 2022), we are curious if the MLP delivers a comparable or
249 better performance with higher efficiency.

250 6 EXPERIMENTS

251 6.1 EXPERIMENTAL SETTINGS

252 **Datasets** Following the setting of previous studies (Hou et al., 2022; Kim et al., 2022; Xia et al.,
253 2023), 2 million unlabeled molecules from the ZINC15 dataset (Sterling & Irwin, 2015) was used
254 to pre-train the GNN models. To evaluate the performance on downstream tasks, experiments were
255 conducted across eight binary classification benchmark datasets from MoleculeNet (Wu et al., 2017).

Table 1: Test AUC (%) performance on eight molecular datasets comparing our method with baselines. The best AUC-ROC values for each dataset are in **bold**. All models use same GNN architecture except those indicated by *.

	MUV	ClinTox	SIDER	HIV	Tox21	BACE	ToxCast	BBBP	Avg
No Pretrain	70.7±1.8	58.4±6.4	58.2±1.7	75.5±0.8	74.6±0.4	72.4±3.8	61.7±0.5	65.7±3.3	67.2
MCM* Wang et al. (2022)	74.4±0.6	64.7±0.5	62.3±0.9	72.7±0.3	74.4±0.1	79.5±1.3	61.0±0.4	71.6±0.6	69.7
MGSSL Zhang et al. (2021)	77.6±0.4	77.1±4.5	61.6±1.0	75.8±0.4	75.2±0.6	78.8±0.9	63.3±0.5	68.8±0.9	72.3
Grover* Rong et al. (2020)	50.6±0.4	75.4±8.6	57.1±1.6	67.1±0.3	76.3±0.6	79.5±1.1	63.4±0.6	68.0±1.5	67.2
AttrMask Hu et al. (2020a)	75.8±1.0	73.5±4.3	60.5±0.9	75.3±1.5	75.1±0.9	77.8±1.8	63.3±0.6	65.2±1.4	70.8
ContextPred Hu et al. (2020a)	72.5±1.5	74.0±3.4	59.7±1.8	75.6±1.0	73.6±0.3	78.8±1.2	62.6±0.6	70.6±1.5	70.9
GraphMAE Hou et al. (2022)	76.3±2.4	82.3±1.2	60.3±1.1	77.2±1.0	75.5±0.6	83.1±0.9	64.1±0.3	72.0±0.6	73.9
Mole-BERT Xia et al. (2023)	78.6±1.8	78.9±3.0	62.8±1.1	78.2±0.8	76.8 ±0.5	80.8±1.4	64.3±0.2	71.9±1.6	74.0
JOAO You et al. (2021)	76.9±0.7	66.6±3.1	60.4±1.5	76.9±0.7	74.8±0.6	73.2±1.6	62.8±0.7	66.4±1.0	71.1
GraphLoG Xu et al. (2021)	76.0±1.1	76.7±3.3	61.2±1.1	77.8±0.8	75.7±0.5	83.5±1.2	63.5±0.7	72.5±0.8	73.4
D-SLA Kim et al. (2022)	76.6±0.9	80.2±1.5	60.2±1.1	78.6±0.4	76.8 ±0.5	83.8±1.0	64.2±0.5	72.6±0.8	73.9
MoAMa w/o \mathcal{L}_{aux}	78.5±0.4	84.2±0.8	61.2±0.2	79.5 ±0.5	76.2±0.3	84.1 ±0.2	64.6 ±0.1	71.8±0.7	75.0
MoAMa	80.0 ±0.8	85.3 ±2.2	64.6 ±0.5	79.3±0.6	76.5±0.1	80.1±0.5	63.0±0.4	72.8 ±0.9	75.3

256 **Validation methods and evaluation metrics** In accordance with previous work, we adopt a scaffold
 257 splitting approach (Hu et al., 2020a; Zhang et al., 2021). Random splitting may not reflect the actual
 258 use case, so molecules are divided according to structures into train, validation, and test sets (Wu
 259 et al., 2017), using a 80:10:10 split for the three sets. We use the area under the ROC curve (AUC) to
 260 evaluate the test performance of the best validation step during 10 independent runs.

261 **Model configurations** For fair comparison with previous work, a five-layer Graph Isomorphism
 262 Network (GIN) with an embedding dimension of 300 was chosen for the GNN encoder. The
 263 READOUT strategy is mean pooling. During pre-training and fine-tuning, models were trained for
 264 less than 100 epochs using the Adam optimizer and a learning rate of 0.001. The batch sizes for
 265 pre-training and fine-tuning are 256 and 32, respectively.

266 6.2 BASELINES

267 There are two general types of baseline graph pre-training strategies that we evaluate our work
 268 against: **contrastive learning** tasks, such as D-SLA (Kim et al., 2022), GraphLoG (Xu et al., 2021),
 269 and JOAO (You et al., 2021), and **attribute reconstruction**, including Grover (Rong et al., 2020),
 270 AttrMask (Hu et al., 2020a), ContextPred (Hu et al., 2020a), GraphMAE (Hou et al., 2022), and
 271 Mole-BERT (Xia et al., 2023). Additionally, we evaluate on **motif-based pre-training** strategies,
 272 MGSSL (Zhang et al., 2021), which recurrently generates the motif tree for any molecule, and MCM
 273 (Wang et al., 2022), which uses a motif-based convolution module to generate embeddings.

274 6.3 RESULTS

275 We report AUC-ROC of different graph pre-training methods in Table 1. MoAMa outperforms all
 276 baseline methods on five out of eight datasets. On average, MoAMa outperforms the best baseline
 277 method Mole-BERT (Xia et al., 2023) by 1.3% and the best contrastive learning methods D-SLA (Kim
 278 et al., 2022) by 1.4%. Even without the auxiliary loss \mathcal{L}_{aux} , our motif-aware masking strategy still
 279 maintains a performance improvement of 1.0%, which is still competitive with previous methods.

280 6.4 ABLATION STUDIES

281 To verify motif-aware masking parameters, we conduct ablation studies on the selection of masking
 282 distributions, reconstruction target attribute(s), reconstruction loss function, and decoder model.

283 **Study on Masking Distributions** For motif-aware masking, there is the choice of masking the
 284 features of all nodes within the motif or choosing to only mask the features of a percentage of nodes
 285 within each sampled motif. For our study, we choose a motif coverage parameter to decide what
 286 percentage of nodes within each motif to mask, ranging from 25%, 50%, 75%, or 100%.

287 Furthermore, the masking strategy utilized by previous work performs node-wise masking (Hu et al.,
 288 2020a; Hou et al., 2022), where all features of a node are masked. An alternative strategy may be
 289 element-wise masking, where masked elements are chosen over all feature dimensions and implies
 290 that not all features of a node may necessarily be masked. Note that 100% masking will behave the
 291 exact same as node-wise masking, as 100% of nodes within a motif will have each feature masked.

Table 2: Strategy design for motif-aware attribute masking: (1) masking distribution, (2) reconstruction target, (3) reconstruction loss, and (4) decoder model. The chosen design is highlighted.

Design Space	MUV	ClinTox	SIDER	HIV	Tox21	BACE	ToxCast	BBBP	Avg
100% Motif Coverage	80.0±0.8	85.3±2.2	64.6±0.5	79.3±0.6	76.5±0.1	80.1±0.5	63.0±0.4	72.8±0.9	75.3
75% Node-wise	74.9±1.1	82.3±0.4	60.1±0.3	78.8±0.9	76.1±0.1	82.3±0.4	63.4±0.1	72.1±1.0	73.7
75% Element-wise	74.8±0.7	84.9±1.0	58.7±0.1	79.7±0.7	75.6±0.1	85.7±0.4	63.4±0.2	72.6±0.4	74.4
(1) 50% Node-wise	76.6±1.2	86.4±0.6	58.3±0.1	78.1±0.3	75.1±0.2	81.9±0.3	64.6±0.1	72.7±0.1	74.2
50% Element-wise	73.9±0.2	71.2±4.0	61.2±0.4	77.5±0.8	74.9±0.4	81.1±0.7	62.5±0.1	70.6±1.8	71.6
25% Node-wise	76.6±1.5	86.3±0.7	62.4±0.2	78.4±0.2	75.9±0.2	81.8±0.1	65.1±0.1	74.7±0.2	75.1
25% Element-wise	75.2±1.5	82.1±0.4	58.3±0.1	77.8±1.5	75.5±0.2	81.5±0.2	63.1±0.1	71.6±0.3	73.1
Atom Type	80.0±0.8	85.3±2.2	64.6±0.5	79.3±0.6	76.5±0.1	80.1±0.5	63.0±0.4	72.8±0.9	75.3
(2) Chirality	76.3±1.8	75.1±0.9	59.8±0.5	77.9±0.1	76.6±0.1	79.8±0.5	63.8±0.2	73.8±0.7	72.9
Both w/ one decoder	76.2±1.4	74.4±1.1	62.4±0.9	78.2±1.1	75.5±0.6	82.1±0.4	64.3±0.2	72.9±0.2	73.3
Both w/ two decoders	75.9±0.9	81.5±0.1	60.5±0.1	78.5±0.9	75.8±0.2	82.0±1.0	63.7±0.3	73.4±0.3	73.9
Scaled Cosine Error	80.0±0.8	85.3±2.2	64.6±0.5	79.3±0.6	76.5±0.1	80.1±0.5	63.0±0.4	72.8±0.9	75.3
(3) Cross Entropy	78.8±1.1	84.5±0.7	65.4±0.2	78.6±0.4	76.3±0.1	82.4±0.2	62.9±0.5	72.3±0.2	75.1
Mean Squared Error	80.0±0.5	84.1±1.4	64.6±0.5	78.3±0.4	76.8±0.2	80.5±0.6	62.8±0.3	71.8±0.6	74.9
GNN decoder	80.0±0.8	85.3±2.2	64.6±0.5	79.3±0.6	76.5±0.1	80.1±0.5	63.0±0.4	72.8±0.9	75.3
(4) MLP decoder	78.8±0.5	85.2±0.1	65.5±0.3	78.1±0.6	76.2±0.2	82.1±0.6	62.8±0.8	71.7±0.4	75.1

292 We provide the predictive performance within Table 2. The predictive performance for the node-wise
 293 masking outperforms the element-wise masking for both 25% and 50% node coverage. At 75%
 294 coverage, element-wise masking outperforms node-wise. However, the full coverage masking strategy
 295 outperforms all other masking strategies, due to the hardness of the pre-training task, which enables
 296 greater transfer of inter-motif knowledge.

297 **Study on Reconstruction Targets** The choice of attributes to reconstruct for GNNs towards
 298 molecular property prediction has traditionally been atom type (Hu et al., 2020a; Hou et al., 2022).
 299 However, there are other choices for reconstruction that could be explored. We verify the choice
 300 of reconstruction attributes by comparing the performance of the baseline model against models
 301 trained by reconstructing only chirality, both atom type and chirality using two separate decoders,
 302 or both properties using one unified decoder. From Table 2, we note that predicting solely atom
 303 type yields the best pre-training results. The second best strategy was to predict both atom type and
 304 chirality using two decoders. In this case, the loss of the two decoders are independent, leading to the
 305 conclusion that the chirality prediction task is ill-suited to be the pre-training task. Because choice of
 306 chirality is limited to four extremely imbalanced outputs, the useful transferable knowledge may be
 307 significantly lesser than that of atom prediction, which, for the ZINC15 dataset, has nine types.

308 **Study on Reconstruction Loss Functions** For the pretraining task, we have three choices of error
 309 functions to calculate training loss. A standard error function used for masked autoencoders within
 310 computer vision (He et al., 2022; Zhang et al., 2022; Germain et al., 2015) is the cross-entropy loss,
 311 whereas previous GNN solutions utilize mean squared error (MSE) (Hu et al., 2020b; Park et al.,
 312 2019; Salehi & Davulcu, 2019; Wang et al., 2017). GraphMAE (Hou et al., 2022) proposed that
 313 cosine error could mitigate sensitivity and selectivity issues:

$$\mathcal{L}_{\text{rec}} = \frac{1}{|\mathcal{V}_{[\text{MASK}]}|} \sum_{v \in \mathcal{V}_{[\text{MASK}]}} \left(1 - \frac{\mathbf{X}_v^T \mathbf{H}_v}{\|\mathbf{X}_v\| \cdot \|\mathbf{H}_v\|}\right)^\gamma, \gamma \geq 1. \quad (15)$$

314 This equation is called the scaled cosine error (SCE). \mathbf{H} are the reconstructed features, \mathbf{X} are the
 315 ground-truth node features, and γ is a scaling factor ($\gamma = 1$) We investigate the effect these different
 316 error functions have on downstream predictive performance in Table 2 and find that SCE outperforms
 317 CE and MSE, in accordance with previous work.

318 **Study on Decoder Model Choices** We follow the GNN decoder settings from previous work (Hou
 319 et al., 2022) to conduct our study to determine which decoder leads to better downstream predictive
 320 performance. In Table 2, we show that our method outperforms the MLP-decoder strategy, which
 321 support previous work that show MLP-based decoders lead to reduced model expressiveness because
 322 of the inability of MLPs to utilize the high number of embedded features (Hou et al., 2022).

323 6.5 INTER-MOTIF INFLUENCE ANALYSIS

324 In Table 3, we report the two InfRatio and three MRR measurements for our model and several
 325 baselines. A higher influence ratio indicates that inter-motif nodes have a greater effect on the target

Table 3: Measurements of inter-motif knowledge transfer using pre-trained models. A higher ratio is preferred for the InfRatio measurements, and a lower score is preferred for the MRR measurements.

Model	Avg Test AUC	InfRatio _{node} ↑	InfRatio _{graph} ↑	MRR _{node} ↓	MRR _{graph} ↓	MRR _{motif} ↓
AttrMask	70.8	0.70	0.44	0.66	0.64	0.51
MGSSL	72.3	0.60	0.38	0.77	0.75	0.64
GraphLoG	73.4	0.79	0.50	0.61	0.59	0.48
D-SLA	73.8	0.76	0.49	0.67	0.66	0.44
GraphMAE	73.9	0.76	0.48	0.64	0.61	0.49
Mole-BERT	74.0	0.66	0.42	0.72	0.70	0.59
MoAMa	75.3	0.80	0.51	0.59	0.55	0.41

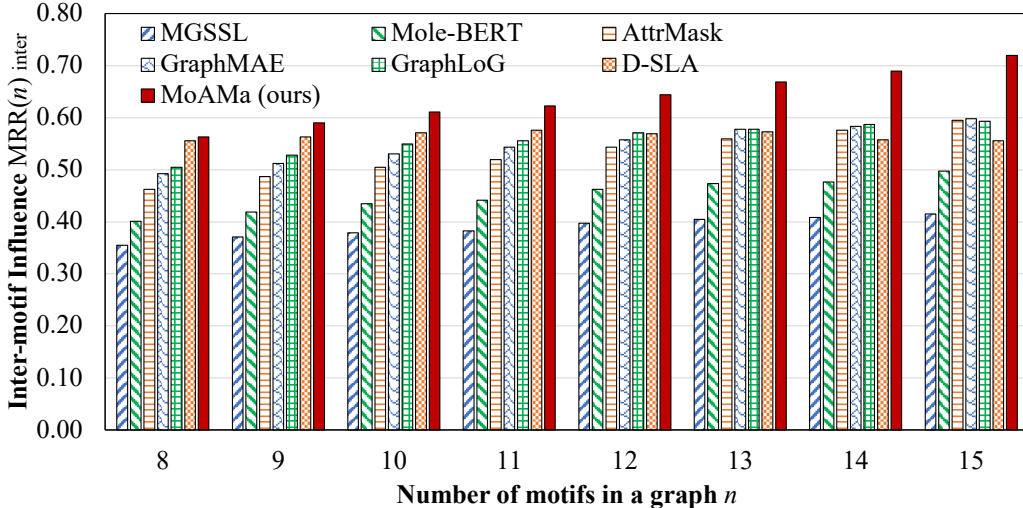


Figure 2: Inter-motif knowledge transfer score by motif count. A higher $MRR_{inter}^{(n)}$ score denotes greater inter-motif knowledge transfer.

node. The relatively low values indicate that the intra-motif node influence is still highly important for the pre-training task, but our method demonstrates the highest inter-motif knowledge transfer amongst the baselines. We see that there is a small positive correlation between the average test AUC for each model and the InfRatio measurements, which supports our claim that greater inter-motif knowledge transfer leads to higher predictive performance. For the MRR measurements, our method boasts the lowest scores, which indicates less intra-motif knowledge dependence and greater inter-motif knowledge transfer.

For the sake of clear visualization, we define an inter-motif score which indicates inter-motif knowledge transfer according to the number of motifs n within a graph:

$$MRR_{inter}^{(n)} = 1 - \frac{1}{\sum_{(\mathcal{V}, \mathcal{E}) \in \mathcal{G}^{(n)}} |\mathcal{V}|} \sum_{(\mathcal{V}, \mathcal{E}) \in \mathcal{G}^{(n)}} \sum_{v \in \mathcal{V}} \frac{1}{\text{rank}_v}. \quad (16)$$

Figure 2 shows that our method outperforms all other models in terms of inter-motif knowledge transfer as shown by the higher $MRR_{inter}^{(n)}$ scores across different motif counts. Additionally, the inter-motif knowledge transfer using our method becomes more pronounced on graphs with higher numbers of motifs.

7 CONCLUSIONS

In this work, we introduced a novel motif-aware attribute masking strategy for attribute reconstruction during graph model pre-training. This motif-aware masking strategy outperformed existing methods that used random attribute masking, and achieved competitive results with the state-of-the-art methods because of the explicit transfer of long-range inter-motif knowledge and intra-motif structural information. We quantitatively verify the increase in inter-motif knowledge transfer of our strategy over previous works using inter-motif node influence measurements.

346 REFERENCES

- 347 Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications,
348 2021.
- 349 Austin Atsango, Nathaniel L. Diamant, Ziqing Lu, Tommaso Biancalani, Gabriele Scalia, and
350 Kangway V. Chuang. A 3d-shape similarity-based contrastive approach to molecular representation
351 learning, 2022.
- 352 Dávid Bajusz, Anita Rácz, and Károly Héberger. Why is tanimoto index an appropriate choice for
353 fingerprint-based similarity calculations? *Journal of Cheminformatics*, 7, 2015.
- 354 Rees Chang, Yu-Xiong Wang, and Elif Ertekin. Towards overcoming data scarcity in materials
355 science: unifying models and datasets with a mixture of experts framework, 2022.
- 356 Nick Craswell. *Mean Reciprocal Rank*, pp. 1703–1703. Springer US, Boston, MA, 2009. ISBN
357 978-0-387-39940-9. doi: 10.1007/978-0-387-39940-9_488. URL [https://doi.org/10.](https://doi.org/10.1007/978-0-387-39940-9_488)
358 [1007/978-0-387-39940-9_488](https://doi.org/10.1007/978-0-387-39940-9_488).
- 359 Jörg Degen, Christof Wegscheid-Gerlach, Andrea Zaliani, and Matthias Rarey. On the art of compiling
360 and using 'drug-like' chemical fragment spaces. *ChemMedChem*, 3, 2008.
- 361 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep
362 bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- 363 Vijay Prakash Dwivedi, Ladislav Rampásek, Mikhail Galkin, Ali Parviz, Guy Wolf, Anh Tuan Luu,
364 and Dominique Beaini. Long range graph benchmark, 2023.
- 365 Jean MJ Frechet. Functional polymers and dendrimers: reactivity, molecular architecture, and
366 interfacial energy. *Science*, 263(5154):1710–1715, 1994.
- 367 Zijie Geng, Shufang Xie, Yingce Xia, Lijun Wu, Tao Qin, Jie Wang, Yongdong Zhang, Feng Wu, and
368 Tie-Yan Liu. De novo molecular generation via connection-aware motif mining. *arXiv preprint*
369 *arXiv:2302.01129*, 2023.
- 370 Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. Made: Masked autoencoder for
371 distribution estimation, 2015.
- 372 Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural
373 message passing for quantum chemistry, 2017.
- 374 Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked
375 autoencoders are scalable vision learners, 2021.
- 376 Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked
377 autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on*
378 *Computer Vision and Pattern Recognition*, pp. 16000–16009, 2022.
- 379 Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, C. Wang, and Jie Tang. Graphmae:
380 Self-supervised masked graph autoencoders. *Proceedings of the 28th ACM SIGKDD Conference*
381 *on Knowledge Discovery and Data Mining*, 2022.
- 382 Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec.
383 Strategies for pre-training graph neural networks, 2020a.
- 384 Ziniu Hu, Yuxiao Dong, Kuansan Wang, Kai-Wei Chang, and Yizhou Sun. Gpt-gnn: Generative
385 pre-training of graph neural networks, 2020b.
- 386 Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. Spanbert:
387 Improving pre-training by representing and predicting spans, 2020.
- 388 Dongki Kim, Jinheon Baek, and Sung Ju Hwang. Graph self-supervised learning with accurate
389 discrepancy learning. *ArXiv*, abs/2202.02989, 2022.
- 390 Thomas N. Kipf and Max Welling. Variational graph auto-encoders, 2016.

-
- 391 Yoav Levine, Barak Lenz, Opher Lieber, Omri Abend, Kevin Leyton-Brown, Moshe Tennenholtz,
392 and Yoav Shoham. Pmi-masking: Principled masking of correlated spans, 2020.
- 393 Gang Li, Heliang Zheng, Daqing Liu, Chaoyue Wang, Bing Su, and Changwen Zheng. Semmae:
394 Semantic-guided masking for learning masked autoencoders, 2022.
- 395 Gang Liu, Eric Inae, Tong Zhao, Jiaxin Xu, Tengfei Luo, and Meng Jiang. Data-centric learning from
396 unlabeled graphs with diffusion model. *arXiv preprint arXiv:2303.10108*, 2023.
- 397 Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, Lina Yao, and Chengqi Zhang. Adversarially
398 regularized graph autoencoder for graph embedding, 2019.
- 399 Jiwoong Park, Minsik Lee, Hyung Jin Chang, Kyuewang Lee, and Jin Young Choi. Symmetric graph
400 convolutional autoencoder for unsupervised graph representation learning. In *Proceedings of the*
401 *IEEE International Conference on Computer Vision*, pp. 6519–6528, 2019.
- 402 Merichel Plaza, Tania Pozzo, Jiayin Liu, Kazi Zubaida Gulshan Ara, Charlotta Turner, and Eva
403 Nordberg Karlsson. Substituent effects on in vitro antioxidizing properties, stability, and solubility
404 in flavonoids. *Journal of agricultural and food chemistry*, 62(15):3321–3333, 2014.
- 405 Phillip Pope, Soheil Kolouri, Mohammad Rostrami, Charles Martin, and Heiko Hoffmann. Discover-
406 ing molecular functional groups using graph convolutional neural networks, 2019.
- 407 Yu Rong, Yatao Bian, Tingyang Xu, Weiyang Xie, Ying Wei, Wenbing Huang, and Junzhou Huang.
408 Self-supervised graph transformer on large-scale molecular data. In *Proceedings of the 34th*
409 *International Conference on Neural Information Processing Systems, NIPS’20*, Red Hook, NY,
410 USA, 2020. Curran Associates Inc. ISBN 9781713829546.
- 411 Amin Salehi and Hasan Davulcu. Graph attention auto-encoders, 2019.
- 412 T. Sterling and John J. Irwin. Zinc 15 – ligand discovery for everyone. *Journal of Chemical*
413 *Information and Modeling*, 55:2324 – 2337, 2015.
- 414 Mengying Sun, Jing Xing, Huijun Wang, Bin Chen, and Jiayu Zhou. Mocl: Data-driven molecular fin-
415 gerprint via knowledge-aware contrastive learning from molecular graph. In *Proceedings of the 27th*
416 *ACM SIGKDD Conference on Knowledge Discovery & Data Mining, KDD ’21*, pp. 3585–3594,
417 New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383325. doi:
418 10.1145/3447548.3467186. URL <https://doi.org/10.1145/3447548.3467186>.
- 419 Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu,
420 Hao Tian, and Hua Wu. Ernie: Enhanced representation through knowledge integration, 2019.
- 421 Chun Wang, Shirui Pan, Guodong Long, Xingquan Zhu, and Jing Jiang. Mgae: Marginalized
422 graph autoencoder for graph clustering. In *Proceedings of the 2017 ACM on Conference on*
423 *Information and Knowledge Management, CIKM ’17*, pp. 889–898, New York, NY, USA, 2017.
424 Association for Computing Machinery. ISBN 9781450349185. doi: 10.1145/3132847.3132967.
425 URL <https://doi.org/10.1145/3132847.3132967>.
- 426 Yifei Wang, Shiyang Chen, Guobin Chen, Ethan Shurberg, Hang Liu, and Pengyu Hong. Motif-based
427 graph representation learning with application to chemical molecules, 2022.
- 428 Zhenqin Wu, Bharath Ramsundar, Evan Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh Pappu,
429 Karl Leswing, and Vijay Pande. Moleculenet: A benchmark for molecular machine learning.
430 *Chemical Science*, 9, 03 2017. doi: 10.1039/C7SC02664A.
- 431 Jun Xia, Yanqiao Zhu, Yuanqi Du, and Stan Z. Li. A survey of pretraining on graphs: Taxonomy,
432 methods, and applications, 2022.
- 433 Jun Xia, Chengshuai Zhao, Bozhen Hu, Zhangyang Gao, Cheng Tan, Yue Liu, Siyuan Li, and Stan Z.
434 Li. Mole-BERT: Rethinking pre-training graph neural networks for molecules. In *The Eleventh*
435 *International Conference on Learning Representations*, 2023. URL [https://openreview.](https://openreview.net/forum?id=jevY-DtiZTR)
436 [net/forum?id=jevY-DtiZTR](https://openreview.net/forum?id=jevY-DtiZTR).

-
- 437 Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu.
438 Simmim: A simple framework for masked image modeling, 2022.
- 439 Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural
440 networks? In *7th International Conference on Learning Representations, ICLR 2019, New Orleans,
441 LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL [https://openreview.net/forum?
442 id=ryGs6iA5Km](https://openreview.net/forum?id=ryGs6iA5Km).
- 443 Minghao Xu, Hang Wang, Bingbing Ni, Hongyu Guo, and Jian Tang. Self-supervised graph-
444 level representation learning with local and global structure. In Marina Meila and Tong Zhang
445 (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of
446 *Proceedings of Machine Learning Research*, pp. 11548–11558. PMLR, 18–24 Jul 2021. URL
447 <https://proceedings.mlr.press/v139/xu21g.html>.
- 448 Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph
449 contrastive learning with augmentations. *Advances in neural information processing systems*, 33:
450 5812–5823, 2020.
- 451 Yuning You, Tianlong Chen, Yang Shen, and Zhangyang Wang. Graph contrastive learning automated.
452 In *International Conference on Machine Learning*, pp. 12121–12132. PMLR, 2021.
- 453 Seongjun Yun, Seoyoon Kim, Junhyun Lee, Jaewoo Kang, and Hyunwoo J Kim. Neo-
454 gnns: Neighborhood overlap-aware graph neural networks for link prediction. In M. Ran-
455 zato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances
456 in Neural Information Processing Systems*, volume 34, pp. 13683–13694. Curran Asso-
457 ciates, Inc., 2021. URL [https://proceedings.neurips.cc/paper_files/paper/
458 2021/file/71ddb91e8fa0541e426a54e538075a5a-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2021/file/71ddb91e8fa0541e426a54e538075a5a-Paper.pdf).
- 459 Chaoning Zhang, Chenshuang Zhang, Junha Song, John Seon Keun Yi, Kang Zhang, and In So
460 Kweon. A survey on masked autoencoder for self-supervised learning in vision and beyond, 2022.
- 461 Zaixin Zhang, Qi Liu, Hao Wang, Chengqiang Lu, and Chee-Kong Lee. Motif-based graph self-
462 supervised learning for molecular property prediction. In *Neural Information Processing Systems*,
463 2021.
- 464 Yanqiao Zhu, Yichen Xu, Qiang Liu, and Shu Wu. An empirical study of graph contrastive learning.
465 *arXiv preprint arXiv:2109.01116*, 2021.

Table 4: Test RSME performance on three molecular datasets comparing our method with baselines. The best RMSE values for each dataset are in **bold**.

	ESOL	FreeSolv	Lipophilicity	Avg
No Pretrain	1.388±0.05	2.965±0.14	0.794±0.005	1.716±0.065
MGSSL Zhang et al. (2021)	1.259±0.01	2.519±0.006	0.722 ±0.002	1.500 ±0.006
AttrMask Hu et al. (2020a)	1.307±0.01	2.699±0.03	0.766±0.002	1.591±0.014
ContextPred Hu et al. (2020a)	1.350±0.08	2.784±0.06	0.777 ±0.008	1.637±0.049
GraphMAE Hou et al. (2022)	1.235±0.05	2.598±0.02	0.755±0.008	1.530±0.026
Mole-BERT Xia et al. (2023)	1.239±0.02	2.504 ±0.03	0.740±0.006	1.494 ±0.019
JOAO You et al. (2021)	1.341±0.02	3.243±0.09	0.774±0.008	1.786±0.039
GraphLoG Xu et al. (2021)	1.341±0.01	2.742±0.01	0.739±0.008	1.607±0.009
D-SLA Kim et al. (2022)	1.289±0.02	2.526±0.01	0.730±0.004	1.515±0.011
MoAMa	1.228 ±0.01	2.552±0.01	0.746±0.001	1.509 ±0.007

466 A APPENDIX

467 A.1 INTER-MOTIF INFLUENCE EVALUATION COMPLEXITY

468 In the worst-case, evaluation of inter-motif node influence can be computed between every pair of
 469 nodes within a molecule, causing an evaluation complexity of $O(n^2)$, for n is the number of nodes in
 470 a graph G . However, GNN message passing is limited by the number of layers used, k . Therefore,
 471 the node influence calculations will only need to be performed on neighbors within a k -hop radius of
 472 each other. This means that the time complexity of our evaluation is $O(n\bar{d}^k)$, where n is the number
 473 of nodes in the graph, k is the number of layers of our GNN ($k = 5$), and \bar{d} is the average degree of a
 474 node. $\bar{d}^k \leq n$ as molecular graphs are sparse, so the evaluation is not nearly as inefficient as $O(n^2)$.

475 A.2 REGRESSION TASKS

476 We conducted additional evaluations on three regression datasets from MoleculeNet, ESOL, FreeSolv,
 477 and Lipophilicity (Wu et al., 2017). We use RMSE to measure the test performance of the best
 478 validation step during 3 independent runs.

479 Our method outperforms all baselines on the ESOL dataset and shows comparative results with
 480 previous methods when considering average RMSE across all three datasets.

481 A.3 CASE STUDY

482 In Figure 3 there are two pairs of molecules, colored in blue and purple, that domain experts suggest
 483 to study. They are pairs because they look similar, have mostly similar properties, and have some
 484 different properties due to the structural differences. The molecular embedding space is obtained by
 485 a two-dimensional t-SNE algorithm on the pre-trained embeddings of the methods. The distributions
 486 and average distance between the graph examples are similar across the methods. However, contrastive
 487 learning strategies such as GraphLoG and JOAO failed to capture the proximities and put the pairs
 488 too distantly from each other. AttrMask based on random masking strategy was not able to learn
 489 from the structural difference at a higher level and put the pairs too close to each other. The proposed
 490 MoAMa provides a more reasonable set of embeddings for downstream fine-tuning.

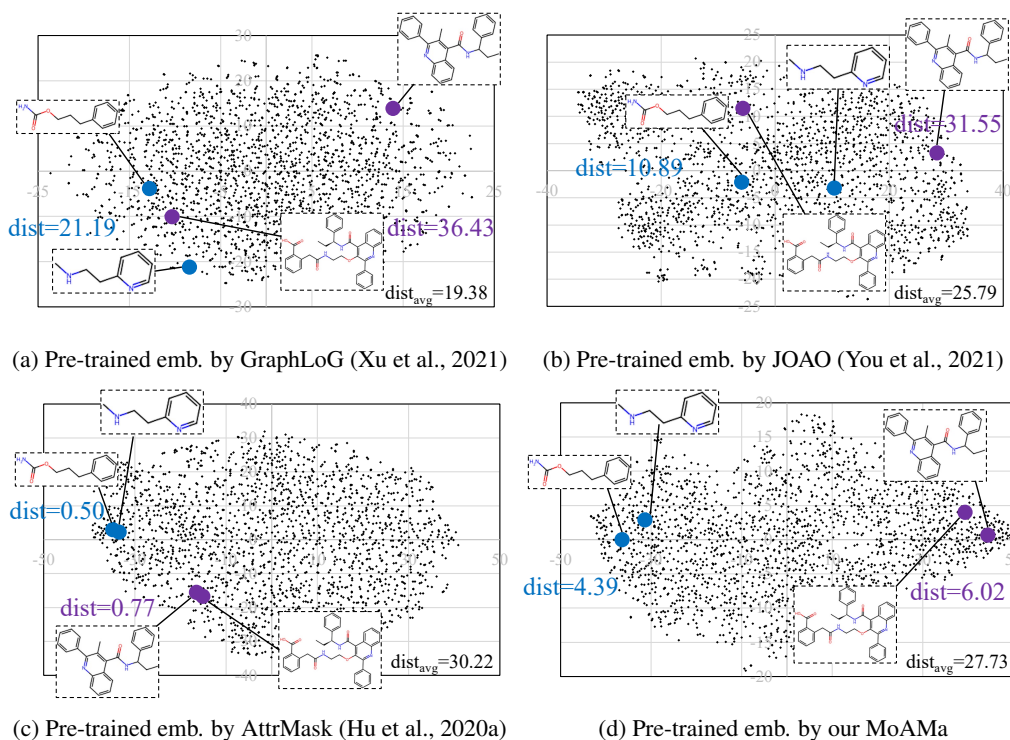


Figure 3: Case study: MoAMA in (d) preserves the structural proximities of molecules better than other methods in (a-c) into pre-trained graph embeddings. The blue pair and purple pair are similar molecules yet have motif-based structural difference.