反思分子屏蔽图建模中的令牌器和解码器

Zhiyuan Liu[†] **Yaorui Shi**[‡] **An Zhang**[†] **Enzhi Zhang**[§] **Kenji Kawaguchi**[†] **Xiang Wang**[‡*] **Tat-Seng Chua**[†]
[†]新加坡国立大学、中国科学技术大学[‡]
[§]北海道大学
{acharkq,shiyaorui,xiangwang1223}@gmail.com, anzhang@u.nus.edu
enzhi.zhang.n6@elms.hokudai.ac.jp, {kenji,chuats}@comp.nus.edu.sg

摘要

掩蔽图建模在分子图的自我监督表征学习中表现出色。仔细研究以往的研究，我们可以发现一个由三个关键部分组成的共同方案：(1) 图标记化器（graph tokenizer），将分子图分解成更小的片段（*即*子图）并转换成标记；(2) 图掩码（graph masking），用掩码破坏图；(3) 图自动编码器（graph autoencoder），首先在掩码图上应用编码器生成表示，然后在表示上使用解码器恢复原始图的标记。然而，以往的美高梅娱乐平台研究广泛关注图屏蔽和编码器，而对标记器和解码器的了解却很有限。为了弥补这一差距，我们首先总结了节点、边、图案和图神经网络（GNN）等粒度的流行分子标记化器，然后研究了它们作为 MGM 重建目标的作用。此外，我们还探索了在 MGM 中采用表达式解码器的可能性。我们的研究结果表明，子图层标记器和具有足够表现力的解码器与重任务解码对编码器的表征学习有很大影响。最后，我们提出了一种新颖的 MGM 方法 **SimSGT**，其特点是**基于**简单 GNN 的标记器（**SGT**）和有效的解码策略。通过经验验证，我们的方法优于现有的分子自监督学习方法。我们的代码和检查点可在 https://github.com/syr-cn/SimSGT 上获得。

1 导言

分子表征学习（MRL）[1, 2, 3]是一个重要的研究领域，它有许多重要的下游应用，如分子性质预测[4]、药物发现[5, 6]和逆合成[7, 8]。鉴于分子可以用图表示，图自监督学习（SSL）自然而然地适合这一问题。在各种图自监督学习技术中，屏蔽图建模（MGM）最近引起了极大的兴趣[9, 10, 11]。

本文通过 MGM 研究 MRL，旨在预训练分子编码器，以便在下游应用中进行微调。在研究了图[12, 9, 10]、语言[13, 14]和计算机视觉[15, 16]中的屏蔽建模方法后，我们总结出 MGM 依赖于三个关键组件--图标记器、图屏蔽和图自动编码器，如图 1 所示：

- **图标记器。**给定一个图 g，图标记器利用图分割函数 [1, 17, 18, 2] 将 g 分割成更小的子图，如节点和图案。然后，这些

*通讯作者：王翔王翔，合肥综合性国家科学中心数据空间研究所人工智能研究所研究员。

第 37 届神经信息处理系统大会（NeurIPS 2023）。

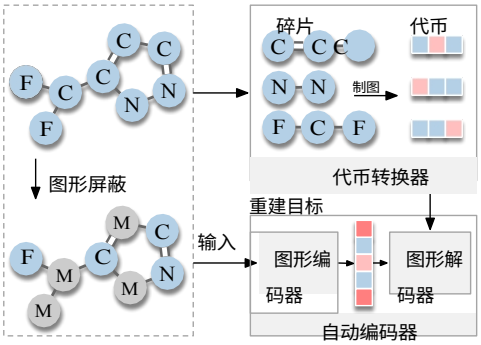


图 1：屏蔽图建模的流程。

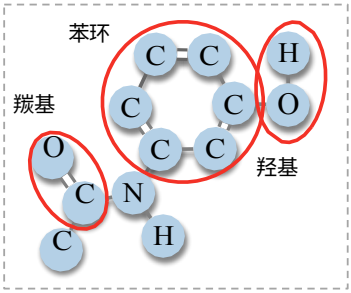


图 2：分子中的子图层模式示例 。 SMILES: CC(=O)Nc1cccc(O)c1.

片段被映射成固定长度的标记，作为以后重建的目标。显然，图标记的粒度决定了屏蔽建模中表征的抽象程度[19, 16]。这一点与分子尤其相关，因为分子的性质主要由子图粒度的模式决定[20]。例如，图 2 所示的分子包含一个苯环子图。苯环赋予分子芳香性，使其比只有单键的饱和化合物更稳定[21]。因此，应用能生成子图级标记的图标记化器可能会提高下游性能。

- **图形屏蔽。**在输入自动编码器之前，g *会*被添加随机噪音破坏，通常是通过随机屏蔽节点或丢弃边[12, 11]。图形掩码对于防止自动编码器仅仅复制输入内容以及引导自动编码器学习共现图形模式之间的关系至关重要。

- **图形自动编码器**图形自动编码器由图形编码器和图形解码器组成 [12, 9]。图编码器生成被破坏图的隐藏表示，图解码器根据这些表示尝试恢复被破坏的信息。编码器和解码器通过最小化解码器输出与重建目标（*即*图标记器诱导的图标记）之间的距离来共同优化。考虑到目标是复杂的子图级标记，有效的重建可能需要一个具有足够表现力的图解码器。

虽然上述三个部分都很重要，但以往的 MGM 研究主要集中在图掩码 [12, 11, 22, 23] 和图编码器 [2, 24, 25, 26]，对标记化器和解码器重视不够。例如，虽然有大量基于图案的碎片函数用于 MRL [1, 17, 18, 2]，但它们作为 MGM 的标记化器却被忽视了。此外，以前的许多研究 [12, 10, 11, 22, 23, 2, 25] 都采用线性或 MLP 解码器进行图重构，而对更具表现力的解码器基本上没有进行研究。

在这项工作中，我们首先从节点、边、图案和图神经网络（GNN）的粒度出发，总结了作为图标记化器的各种分割函数。根据这一总结，我们系统地评估了它们在 MGM 中的经验性能。我们的分析表明，在 MGM 中重建子图层标记比重建节点标记更有效。此外，我们还发现，一个具有足够表现力的解码器结合 remask 解码 [9] 可以提高编码器的再现质量。值得注意的是，remask 将编码器和解码器 "解耦"，将编码器的重点从分子重构转向 MRL，从而提高下游性能。总之，我们发现，将子图级标记器和具有足够表现力的解码器与 remask 解码相结合，可以提高 MGM 性能。

基于上述发现，我们提出了一个新颖的预训练框架--基于简单 GNN 标记器的屏蔽图建模（

SimSGT）。SimSGT 采用了一种基于简单 **GNN 的标记**器（**SGT**），可以去除每个 GNN 层中的非线性更新函数。令人惊讶的是，我们发现单层 SGT 与其他基于 GNN 的预训练标记器和化学启发标记器相比，具有竞争力甚至更好的性能。SimSGT 的编码器采用 GraphTrans [27] 架构，解码器采用较小的 GraphTrans 架构，以便为 MRL 和分子重构任务提供足够的容量。此外，我们还提出了 remask-v2，以解耦 GraphTrans 架构的编码器和解码器。最后，SimSGT 在下游分子性质预测和药物靶点亲和力任务 [28, 29] 中得到了验证，超越了领先的图 SSL 方法（*如*GraphMAE [9] 和 Mole-BERT [10]）。

2 预备

在本节中，我们首先介绍 MGM。然后，我们对现有的图标记化器进行分类。最后，我们将讨论用于 MGM 的图自动编码器的架构。

符号让 G 表示图空间。分子可表示为一个图 $g = (V, E) \in G$，其中 V 是节点集，E 是边集。每个节点 $i \in V$ 都与节点特征 $\boldsymbol{x}_i \in \mathrm{R}^{d_0}$ 相关联，而每条边 $(i, j) \in E$ 都与边特征 $\boldsymbol{e}_{ij} \in \mathrm{R}^{d_1}$ 相关联。图 g 的结构也可以用其邻接矩阵 $\mathbf{A} \in \{0, 1\}^{|V|\times|V|}$ 表示，如果 $(i, j) \in E$，则 $\mathbf{A}_{ij} = 1$，否则 $\mathbf{A}_{ij} = 0$。

2.1 初步：屏蔽图形建模

在此，我们将说明 MGM 的三个关键步骤：图形标记器、图形屏蔽和图形自动编码器。

图标记符。给定一个图 g，我们利用图标记器 $tok(g) = \mathbf{y}_t = m(t)$ $t \in f(g)$ 生成图标记作为重建目标。标记符号生成器 $tok(\,)$ 由一个分片函数 f 和一个映射函数 m 组成，前者可将 g 分解为一组子图 $f(g) = t = (\,, t)$ g，以及将子图转换为固定长度向量的映射函数 $m(t)$ R^d。在这项工作中，我们允许 $f(g)$ 包括重叠子图，以扩大图标记化器的范围。

图屏蔽此外，我们还通过随机节点屏蔽为 g 添加噪音。这里我们不使用边缘删除，因为 Hou 等人 [9] 的经验表明，边缘删除很容易导致下游任务的性能下降。具体来说，节点屏蔽会随机采样一个节点子集 $V_m \subseteq V$，并用一个特殊标记 $\mathbf{m}_0 \in \mathrm{R}^{d_0}$ 替换它们的特征。我们用 $\tilde{\mathbf{x}}_i$ 表示被屏蔽的节点特征：

$$\tilde{\mathbf{x}}_i = \begin{cases} \mathbf{m}_0, & \forall i \in V_m \\ \mathbf{x}_i, & \text{否则} \end{cases}. \tag{1}$$

图自动编码器。然后将损坏的图 $\tilde{g}$ 送入图自动编码器，以重新构建图。有关图自动编码器架构的细节，我们将在第 2.3 节中讨论。假设 $\{\mathbf{z}_i \mid i \in V\}$ 是图自动编码器的节点输出。如果没有特别说明，我们通过对子图 t 的节点表示进行均值池化，得到子图 t 的预测 $\hat{\mathrm{y}}_t = \mathrm{MEAN}(\{\mathbf{z}_i \mid i \in V_t\})$。图自动编码器是通过最小化预测 $\{\hat{\mathrm{y}}_t \mid t \in f(g)\}$ 与目标 $tok(g) = \{\mathbf{y}_t \mid t \in f(g)\}$ 之间的距离来训练的。重建损失是在包含损坏信息的标记上累积的 $\{t \mid t \in f(g), V_t \cap V_m \neq \varnothing\}$：

$$L_0 = \frac{1}{|f(g)|} \sum_{t \in f(g), V_t \cap V_m \neq \varnothing} \ell(\hat{\mathrm{y}}_t, \mathbf{y}_t), \tag{2}$$

其中 $\ell(\cdot, \cdot)$ 是损失函数，取决于 $\mathbf{y}_t$ 的类型。我们使用均方误差 [15] 来计算 $\ell(\cdot, \cdot)$ 当 $\mathbf{y}_t$ 是连续向量时，使用交叉熵[16]；当 $\mathbf{y}_t$ 是离散值时，使用交叉熵[16]。

2.2 重温分子标记符

对当前的 MRL 方法进行仔细研究后，我们将分子标记符归纳为四个不同的类别，如表 1 所示。本文将系统地详细介绍前三类，而基于简单 GNN 的标记化器将在第 3 节中介绍。

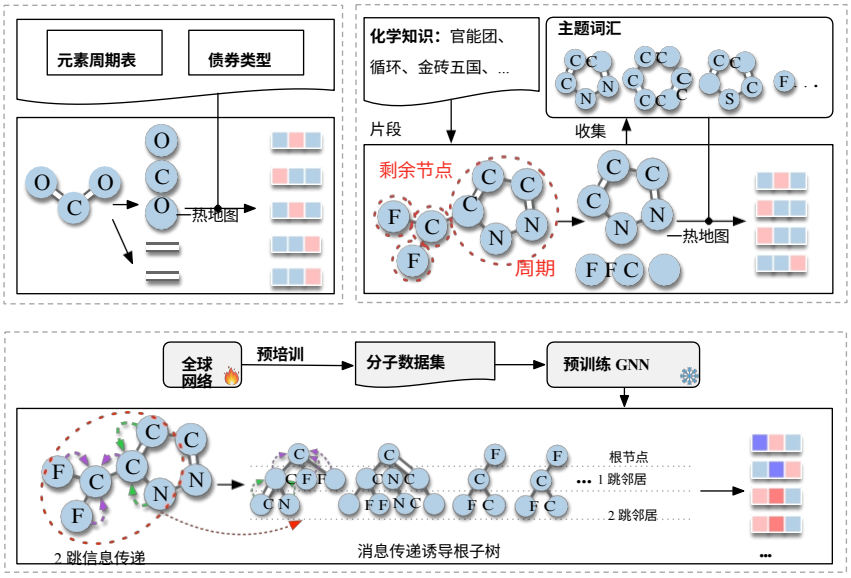
表 1：图形标记符号化器汇总。

标记符子	图类型 标记符	潜在限制
节点、MotifFGs	边节点和边节点和边的特征 低级特征、循环 等。Motif 类型	专家知识 预训练GNN 有
根子树 冻结 GNN 表示法tokenizer额外预训练 简单 GNN -		有根子树冻结 GNN 表示法

节点、边标记符 [12, 11]。图的节点和边可以直接用作图标记：

$$tok_{\text{node}}(g) = \{\mathbf{y}_i = \boldsymbol{x}_i \mid i \in V\}、tok_{\text{edge}}(g) = \{\mathbf{y}_{ij} = \boldsymbol{e}_{ij} \mid (i, j) \in E\}。 \quad (3)$$

图 3a 展示了使用节点的原子序数和边的键类型作为分子中的图标记。这些符号在以往的研究中得到了广泛应用 [12, 11, 25, 24]，主要是因为它们具有以下优点



(a) 节点、边缘标记符　　(b) 基于图案的标记符

(c) 基于 GNN 的预训练标记符

图 3：前三种图标记化器及其诱导子图的示例。(b) 基于图案的标记符，应用循环和剩余节点的碎裂函数。(c) 基于 GIN 的双层标记化器，为图中的每个节点提取 2 跳根子树。

简单。然而，原子序数和键类型属于低级特征。对于需要对图形语义有高层次理解的下游任务来说，重建它们可能不是最佳选择。

基于 Motif 的标记符。母题是图结构中具有统计意义的子图模式。对于分子来说，功能基团（FGs）是由专家根据 FGs 的生化特征人工策划的主题图[30, 31]。例如，含有苯环的分子具有芳香性。考虑到人工策划的 FGs 数量有限，不能完全覆盖所有分子，以往的研究[32, 1, 18]采用了化学启发的片段函数来发现主题。在此，我们总结了常用的破碎函数：

- **FG** [18, 2]。FG 是分子的子图，在不同化合物中表现出一致的化学行为。在化学工具包[30, 31]中，FG 的子结构模式由 SMARTS 语言[33]描述。设 $S_0 = \{s_i\}_{i=1}^{|S_0|}$ 是 FG 的 SMARTS 模式集，设 $p_s(g)$ 是返回 g 中 s 的函数。基于 FG 的分片工作原理如下

$$f_{\text{FG}}(g, S_0) = \bigcup_{s \in S_0} p_s(g), \tag{4}$$

- **循环** [32, 1, 18]。由于分子中的循环具有潜在的化学意义，因此经常被提取为主题。图 3b 描述了将一个五节点循环分割为一个母题的过程。如果两个循环有两个以上的原子重叠，它们可以合并，因为它们构成了一个桥式化合物 [34]。让 c_n 表示由 n 个节点组成的循环。它们可以写成

$$f_{\text{cycle}}(g) = \{t \mid t = c_{|t|}, t \subseteq g\}, \tag{5}$$

$$f_{\text{cycle-merge}}(g) = \{t_i \cup t_j \mid t_i, t_j \in f_{\text{cycle}}(g), i \neq j, |t_i \cap t_j| > 2\}, \tag{6}$$

- **BRICS** [35, 1]。BRICS 在潜在的裂解位点上分裂分子，在特定的环境条件或催化剂作用下，化学键可在这些位点上断裂。BRICS 的关键步骤是识别分子 g 的潜在裂解位点，用

$\psi(g)$ 表示。要做到这一点，需要找到两边都符合 BRICS 中预先定义的 "类 FG "亚结构模式 S_1 的化学键：

$$\psi(g) = \cup\{E \setminus (E_t \cup E_{g-t}) \mid t, g - t \in f_{\mathrm{FG}}(g, S_1)\} \text{ 、} \quad (7)$$

其中，$g-t = g[_t]$ 表示删除 g 在 t 中的节点和相应的附带边[36]；
$E \cup (E_t - E_{gt})$ 包含连接 t 和 gt 的键。接下来，g 被分割成最大的子图，这些子图不包含裂解位点 $\psi(g)$中的任何键：

$$f_{\mathrm{BRICS}}(g) = \{t \mid \psi(g) \cap E_t = \varnothing, f_{\mathrm{BRICS}}(t) \cap 2^t = \{t\}, t \subseteq g\}. \quad (8)$$

需要注意的是，最初的金砖五国包含更多规则，例如对 t 和 g t 组合的限制。为简单起见，我们在此只介绍关键步骤。

- **剩余节点和边** [32, 18]：给定另一个分片函数 f_0，未包含在任何 f_0 的输出中的节点和边将被视为单独的子图。这就提高了 f_0 对原始图的覆盖率。图 3b 显示了在 f_{cycle} 之后对剩余节点进行分片的示例：不在任何循环中的节点被视为独立子图。

为了得到更细粒度的子图，以往的研究 [1, 32, 18] 通常通过联合（如 $f_1(g) \cup f_2(g)$）或组合（如 $f_2(t)\ t\ f_1(g)$）的方式将多个分片函数组合在一起。让 f_{motif} 成为组合后的最终破碎函数。我们通过 f_{motif} 分割数据集中的每个分子，并收集一个主题词词汇表，通过阈值过滤掉不常见的主题词。然后，给定一个新分子 g'，我们可以通过对其主题词 $f_{\text{motif}}(g')$ 进行单次编码来生成其标记：

$$tok_{\text{motif}}(g') = \{\mathbf{y}_t = \text{one-hot}(t, \mathcal{M}) \mid t \in f_{\text{motif}}(g')\} \quad 。 \tag{9}$$

基于预训练 GNN 的标记器 [10]。预训练的 GNN 可以作为图标记符。以 k 层图同构网络（GIN）[37] 为例。它的节点嵌入总结了节点 k 跳根子树的结构信息，使其成为子图层的图标记（图 3c）。GIN 同时执行分片和映射。它可以写成

$$tok_{\text{GIN}}(g) = \{\mathbf{y}_i = \text{SG}(\boldsymbol{h}_i^{(k)}) \mid i \in V\}、 \tag{10}$$

$$\boldsymbol{h}_i^{(k)} = \text{COMBINE}^{(k)}(\boldsymbol{h}_i^{(k-1)}, \text{AGGREGATE}^{(k)}(\{\boldsymbol{h}_j^{k-1}, j \in N(i)\}))、 \tag{11}$$

其中，AGGREGATE( ) 从节点 i 的邻居收集信息，COMBINE( ) 据此更新 i 的表示。SG( )表示停止梯度，用于在 MGM 预训练期间停止梯度流向标记化器。除了 GIN，其他 GNN 也可以作为标记化器。为了获得有意义的图标记，我们会在使用 GNN 作为标记化器之前对其进行预训练 [10]。预训练完成后，该 GNN 将被冻结并用于后续的 MGM 预训练。在第 4 节中，我们将对标记化器常用的图预训练策略进行评估。鉴于基于 GNN 的标记化器提供节点标记，我们将直接最小化图标记与自动编码器标记之间的距离。

输出 $\{\mathbf{z}_i\}_{i=1}^{|V|}$ 的屏蔽节点 V_m，即 $L_O = \frac{1}{|V_m|}\sum_{i \in V_m} \ell(\hat{\mathbf{y}}_i = \mathbf{z}_i, \mathbf{y}_i)$。

2.3 重新审视图形自动编码器

背景：图形自动编码器由图形编码器和图形解码器组成。图形自动编码器由图形编码器和图形解码器组成。我们用
图重建的目标。 一旦预
经过培训后，编码器将被保存下来，用于下游任务。MGM 作品 [12, 2] 通常采用表达式图形编码器，如 GINEs [12] 和 Graph Transformers [2]。然而，对表现性解码器的研究却很有限。之前的许多作品 [12, 10, 11, 22, 2, 25] 都采用了线性或 MLP 解码器，类似于 BERT 的设计 [13]。

表 2：用于编码器和解码器的 GNN 架构比较。

型号	GINE，尺寸 300 变压器，尺寸 128	
线性	-	-
GINE	5 层	-
GINE-Small	3 层	-
GTS	5 层	4 层
小号 GTS	3 层	1 层
GTS-Tiny	1 层	1 层

然而，最近的研究 [15, 38] 揭示了表征学习和重构任务之间的差异。这些研究表明，一个具有足够表现力的解码器可以提高编码器的表征质量。深入研究这些研究后，我们发现了

提高表征质量的两个关键因素：具有足够表现力的解码器和重任务解码 [9]。

具有充分表现力的解码器。继 [15, 38] 之后，我们设计了编码器架构的较小版本作为解码器。我们采用 GINE [12] 和 **GraphTrans** [27]（简称 **GTS**）作为编码器。GTS 在 GINE 层之上堆叠了转换器层，以提高全局交互建模能力。表 2 总结了我们在这项工作中比较的它们的不同版本。

Remask 解码 [9]。 Remask 控制编码器和解码器的焦点 $\{\boldsymbol{h}_i \mid i \in V\}$，让 $\boldsymbol{h}_i$ $i \in V$ 是编码器对屏蔽图的节点隐藏表示。重掩码解码会通过特殊标记 $\mathbf{m}_1 \in \mathrm{R}^d$ 再次掩码被掩码节点的隐藏表示 $_m$，然后再将其输入解码器。形式上，重掩码隐藏表示 $\tilde{\mathbf{h}}_i$ 如下：

$$\tilde{\boldsymbol{h}}_i = \begin{cases} \mathbf{m}_1, & \forall i \in V_m \\ \boldsymbol{h}_i, & \text{否则} \end{cases}. \tag{12}$$

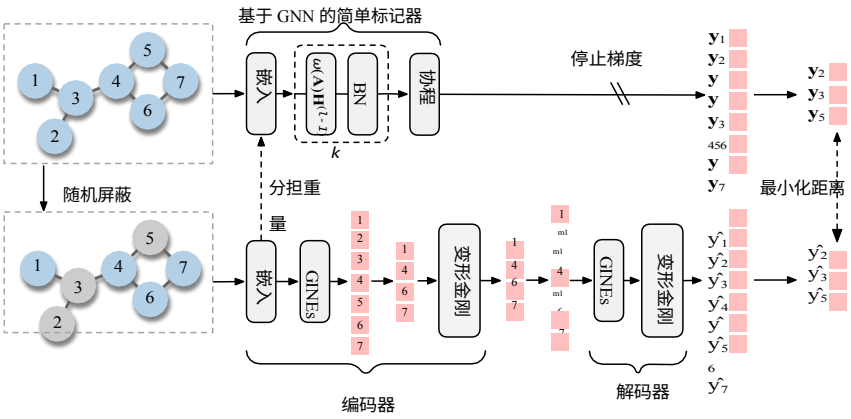


图 4：SimSGT 框架概览。

Remask 通过消除编码器对屏蔽部分的表示，限制编码器预测损坏信息的能力。编码器必须为未屏蔽部分生成有效的表示，以便为解码器提供图形重建信号。

3 方法

在本节中，我们将介绍我们的方法--基于简单 GNN 标记符号的屏蔽图建模（**SimSGT**）（图 4）。具体来说，它的编码器和解码器都采用了 GTS [27] 架构。SimSGT 采用**基于**简单 GNN 的标记符号**生成器** (**SGT**)，并采用新的重掩码策略来解耦 GTS 架构的编码器和解码器。

基于简单 GNN 的标记器。SGT 通过去除 GNN 层中的非线性更新函数，简化了现有的基于聚合的 GNN [37]。它的灵感来自于一些研究，这些研究表明精心设计的图算子可以生成有效的节点表示法 [39，40]。形式上，*k 层*SGT 是：

$$tok_{\text{SGT}}(g) = \{\mathbf{y}_i = \text{SG}([\mathbf{H}^{(1)}_i, \ldots, \mathbf{H}^{(k)}_i]) \mid i \in V\}、 \tag{13}$$

$$\mathbf{H}^{(0)}_i = \text{Embedding}(x_i) \in \mathbb{R}^d、 \quad \forall i \in V、 \tag{14}$$

$$\hat{\mathbf{H}}^{(l)} = \omega(\mathbf{A}) \cdot \mathbf{H}^{(l-1)} \in \mathbb{R}^{|V| \times d}、 \quad 1 \le l \le k、 \tag{15}$$

$$\mathbf{H}^{(l)} = \text{BatchNorm}(\hat{\mathbf{H}}^{(l)})、 \tag{16}$$

其中，Embedding$(\cdot)$ 是一个线性层，使用编码器节点嵌入函数的权重；

$\mathbf{H}^{(l)}_i$ 是 $\mathbf{H}^{(l)}$ 的*第* i 行；BatchNorm( ) 是标准的批归一化层，不含可训练的缩放和移动参数[41]；$\omega(\mathbf{A})$ 是表示原始 GNN 聚合函数的图算子。例如，GIN 的 $\omega(\mathbf{A}) = \mathbf{A} + (1+\epsilon)\mathbf{I}$ 和 GCN 的 $\omega(\mathbf{A}) = \tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2}$，其中 $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ 和 $\tilde{\mathbf{D}}$ 是 $\tilde{\mathbf{A}}$ 的度矩阵。

请注意，SGT 没有可训练的权重，因此无需预训练即可部署。它的标记化能力依赖于图算子 $\omega(\mathbf{A})$，该算子总结了每个节点的邻居信息。此外，我们将每个 SGT 层的输出连接起来，以包含多尺度信息。实验表明，SGT 将原始 GNN 转化为有效的标记化器。

图形自动编码器。SimSGT 采用 GTS 架构作为编码器，并采用较小版本的 GTS（*即*表 2 中的 GTS-Small）作为解码器。这一架构沿用了之前作品 [15, 38] 中的非对称编码器-解码器设计。此外，我们还提出了一种名为 **remask-v2** 的新重掩码策略，以解耦 SimSGT 的编码器层和解码器层、

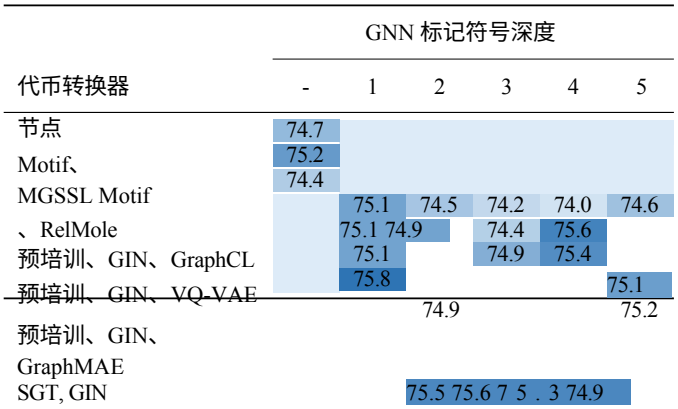
Remask-v2。Remask-v2 通过在转换器层之前丢弃被掩码节点*的*$_m$ 表示来限制编码器预测损坏信息的能力（图 4）。在变换器层之后，我们填充了特殊的掩码标记 $\mathbf{m}_1$ R^d 来弥补之前丢弃的节点的隐藏表示。与原始的 remask 相比，remask-v2 增加了防止转换器层处理屏蔽节点的功能。因此，它避免了在预训练中处理掩码节点而在微调中不处理掩码节点的缺陷 [15]。

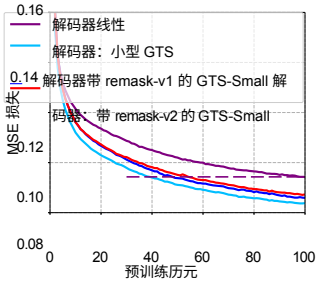
表 3：MoleculeNet 中八个分类数据集的平均 ROC-AUC 分数（%）。

(a) 消融解码器解码器是 GIN 的单层 SGT。

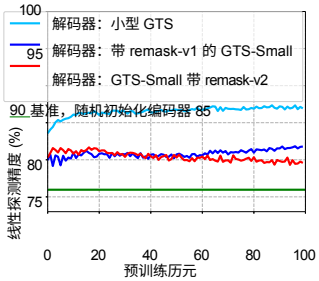
编码器	解码器	重置	平均值
GINE	Linear	-	73.2
GINE	GINE-Small	-	73.0
GINE	GINE-Small	v1	**74.4**
GTS	线性	-	74.1
GTS	小号 GTS	-	74.1
GTS	小号 GTS	v1	75.2
GTS	小号 GTS	v2	**75.8**

(b) 消融标记符。蓝色越深表示性能越高。编码器为 GTS，解码器为 GTS-Small。使用 Remask-v2。





(a) 预训练的 MSE 损失



(b) 屏蔽原子的探针编码器。

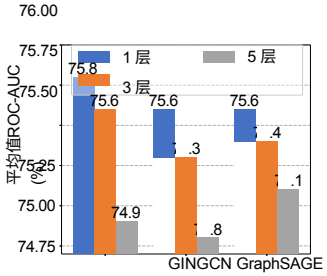


图 6：带有不同 GNN 的 SGT 的 MGM。GTS 编码器

图 5：GTS 编码器。令牌器是 GIN 的单层 SGT。

4 反思分子的遮蔽图建模

实验环境。在本节中，我们将通过实验来评估标记器和解码器在分子 MGM 中的作用。我们的实验采用了 [12, 9] 中的迁移学习设置。我们在 ZINC15 [42] 中的 200 万个分子上对 MGM 模型进行预训练，并在 MoleculeNet [28] 中的八个分类数据集上对预训练模型进行评估：BBBP、Tox21、ToxCast、Sider、ClinTox、MUV、HIV 和 Bace。这些下游数据集按支架拆分分为训练集/验证集/测试集，以提供分布外评估环境。我们报告了十个随机种子在下游数据集上的平均性能和标准偏差。在整个实验中，我们使用比率为 0.35 的随机节点屏蔽。更详细的实验设置见附录 D。

4.1 重新思考解码器

我们研究了富有表现力的解码器对 MGM 性能的影响。这些实验使用了基于单层 GIN 的 SGT 标记器。表 3a 总结了实验结果。

发现 1.具有足够表现力的解码器和重任务解码对 MGM 至关重要。表 3a 显示，使用具有足够表现力的解码器和重任务解码可以显著提高下游性能。这可以归因于分子重构和 MRL 任务之间的差异：自动编码器的最后几层将专门用于分子重构，而失去了一些表示能力[15,

38]。使用线性解码器时，编码器的最后几层将专门用于重构，这可能会在微调中产生次优结果。

一个具有足够表现力的解码器对于解释重构专业化至关重要。如图 5a 所示，与线性解码器相比，采用表现力强的解码器可显著降低重建损耗。然而，在不进行重掩码解码的情况下提高解码器的表现力并不能改善下游性能（表 3a）。这就引出了我们对重掩码的探索。

发现 2.Remask 限制了编码器在图 Table 4: 测试 解码器的**工作。重建有效的 MRL。**图 5b 显示，线性size. Remask-v2 是 used. 编码器预训练的被掩蔽原子类型的探测精度

编码器	解码器	平均值
	GTSGTS-Tiny	74.7
	GTSGTS-Small	**75.8**
GTS	GTS	74.9

有 remask 的编码器的准确率明显低于没有 remask 的编码器的准确率。这说明 remask 使编码器在预测损坏信息上花费的精力更少。此外，当与 GTS-Small 解码器搭配使用时，remask 只会稍微牺牲自动编码器的重构能力（图 5a）。结合

从表 3a 中可以看出，remask 提高了下游性能，这表明remask 限制了编码器在图形重构方面的工作，使其能够专注于 MRL。

此外，采用 GTS 架构的 remask-v2 优于 remask-v1。这一改进可归因于 remask-v2 能够阻止转换器层处理屏蔽节点，避免了在预训练中使用屏蔽节点而在微调中不使用屏蔽节点的缺陷。最后，我们在表 4 中测试了解码器的大小。当编码器为 GTS 时，GTS-Small 解码器的性能最佳。

4.2 重新思考代币转换器

我们研究了标记化器对 MGM 性能的影响。在下面的实验中，图自动编码器采用了 GTS 编码器和带有 remask-v2 的 GTS-Small 解码器，因为它们在上一节中表现出色。实验结果汇总于表 3b。

比较标记符。我们以节点标记器为基准。对于基于图案的标记符，我们采用了领先的片段化方法：MGSSL [1] 和 RelMole [18]。对于基于 GNN 的标记化器，我们比较了流行的预训练策略--GraphCL [4]、GraphMAE [9] 和 VQ-VAE [10, 43]--在 ZINC15 的 200 万个分子上预训练标记化器。

发现 3.重建子图层标记可产生 MRL。表 3b 显示，在适当的设置下，在 MGM 中加入基于图案的标记符或基于 GNN 的标记符可以提供比节点标记符更好的下游性能。这一观察结果强调了在 MGM 中应用子图层标记器的重要性。

发现 4.单层 SGT 性能优于或与其他标记化器不相上下。表 3b 显示，应用于 GIN 的单层 SGT 可提供与 VQ-VAE 预训练的四层 GIN 相当的性能，并超越其他标记化器。此外，图 6 显示 SGT 可以将 GCN 和 GraphSAGE 转变为具有竞争力的标记化器。我们将 SGT 的出色表现归功于其图形算子在提取结构信息方面的有效性。已有研究表明，线性图算子能有效总结节点分类的结构模式 [39，40]。

发现 5.基于 GNN 的标记化器比基于图案的标记化器性能更高。我们假设这是由于 GNN 能够总结结构模式。当使用 GNN 表示法作为重构目标时，目标之间的距离反映了它们底层子图之间的相似性--这是一种微妙的关系，而主题的单次编码无法捕捉到这种关系。我们将把 GNN 纳入基于主题的标记化器的可能性留待未来的工作中进行探讨。

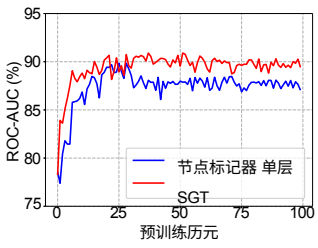


图 7：线性探测编码器的 FG 输出。

最后，我们表明，虽然基于 GNN 的标记化器与化学知识无关，但将其纳入 MGM 可以提高对 FG 的识别率。在图 7 中，我们使用线性分类器来识别 FGs。探测编码器的均值池输出，以预测分子内的 FG。我们使用 RDkit [30] 提取了 85 种 FG。详情见附录 D。可以看出，与节点标记器相比，在 MGM 中加入单层 SGT 提高了编码器识别

FG 的能力。

5 与最新方法的比较

在本节中，我们将把 SimSGT 与用于分子性质预测和更广泛下游任务的主要分子 SSL 方法进行比较。为了进行公平比较，我们报告了 SimSGT 及其使用 GINE 架构的变体的性能。该变体采用 GINE 作为编码器，GINE-Small 作为解码器（表 2），并实现了 remask-v1 解码。

分子性质预测。分子性质预测实验采用了与第 3 节相同的迁移学习设置。表 5 列出了实验结果。可以看出，采用 GTS 和 GINE 架构的 SimSGT 在平均性能上优于所有基线。值得注意的是，使用 GTS 的 SimSGT 创造了 75.8% 的 ROC-AUC 新纪录。它的平均性能比第二种方法高出 1.8 个百分点，并在八个分子性质预测数据集中的五个数据集上取得了最佳性能。使用 GINE 的 SimSGT 在平均性能上比基线方法高出 0.4%。这些改进证明了我们提出的标记化器和解码器在提高分子 MGM 性能方面的有效性。

表 5：八个 MoleculeNet 数据集的迁移学习 ROC-AUC (%) 分数。**粗体**表示最佳性能。基线使用源代码重现。所有方法均使用 GTS 编码器。

数据	集BBBP Tox21 ToxCast Sider ClinTox MUV					艾滋病毒		平均
	贝采							值
没有预培训	68.7±1.3	75.5±0.5	64.5±0.8	58.0±1.0	71.2±4.9	69.7±1.5	74.2±2.2	77.2±1.6	69.9
Infomax	69.2±0.7	74.6±0.5	61.8±0.8	60.1±0.7	74.8±2.7	74.8±1.5	75.0±1.3	76.3±1.8	70.8
上下文预设	68.5±1.1	75.3±0.6	63.2±0.5	61.3±0.8	74.9±3.2	70.9±2.4	76.8±1.1	80.2±1.8	71.4
图表	70.4±1.1	73.8±1.0	63.1±0.4	60.4±1.3	77.8±3.0	73.8±2.0	75.6±0.9	78.3±1.1	71.6
JOAO	70.8±0.6	75.2±1.5	63.8±0.8	61.0±0.9	78.7±3.0	75.7±2.6	77.0±1.2	77.5±1.2	72.5
ADGCL	67.9±1.0	73.0±1.2	63.2±0.4	60.2±1.0	78.8±1.6	75.6±2.5	75.5±1.3	73.4±1.6	71.0
BGRL	**72.7±0.9**	75.8±1.0	65.1±0.5	60.4±1.2	77.6±4.1	76.7±2.8	77.1±1.2	74.7±2.6	72.5
图表LOG	67.6±1.6	76.0±0.8	63.6±0.6	59.8±2.1	79.1±3.2	72.8±1.8	72.5±1.6	83.6±2.0	71.9
RGCL	71.4±0.8	75.7±0.4	63.9±0.3	60.9±0.6	80.0±1.6	75.9±1.2	77.8±0.6	79.9±1.0	73.2
S2GAE	67.6±2.0	69.6±1.0	58.7±0.8	55.4±1.3	59.6±1.1	60.1±2.4	68.0±3.7	68.6±2.1	63.5
鼹鼠-贝特	70.8±0.5	76.6±0.7	63.7±0.5	59.2±1.1	77.2±1.4	77.2±1.1	76.5±0.8	82.8±1.4	73.0
GraphMAE	71.7±0.8	76.0±0.9	65.8±0.6	60.0±1.0	79.2±2.2	76.3±1.9	75.9±1.8	81.7±1.6	73.3
GraphMAE2	71.6±1.6	75.9±0.8	65.6±0.7	59.6±0.6	78.8±3.0	78.5±1.1	76.1±2.2	81.0±1.4	73.4
SimSGT	72.2±0.9	**76.8±0.9**	**65.9±0.8**	**61.7±0.8**	**85.7±1.8**	**81.4±1.4**	**78.0±1.9**	**84.3±0.6**	**75.8**

表 6：分子性质预测（回归）和药物靶点亲和性预测（回归）的迁移学习性能。**粗体**表示最佳性能，下划线表示次佳性能。* 表示使用已发布代码的重现结果。其他基线结果借鉴自 [ICLR'22]。

	分子性质预测 (RMSE ↓) 药物-靶标亲和力 (MSE ↓) ESOL Lipo Malaria CEP							
	Avg. 戴维斯 KIBA 平均值							
没有预培训	1.178±0.044	0.744±0.007	1.127±0.003	1.254±0.030	1.076	0.286±0.006	0.206±0.004	0.246
上下文预设	1.196±0.037	0.702±0.020	1.101±0.015	1.243±0.025	1.061	0.279±0.002	0.198±0.004	0.238
AttrMask	1.112±0.048	0.730±0.004	1.119±0.014	1.256±0.000	1.054	0.291±0.007	0.203±0.003	0.248
JOAO	1.120±0.019	0.708±0.007	1.145±0.010	1.293±0.003	1.066	0.281±0.004	0.196±0.005	0.239
GraphMVP	1.064±0.045	0.691±0.013	1.106±0.013	1.228±0.001	1.022	0.274±0.002	0.175±0.001	0.225
鼹鼠-伯特*	1.192±0.028	0.706±0.008	1.117±0.008	1.078±0.002	1.024	0.277±0.004	0.210±0.003	0.243
SimSGT, GINE	1.039±0.012	**0.670±0.015**	1.090±0.013	1.060±0.011	0.965	0.263±0.006	**0.144±0.001**	0.204
SimSGT, GTS	**0.917±0.028**	0.695±0.012	**1.078±0.012**	**1.036±0.022**	**0.932**	**0.251±0.001**	0.153±0.001	**0.202**

下游任务范围更广。我们报告了迁移学习在分子性质回归预测和药物-靶标亲和力（DTA）任务中的表现。DTA 的目的是预测分子药物与靶蛋白之间的亲和力得分[44, 29]。具体来说，我们用 SimSGT 预训练编码器替代 [44] 中的分子编码器来评估 DTA 性能。按照文献[45]中的实验设置，我们在 GEOM 数据集[46]的 5 万个分子样本上对 SimSGT 进行了预训练，并报告了三个随机种子的平均性能和标准偏差。我们报告了采用支架拆分的分子性质预测数据集的 RMSE，并报告了采用随机拆分的 DTA 数据集的 MSE。结果汇总于表 6。可以看出，与基线模型相比，SimSGT 取得了显著的改进。

表 7：QM 数据集的平均误差 (MAE) 性能。使用 GTS 编码器。

	QM7	QM8	QM9
#任务	1	12	12

图表	80.4±3.3	0.0200±0.0004	5.76±0.37
GraphMAE	78.4±2.3	0.0190±0.0003	5.84±0.16
鼹鼠-贝特	79.8±2.6	0.0190±0.0003	5.75±0.16
SimSGT	**75.4±0.7**	**0.0183±0.0003**	**5.53±0.25**

量子化学性质预测。我们报告了预测分子量子化学性质的性能[47]。按照文献[48]，我们将下游数据集按支架拆分。本实验重复使用表 5 中的模型检查点，这些检查点在 ZINC15 的 200 万个分子上进行了预训练。具体来说，我们在预训练的分子编码器后附加了一个双层 MLP，并对性质预测模型进行了微调。我们报告了平均性能

表 8：使用 GTS 编码器在 ZINC15 上进行 100 次预训练所花费的时间。

模型	GraphMAE	Mole-BERT	S2GAE	GraphMAE2	SimSGT
预训练时间	527 分钟	2199 分钟	1763 分钟	1195 分钟	645 分钟

和标准偏差。表 7 报告了这些性能。我们可以看到，SimSGT 的性能始终优于具有代表性的基线 GraphCL、GraphMAE 和 Mole-BERT。

计算成本我们在表 8 中比较了 SimSGT 和主要基线的壁钟预训练时间。我们可以发现1) SimSGT 的预训练时间与 GraphMAE [9] 相当。这种效率主要归功于我们的 SGT 标记符号化器的最小计算开销；2）与分子 SSL 的先前基准 Mole-BERT [10]相比，SimSGT 快了约三倍。Mole-BERT 的计算需求可归因于其结合了 MGM 训练和对比学习的方法。

6 结论和未来工作

在这项工作中，我们广泛研究了分子 MGM 中标记符和解码器的作用。我们编译并评估了作为分子标记符号化器的各种分子破碎函数。结果表明，子图级标记符号化器具有 MRL 性能。此外，我们还通过经验分析表明，具有足够表现力的解码器和重掩码解码器可以提高分子编码器的表示质量。鉴于这些发现，我们介绍了 SimSGT，一种带有子图层标记的新型 MGM 方法。SimSGT 的特点是基于简单 GNN 的标记器，能够将 GNN 转化为有效的图标记器。它的编码器和解码器进一步采用了 GTS 架构，并采用了新的重掩码策略。与现有的分子 SSL 方法相比，SimSGT 有了很大的改进。对于未来的工作，分子标记器在分子-文本联合建模中的潜在应用[3]仍然是一个有趣的方向。

鸣谢

本研究由国家自然科学基金（9227010114）和安徽省高校协同创新项目（GXXT-2022-040）资助。本材料基于谷歌云研究信贷计划（6NW8- CF7K-3AG4-1WH1）支持的工作。本研究由 NExT 研究中心支持。

参考资料

[1] Zaixi Zhang, Qi Liu, Hao Wang, Chengqiang Lu, and Chee-Kong Lee.用于分子性质预测的基于图案的图自监督学习。In *NeurIPS*, pages 15870-15882, 2021.

[2] 于荣、边亚涛、徐汀阳、谢伟阳、魏颖、黄文兵、黄俊洲。大规模分子数据的自监督图转换器。2020年*NeurIPS*大会

[3] 刘志远、李思航、罗彦辰、费浩、曹一心、川口健二、王翔、蔡达生。Molca：使用跨模态投影仪和单模态适配器的分子图语言建模。*EMNLP*, 2023.

[4] 尤宁、陈天龙、隋永铎、陈婷、王占洋和沈洋。带有增强功能的图形对比学习2020年*NeurIPS*大会

[5] Laurianne David、Amol Thakkar、Rocío Mercado 和 Ola Engkvist。人工智能驱动药物发

现中的分子表征：综述与实践指南》。*化学信息学杂志*》，12（1）：1- 22，2020 年。

[6] Justin Gilmer、Samuel S Schoenholz、Patrick F Riley、Oriol Vinyals 和 George E Dahl。量子化学的神经信息传递。在 *ICML 上*，第 1263-1272 页。PMLR, 2017.

[7] Chaochao Yan, Qianggang Ding, Peilin Zhao, Shuangjia Zheng, Jinyu Yang, Yang Yu, and Junzhou Huang.Retroxpert：像化学家一样分解逆合成预测。*NeurIPS*, 2020.

[8] Umit V Ucak, Islambek Ashyrmamatov, Junsu Ko, and Juyong Lee.通过原子环境的神经机器翻译预测逆合成反 应 途径。*自然通讯*，13（1）：1186，2022。

[9] Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, Chunjie Wang, and Jie Tang.Graphmae：自监督屏蔽图自编码器。在 *KDD* 中，第 594-604 页。ACM, 2022.

[10] Jun Xia、Chengshuai Zhao、Bozhen Hu、Zhangyang Gao、Cheng Tan、Yue Liu、Siyuan Li 和 Stan Z. Li。Mole-BERT：反思分子的预训练图神经网络。*第十一届学习表征国际会议*，2023 年。

[11] 游宇宁、陈天龙、王占阳、沈阳。自监督何时有助于图卷积网络？在 *ICML 上*，《*机器学习研究论文集》第*119 卷，第 10871-10880 页。PMLR, 2020.

[12] 胡伟华、刘博文、约瑟夫-戈麦斯、马林卡-齐特尼克、梁珀西、维杰-潘德和尤雷-莱斯科维奇。预训练图神经网络的策略。在 2020 年的 *ICLR 上*。

[13] Jacob Devlin、Ming-Wei Chang、Kenton Lee 和 Kristina Toutanova。BERT：用于语言理解的深度双向变换器预训练。在 *NAACL-HLT* 中，第 4171-4186 页。计算语言学协会，2019 年。

[14] Colin Raffel、Noam Shazeer、Adam Roberts、Katherine Lee、Sharan Narang、Michael Matena、Yanqi Zhou、Wei Li 和 Peter J. Liu。用统一的文本到文本转换器探索迁移学习的极限。*J. Mach.Learn.*21:140:1-140:67, 2020.

[15] 何开明、陈新磊、谢赛宁、李阳浩、Piotr Dollár 和 Ross B. Girshick。遮蔽式自动编码器是可扩展的视觉学习器。*CVPR*，第 15979-15988 页。IEEE, 2022.

[16] Hangbo Bao、Li Dong、Songhao Piao 和 Furu Wei。Beit：图像变换器的 BERT 预训练。In *ICLR*.OpenReview.net, 2022.

[17] 孙梦莹、邢晶、王慧君、陈斌和周家玉。Mocl：通过分子图的知识感知对比学习实现数据驱动的分子指纹。In *KDD*, pages 3585-3594.ACM, 2021.

[18] 季泽伟、史润汉、陆家瑞、李芳和杨阳。Relmole：基于两级图相似性的分子表征学习。*化学信息建模学报》*，62（22）：5361-5372，2022。

[19] Aditya Ramesh、Mikhail Pavlov、Gabriel Goh、Scott Gray、Chelsea Voss、Alec Radford、Mark Chen 和 Ilya Sutskever。零镜头文本到图像生成。In *ICML*, pages 8821-8831, 2021.

[20] Mukund Deshpande、Michihiro Kuramochi、Nikil Wale 和 George Karypis。基于频繁子结构的化合物分类方法。*IEEE 知识与数据工程论文集》*，17（8）：1036-1050，2005 年。

[21] 乔纳森-克莱登、尼克-格里夫斯和斯图尔特-沃伦。*有机化学》*。牛津大学出版社，2012 年。

[22] 李　金堂、吴若凡、孙望斌、陈亮、田胜、朱亮、孟长华、郑子斌、王伟强。Maskgae：屏蔽图建模与图自动编码器。*CoRR*，abs/2205.10053，2022。

[23] Chun Wang, Shirui Pan, Guodong Long, Xingquan Zhu, and Jing Jiang.MGAE：用于图聚

类的边际图自编码器。In *CIKM*, pages 889-898.ACM, 2017.

[24] Sixiao Zhang、Hongxu Chen、Haoran Yang、Xiangguo Sun、Philip S. Yu 和 Guandong Xu。带变换器的图掩码自动编码器。*CoRR*，abs/2202.08391，2022。

[25] Ganqu Cui、Jie Zhou、Cheng Yang 和 Zhiyuan Liu.属性图嵌入的自适应图编码器。*KDD* 中，第 976-985 页。ACM, 2020.

[26] Xiao Liu, Shiyu Zhao, Kai Su, Yukuo Cen, Jiezhong Qiu, Mengdi Zhang, Wei Wu, Yuxiao Dong, and Jie Tang.掩码与推理：预训练复杂逻辑查询的知识图谱转换器。在 *KDD* 中，第 1120-1130 页。ACM, 2022.

[27] Zhanghao Wu, Paras Jain, Matthew A. Wright, Azalia Mirhoseini, Joseph E. Gonzalez, and Ion Stoica.用全局注意力表示图神经网络的长程上下文。In *NeurIPS*, pages 13266-13279, 2021.

[28] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande.Moleculenet：分子机器学习的基准。*化学科学*》，9（2）：513-530，2018。

[29] Tapio Pahikkala、Antti Airola、Sami Pietilä、Sushil Shakyawar、Agnieszka Szwajda、Jing Tang 和 Tero Aittokallio。迈向更真实的药物-靶点相互作用预测。*生物信息简报》*，16（2）：325-337，2015。

[30] Greg Landrum.Rdkit 文档。*Release*, 1(1-79):4, 2013.

[31] Elena S Salmina、Norbert Haider 和 Igor V Tetko。扩展官能团（efg）：化合物化学特征和结构-活性关系研究的高效集合。*分子*》，21（1）：1，2015.

[32] Wengong Jin, Regina Barzilay, and Tommi S. Jaakkola.Jaakkola.用于分子图生成的结点树变分自动编码器。在《*机器学习研究论文集》*第 80 卷 *ICML 中，*第 2328-2337 页。PMLR, 2018.

[33] Daylight Chemical Information Systems, Inc.《日光 *理 论 手 册 》* 。http://www.daylight.com/dayhtml/doc/theory/theory.smarts.html.

[34] 乔纳森-克莱登、尼克-格里夫斯和斯图尔特-沃伦。*有机化学》*。牛津大学出版社，2012 年。

[35] 约尔格-德根、克里斯托夫-韦格沙伊德-格拉赫、安德烈娅-扎里亚尼和马蒂亚斯-拉瑞。关于编译和使用 "类药物 "化学片段空间的艺术。*ChemMedChem：ChemMedChem: Chemistry Enabling Drug Discovery*, 3(10):1503-1507, 2008.

[36] Reinhard Diestel.*图论》，第 4 版，《数学研究生文集》第* 173 卷。Springer, 2012.

[37] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka.图神经网络有多强大？In *ICLR*, 2019.

[38] Christoph Feichtenhofer、范浩琦、李阳浩、何开明。作为时空学习器的掩码自动编码器。见 Alice H. Oh、Alekh Agarwal、Danielle Belgrave 和 Kyunghyun Cho 编著的《2022 年 *NeurIPS》*。

[39] Felix Wu, Amauri H. Souza Jr., Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Q. Wein- berger.简化图卷积网络。*机器学习研究论文集》第* 97 卷，第 6861-6871 页。PMLR，2019。

[40] 陈磊、陈正道、琼-布鲁纳。论图神经网络与图增强 mlps。In *ICLR*.OpenReview.net, 2021.

[41] Sergey Ioffe 和 Christian Szegedy.批量归一化：通过减少内部协变量偏移加速深度网络训练。In *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 448-456.JMLR.org, 2015.

[42] Teague Sterling 和 John J. Irwin.ZINC 15 - 人人都能发现的配体。*J. Chem.Inf.Model.*, 55(11):2324-2337, 2015.

[43] Ali Razavi、Aäron van den Oord 和 Oriol Vinyals。用 VQ-VAE-2 生成多样化的高保真图像。In *NeurIPS*, pages 14837-14847, 2019.

[44] Thin Nguyen、Hang Le、Thomas P. Quinn、Tri Nguyen、Thuc Duy Le 和 Svetha

Venkatesh。Graphdta: predicting drug-target binding affinity with graph neural networks.*生物信息》*，37（8）：1140-1147，2021 年。

[45] 刘胜超、王汉臣、刘伟阳、Joan Lasenby、郭宏宇和唐健。用三维几何预训练分子图表示。In *ICLR*.OpenReview.net, 2022.

[46] Simon Axelrod 和 Rafael Gómez-Bombarelli.GEOM：用于性质预测和分子生成的能量注释分子构象。*CoRR*，abs/2006.05531，2020。

[47] 量子机器。http://quantum-machine.org/datasets/。访问日期：2023-03。

[48] 方晓敏、刘力航、雷洁琼、何东龙、张善卓、周静波、王帆、吴华、王海峰。用于性质预测的几何增强型分子表征学习。*Nat.机器。*4(2):127-134, 2022.

[49] Emanuele Rossi、Fabrizio Frasca、Ben Chamberlain、Davide Eynard、Michael M. Bronstein 和 Federico Monti。SIGN：可扩展的入门图神经网络。*CoRR*，abs/2004.11198，2020.

[50] Wengong Jin, Regina Barzilay, and Tommi S. Jaakkola.Jaakkola.使用结构主题分层生成分子图。*机器学习研究论文集》第*119 卷第 4839-4848 页。PMLR, 2020.

[51] Shichang Zhang, Ziniu Hu, Arjun Subramonian, and Yizhou Sun.图表征的动机驱动对比学习。*arXiv 预印本 arXiv:2012.12533*, 2020.

[52] Tom B. Brown、Benjamin Mann、Nick Ryder、Melanie Subbiah、Jared Kaplan、Prafulla Dhariwal、Arvind Neelakantan、Pranav Shyam、Girish Sastry、Amanda Askell、Sandhini Agarwal、Ariel Herbert-Voss、Gretchen Krueger、Tom Henighan、Rewon Child、Aditya Ramesh、Daniel M. Ziegler、Jeff Wu、Clemens Winter、Christopher Hesse、Mark Chen、Eric Sigler、Mateusz Litwin、Scott Gray。Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei.语言模型是少量学习者。2020年，*NeurIPS*。

[53] 史蒂文-伯德Nltk：自然语言工具包。*COLING/ACL 2006 交互式演示会论文集》*，第 69-72 页，2006 年。

[54] Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N. Gomez、Lukasz Kaiser 和 Illia Polosukhin。注意力就是你所需要的一切。In *NIPS*, pages 5998-6008, 2017.

[55] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer.会倾听的掩码自动编码器。在 2022 年的 *NeurIPS 会议上*。

[56] Alan Baade、Puyuan Peng 和 David Harwath。MAE-AST：屏蔽自动编码音频频谱图变换器。*INTERSPEECH*，第 2438-2442 页。ISCA，2022 年。

[57] Alexey Dosovitskiy、Lucas Beyer、Alexander Kolesnikov、Dirk Weissenborn、Xiaohua Zhai、Thomas Unterthiner、Mostafa Dehghani、Matthias Minderer、Georg Heigold、Sylvain Gelly、Jakob Uszkoreit 和 Neil Houlsby。一幅图像胜过 16x16 个单词：规模图像识别变换器。In *ICLR*.OpenReview.net, 2021.

[58] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky.斯坦福 corenlp 自然语言处理工具包。In *ACL (System Demonstrations)*, pages 55-60, 2014.

[59] Rico Sennrich、Barry Haddow 和 Alexandra Birch。使用子词单元的罕见词神经机器翻译。In *ACL (1)*.计算机语言学协会，2016 年。

[60] Jean-Bastien Grill、Florian Strub、Florent Altché、Corentin Tallec、Pierre H. Richemond、Elena Buchatskaya、Carl Doersch、Bernardo Ávila Pires、Zhaohan Guo、Mohammad Gheshlaghi Azar、Bilal Piot、Koray Kavukcuoglu、Rémi Munos 和 Michal Valko。引导你自己的潜变量--自我监督学习的新方法。2020年，*NeurIPS*。

[61] 陈鑫磊、何开明。探索简单的连体表示学习。*CVPR*， 2021。

[62] Shantanu Thakoor、Corentin Tallec、Mohammad Gheshlaghi Azar、Mehdi Azabou、Eva L Dyer、Remi Munos、Petar Velic˘kovic´ 和 Michal Valko。通过引导对图进行大规模表示学习。2022 年，*ICLR*。

[63] Beatrice Bevilacqua, Fabrizio Frasca, Derek Lim, Balasubramaniam Srinivasan, Chen Cai, Gopinath Balamurugan, Michael M. Bronstein, and Haggai Maron.等变子图聚合网络。In *ICLR*.OpenReview.net, 2022.

[64] Lingxiao Zhao, Wei Jin, Leman Akoglu, and Neil Shah.从恒星到子图：用局部结构意识提升任何 GNN。In *ICLR*.OpenReview.net, 2022.

[65] Ziqiao Zhang, Jihong Guan, and Shuigeng Zhou.Fragat：用于分子性质预测的面向片段的多尺度图注意模型。*生物信息*，37（18）：2981-2987，2021。

[66] Fabrizio Frasca, Beatrice Bevilacqua, Michael M. Bronstein, and Haggai Maron.通过重新思考对称性来理解和扩展子图gnns。在 2022 年的 *NeurIPS 上*。

[67] 钱晨迪、高拉夫-拉坦、弗洛里斯-盖茨、马蒂亚斯-尼佩特、克里斯托弗-莫里斯。有序子图聚合网络。2022年，*NeurIPS*。

[68] Bohang Zhang, Guhao Feng, Yiheng Du, Di He, and Liwei Wang.通过子图 weisfeiler-lehman 检验的子图 gnns 的完整表达性层次结构. *ArXiv 预印本 arXiv:2302.07090*, 2023.

[69] Petar Velic˘kovic´, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm.Deep graph infomax.In *ICLR*, 2019.

[70] 孙凡云、乔丹-霍夫曼、维卡斯-维尔马和唐健。信息图：通过互信息最大化实现无监督和半监督图级表示学习。2020 年，*ICLR*。

[71] 尤宁、陈天龙、沈洋和王占洋。图对比自动学习。In *ICML*, Proceedings of Machine Learning Research, pages 12121-12132, 2021.

[72] Susheel Suresh、Pan Li、Cong Hao 和 Jennifer Neville。提高图对比学习的对抗性图增强。2021 年，*NeurIPS*，第 15920-15933 页。

[73] Minghao Xu, Hang Wang, Bingbing Ni, Hongyu Guo, and Jian Tang.具有局部和全局结构的自监督图级表示学习。In *ICML*, volume 139, pages 11548-11558, 2021.

[74] 李思航、王翔、张安、吴迎新、何湘南、蔡达生。让不变原理发现激发图对比学习。In *ICML*, pages 13052-13065, 2022.

[75] Qiaoyu Tan, Ninghao Liu, Xiao Huang, Soo-Hyun Choi, Li Li, Rui Chen, and Xia Hu.S2GAE：自监督图自编码器是具有图掩码的可泛化学习器。In *WSDM*, pages 787-795.ACM, 2023.

[76] 侯振宇、何宇飞、岑玉国、刘晓、董玉晓、叶夫根尼-哈拉莫夫和唐杰。Graphmae2：解码增强型掩码自监督图学习器。*WWW*, volume abs/2304.04779, 2023.

[77] 胡卫华、马蒂亚斯-费、马林卡-齐特尼克、董玉晓、任宏宇、刘博文、米歇尔-卡塔斯塔和尤雷-莱斯科维奇。开放图基准：*ArXiv preprint arXiv:2005.00687*, 2020.

[78] Qimai Li, Zhichao Han, and Xiao-Ming Wu.深入了解半监督学习的图卷积网络。*AAAI*》，第 3538-3545 页。AAAI Press, 2018.

[79] 乔舒亚-戴维-罗宾逊、庄靖尧、苏弗里特-斯拉和斯特凡妮-耶格尔卡。硬负样本对比学习。In *ICLR*.OpenReview.net, 2021.

[80] Yoav Levine、Barak Lenz、Opher Lieber、Omri Abend、Kevin Leyton-Brown、Moshe Tennen- holtz 和 Yoav Shoham。Pmi- 屏蔽：相关跨度的原则性屏蔽。In *ICLR*.OpenReview.net, 2021.

A 局限性

我们对图形标记器和图形解码器的结果和分析仅限于 MGM 预训练任务。不同的标记器和解码器可能会为其他生成建模方法提供优势，如自回归建模[32]。

与标准 GNN（如 GINs）[37]相比，SGT[39, 40]对图结构的表现力有限。从理论上讲，SGT 与标准 GNN 之间的表现力差距会随着 GNN 的深度呈指数增长[40]。然而，如表 3b 所示，SGT 与经过预训练的基于 GNN 的标记化器相比，表现不相上下，甚至更好。我们将这一有趣的观察结果归因于两个关键因素。首先，SGT（*即*简单 GNN）仍然很强大，可以 "区分几乎所有非同构图"[40]。它们在实践中取得了不错的结果 [39, 49]。其次，我们猜想可能存在一种更好的预训练方法来训练基于 GNN 的标记化器，但目前的预训练技术并不能充分发挥 GNN 作为有效标记化器的潜力。事实上，GraphCL 和 VQ-VAE 之间性能的显著差异（表 3b）强调了预训练方法对标记化器性能的影响。我们将把如何有效预训练基于 GNN 的标记化器作为今后的工作重点。

B 相关作品

我们已将有关美高梅的文献综述纳入本文正文。在此，我们将从以下几个方面阐述文献综述。

带图案的分子 SSL。图案是具有统计意义的子图模式 [32, 50]，已被应用于现有的分子 SSL 方法中。自回归预训练方法[32, 1, 50]在每一步生成过程中都会生成图案而不是节点，以提高生成分子的有效性。对比学习 [17, 51, 18] 中也使用了动机。Sun *等人*[17]用化学性质相似的对应物替代分子中的图案，以创建高质量的增强。[51，18]则在分子主题层面构建分子视图，以补充原子层面的原始视图。在预测性预训练中，Rong *等人*[2]通过预训练图编码器来预测分子内部的 FG。这些前人的工作为发现主题开发了大量的分子片段方法。然而，在 MGM 预训练中，这些片段方法作为标记化器却被忽视了。我们的工作总结了常见的片段规则，并检验了所选片段方法在 MGM 预训练中的性能，从而弥补了这一不足。

数据标记化。标记化是一种数据预处理技术，它将原始数据分割成更小的元素并转换成标记。它在 NLP 中被广泛用于将句子分割为词级单元 [13, 52, 53]。由于人们对变形金刚的兴趣日益高涨 [54]，标记化技术也被应用于图像 [15, 16] 和音频 [55, 56]。标记化将这些数据分割成一系列片段，以适应变换器输入和输出的形状。标记化器可以通过启发式方法 [57]、结合领域知识 [58] 以及在目标数据集上进行预训练 [59, 19, 43] 来设计。在这项工作中，我们研究的是图标记符，这在以前的工作中探讨较少。

与对比学习的关系在使用基于 GNN 的标记化器时，MGM 涉及最小化两个网络分支（*即*标记化器分支和自动编码器分支）输出之间的距离。乍一看，这种设计可能与 BYOL [60]、SimSiam [61] 和 BGRL [62]等对比学习方法类似，也是最小化两个网络分支之间的输出差异。然而，仔细观察就会发现 MGM 与这些方法之间有几个关键的区别。首先，MGM 将

未损坏的数据输入标记化器分支，将损坏的数据输入自动编码器分支，鼓励自动编码器重建缺失的信息。相比之下，BYOL、SimSiam 和 BGRL 在它们的两个分支中都使用了损坏的数据，构成了不同的训练目标。其次，BYOL、SimSiam 和 BGRL 的两个分支采用了几乎完全相同的架构，而 MGM 的自动编码器和标记化器则采用了截然不同的架构。在我们表现最好的实验中，自动编码器有十多层 GNN 和变换器，而标记器则是一个浅层单层网络（表 3）。最后，MGM 采用了重任务解码来限制编码器的重构能力，这在对比学习方法中是没有的[60, 61, 62]。

子图增强型图神经网络。子图增强型图神经网络 [63, 64, 65] 是指一类新兴的图神经网络，它在编码前将图分割成子图，以提高图神经网络的表达能力 [66, 67, 68]。常见的图分割方法是按节点分割，如

每个碎片子图都与原始图中的唯一节点相关联。例如，ESAN [63] 通过抽样自我网络或从原始图中删除一个节点来获取子图。给定子图后，子图增强型 GNN 通过应用一系列等变消息传递层生成每个子图中的节点嵌入[63, 64]。最后，将这些嵌入集合起来，输出图嵌入。我们的工作与子图增强型 GNNs 有关，我们也研究图碎片。主要区别在于，我们侧重于使用从这些碎片图中得到的标记作为分子 MGM 的重构目标。

C 伪代码

我们将介绍 SimSGT 的伪代码。该代码以 GIN 的单层 SGT 为例。

算法 1 SimSGT 的 Pytorch 式伪代码

```
## phi: 图形编码器 ##
rho: 图形解码器

def SGT ( g, embed ):
 ## SGT: 单层 GIN 标记器 x, edge_index = g

 # 信息传递
 x = propagate ( embed ( x), edge_index ) + (1+ eps ) * embed ( x)

 # 批量标准化层 x = batchnorm
 ( x)
 返回 x

对于装载机中的 g :
 # 随机屏蔽
 g_hat , m_pos = random_masking ( g) # m_pos : 遮罩位置

 # 嵌入是一个线性层
 y = SGT ( g, phi . embed ). detach () # detach : stop - gradient

 # 自动编码器前进
 y_hat = rho ( remask ( phi ( g_hat ), m_pos ))

 # 尽量减少损失
 loss = distance_loss ( y_hat [ m_pos ], y[ m_pos ])
 loss . backward ()
```

D 实验装置

计算资源。我们在英伟达 DGX A100 服务器上进行实验。每个实验可在单个 GPU 上运行，GPU 内存不超过 30 GB。

D.1 比较方法

基于图案的标记符。下面我们将详细介绍两款基于图案的标记器：

- **MGSSL** [1] 采用 BRICS [35] 方法进行分子破碎（第 2.2 节）。为了获得更精细的片段，MGSSL 采用了两条额外的规则来分解 BRICS 的输出片段：1) 分离连接到循环上的单个原子；2) 如果由三个或更多原子组成的连接子图不属于循环，则将其分解为一个新片段。

- **RelMole** [18] 结合了 Cycles 和 FGs 的分子破碎功能（第 2.2 节）。此外，它还能提取上一步未涉及的碳碳单键作为新片段。

我们使用他们论文中提供的主题词库进行分子片段化。给定一个分子后，我们将其片段化的主题词转换为单击编码，作为重构目标。

基于 GNN 的预训练标记化器。在基于 GNN 的预训练标记化器中，我们使用原子序数作为节点特征，而不使用边缘特征。我们在附录 E 中表明，将边缘特征纳入

表 9：在 ZINC15 的 200 万个分子上进行预训练以及在 MoleculeNet 的八个数据集上进行微调的实验设置：BBBP、Tox21、ToxCast、Sider、ClinTox、MUV、HIV 和 Bace。

(a) 节点和边缘特征

类型		范围
节点功能	原子序数 手性标签	1~118 {未指定、四面体 cw、四面体 ccw、其他}.
边缘特征	键类型 键方向	{单人、双人、三人、芳香族}..... {-、右上角结尾、右下角结尾}。

(b) 超参数

编码器	预训			微调		
	lr	批量大小	纪元	lr	batch 大小	纪元
GINE	1e-3	1024	100	{1e-3、1e-4}	32	100
GTS	1e-4	2048	100	{1e-4、1e-5}	32	100

GNN 标记化器中的边缘特征会降低性能。由于去除了边缘特征，标记化器使用的是 GIN [37] 的架构，而不是 GINE [12]。我们已经报告了通过 GraphCL [4]、GraphMAE [9] 和 VQ-VAE [10, 43] 预训练的基于 GNN 的标记化器的性能。VQ-VAE 的实现遵循 [10]，并按原子序数对潜码进行分组。我们严格按照上述论文中的程序对 GNN 进行预训练，然后将其用作标记化器。

基于简单 GNN 的标记化器（SGT）。SGT 使用原子序数节点特征。它使用图编码器的原子序数线性嵌入函数。下面我们将介绍经过测试的 SGT 的图算子：

$$\text{杜松子酒} \quad \omega(\mathbf{A}) = \mathbf{A} + (1+\epsilon)\mathbf{I}、 \tag{17}$$

$$\text{GCN：} \quad \omega(\mathbf{A}) = \tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}、^{-1/2} \tag{18}$$

$$\text{GraphSAGE：} \quad \omega(\mathbf{A}) = \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{A}}、 \tag{19}$$

其中，$\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$，$\tilde{\mathbf{D}}$是 $\tilde{\mathbf{A}}$ 的度矩阵；ϵ 根据经验设为 0.5。

基线。现在，我们将详细介绍所报告的基线方法：

- **Infomax** [69] 通过最大化节点补丁局部摘要与补丁图层全局摘要之间的互信息来学习节点表示。
- **ContextPred** [12] 使用子图的嵌入来预测它们的上下文图结构。
- **InfoGraph** [70] 通过最大化图层面表示和不同尺度的局部子结构之间的互信息来进行图表示学习。

- **GraphCL** [4] 结合四种图形增强技术，即节点丢弃、边缘扰动、子图裁剪和特征屏蔽，进行图形级对比学习。
- **JOAO** [71] 提出了一个为 GCL 自动搜索适当数据增强的框架。
- **AD-GCL** [72] 将对抗学习用于自适应图增强，以去除图样本中的冗余信息。
- **GraphLOG** [73] 利用聚类来构建图形样本的分层原型。他们进一步将每个局部实例与其相应的高级原型进行对比，以进行对比学习。
- **RGCL** [74] 训练一个理由生成器来识别图增强中的因果子图。在对比学习中，每个图的因果子图及其补图都会被利用。
- **BGRL** [62] 通过学习预测目标编码器的输出来训练在线编码器。目标编码器与在线编码器采用相同的架构，并通过指数移动平均法进行更新。在线编码器和目标编码器的输入是两个不同的图增强。

表 10：对来自 GEOM 数据集的 5 万个分子进行预训练以及对四个分子性质预测（回归）数据集和 DTA 数据集进行微调的实验设置。

(a) 节点和边缘特征

类型		范围
节点	特征原子序数	1~ 118
	手性标签 节点	{未指定、四面体 cw、四面体ccw、其他} 0~10
	度 H 的形式电	-5~5
		0~8
	荷数	0~4
	e杂交基数	{SP、SP2、SP3、SP3D、SP3D2}{SP、SP2、SP3、SP3D2}.
	是芳香环	{假, 真}。
		{假, 真}。
边缘	特征键类型{单键	、双键、三键、芳香键}
	键立体	,stereoz, stereoe, stereocis, stereotrans,stereoany} 是共轭的
		true}

(b) 超参数及其搜索空间。我们使用验证集上的性能来调整超参数。**粗体**表示实验中使用的最终值。

编码器	预训练			微调（回归）			微调（DTA）	
	lr	批量大小			lrbatch 大小	时间	lrbatch 大小	epochs
基尼系数	1e-3	1024		100	1e-3{32		, 128, 256} 100	{1e-4
、2e-4}	128	500						
GTS	1e-4	1024300		{1e-4，2e-4，3e-4}	**32100**	**{1e-4，2e-4**	32100	{1e-4，2e-4}
	128	500						

- **GraphMAE** [9] 表明，线性分类器不足以解码节点类型。它采用 GNN 进行解码，并提出了 remask 来分离自动编码器中编码器和解码器的功能。
- **GraphMVP** [45] 使用对比损失和生成损失来连接同一分子的二维视图和三维视图，以便将三维知识注入二维图编码器。
- **S2GAE** [75] 会随机屏蔽图形的部分边缘，并对图形编码器进行预训练，以预测缺失的边缘。
- **GraphMAE2** [76] 采用多视角随机重掩码解码作为 MGM 预训练的正则化。
- **Mole-BERT** [10] 将对比学习目标和掩蔽原子建模目标相结合，用于 MRL。具体来说，他们发现掩码原子预测是一项过于简单的预训练任务。因此，他们采用了由 VQ-VAE [43] 预训练的 GNN 标记器，为掩蔽原子建模生成更复杂的重构目标。

D.2 第 4 节和表 5 中的实验数据

在此，我们详细阐述了在 ZINC15 [42] 的 200 万个分子上进行预训练的实验设置，以及在 MoleculeNet [28] 的八个分类数据集上进行微调的实验设置：BBBP、Tox21、ToxCast、Sider、ClinTox、MUV、HIV 和 Bace。这种设置涵盖了第 4 节和表 5 中的实验。

分子表示法。在 SimSGT 和其他比较方法中，我们沿用了以前的研究成果[12, 4]，使用一组最小的分子特征作为图表示（表 9a）。这些特征明确地描述了分子的二维结构。

超参数。表 9b 总结了超参数。针对不同的图编码器，我们使用了不同的超参数。两个图编码器的架构借鉴了之前的工作：GINE [12] 和 GTS [27]。我们使用 1024 和 2048 的大批次规模来加 速 预训练。在预训练过程中，我们不使用 dropout。在微调过程中，我们在 GINE 层中使用 50%的 dropout，在 transformer 层中使用 30% 的 dropout。MUV 数据集的学习率是其他数据集的 10 倍。按照文献[4, 73]的方法，我们报告了最后一个历元的测试性能。我们

表 11：用于质量管理数据集微调的超参数。

QM 数据集	批量大小	lr
QM7	32	4e-4
QM8	256	1e-3
QM9	256	1e-3

报告了 10 个随机种子的平均性能和标准偏差。基线使用相同设置重现。

线性探测实验。在此，我们将详细说明线性探测实验的设置（图 5b 和图 7）。具体来说，我们将 ZINC15 中的 200 万个分子随机分成训练集（90%）和测试集（10%）。我们在训练集上训练 MGM 模型，并保存编码器的每个历时检查点。线性分类器在编码器冻结的隐藏表示上训练 1000 个历时。我们使用训练集中的 25600 个分子样本训练线性分类器，并在整个测试集中对其进行评估。

- **探测被遮蔽的原子类型（图 5b）。**我们让线性分类器使用被掩蔽原子的隐藏表示来预测被掩蔽原子的类型。在线性探测过程中，我们禁用 remask-v2，以获取被掩蔽原子的隐藏表示。在探测过程中，分子被随机屏蔽 0.35。我们使用准确率（%）作为评估指标。
- **探测 FGs（图 7）。**根据文献[2]，我们使用 RDkit [30]为每个分子提取了 85 种 FG。FG 由 85 维的二进制向量表示，每个维度表示存在某种 FG。然后，我们对冷冻编码器的均值池输出进行多标签线性分类器训练，以预测 FG。在探测过程中，分子不会被遮蔽。我们使用 ROC-AUC (%) 作为评估指标。

D.3 表 6 中的实验

我们介绍了对来自 GEOM [46] 的 5 万个分子进行预训练的实验设置，以及对四个分子性质预测（回归）数据集和两个 DTA 数据集进行微调的实验设置。我们的实验设置与文献 [45] 相同。该设置涵盖了表 6 中的实验。

分子表征。在图自动编码器中，我们使用了 OGB [77] 软件包提供的分子的 9 维节点特征和 3 维边特征，并沿用了 Graph- MVP [45]。表 10a 总结了这些特征。请注意，我们的标记器只使用原子序数作为节点特征，而不使用边缘特征。

超参数表 10b 总结了超参数。我们在微调阶段利用验证性能调整超参数。按照文献[45]，我们报告的是在验证性能选定的历元上的测试性能。在预训练期间，我们不使用辍学。在微调过程中，我们在 GINE 层和变换层分别使用了 50% 和 30% 的 dropout。

为了进行公平比较，我们通过对来自 GEOM 数据集 [46] 的 5 万个分子样本进行预训练，再现了 Mole-BERT [10] 的性能。最初的 Mole-BERT 是在来自 ZINC15 [42] 的 200 万个分子的更大数据集上进行训练的。

D.4 表 7 中的实验

表 11 列出了在 QM 数据集上进行微调的超参数。

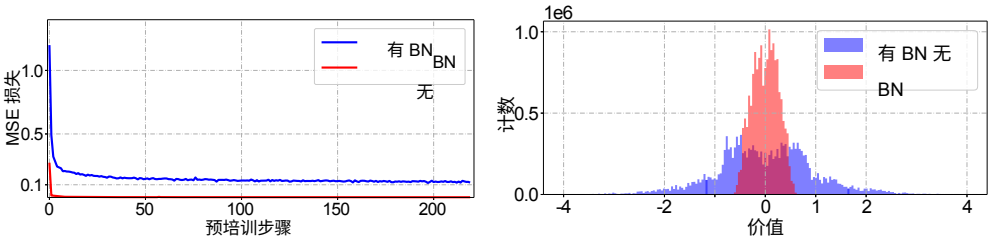
E 更多实验结果

在本节中，我们将提供更多的实验结果。如果没有特别说明，这些实验采用了带有 remask-v2 的 GTS 编码器和 GTS-Small 解码器的自动编码器，以及 GIN 单层 SGT 的标记化器。其他设置见附录 D.2。

边缘特征对基于 GNN 的预训练标记符的影响。我们消除了基于 GNN 的预训练标记器中 " 键类型 "和 " 键方向 " 边缘特征的影响。我们使用 GINE 和

表 12：MoleculeNet 中八个分类数据集的平均迁移学习 ROC-AUC 分数（%）。在标记化器中加入边缘特征会降低性能。

代币转换器	特征	标记符 GNN 的深度 边缘 1	2	3	4	5
预培训，GraphCL	✗	**75.1**	74.5	**74.2**	**74.0**	**74.6**
	✓	74.2	**74.7**	73.7	73.8	73.2
预培训，GraphMAE	✗	**75.1**	**74.9**	**74.9**	**75.4**	**75.2**
	✓	74.6	74.6	74.3	74.6	75.0



(a) 与预训练步骤相关的 MSE 损失曲线。(b) 标记化器输出值的直方图。

图 8：对 ZINC15 的 200 万个分子进行 SimSGT 预训练。

有边缘特征和没有边缘特征的标记符的 GIN。表 12 显示，在基于 GNN 的标记化器中加入边缘特征会对迁移学习性能产生负面影响。因此，我们在实验中排除了基于 GNN 的预训练标记符中的边缘特征。

SGT 中批量归一化层的影响。SGT 中的批量归一化层 [41]（BN）对于避免损耗消失至关重要。图 8a 展示了 "有 BN 层*与*无 BN 层 "的 SGT 比较。如果没有 BN 层，在预训练的几个步骤内，MSE 损失就会下降到 0.01 以下。如此小的损失值会导致严重的模型拟合不足。

如图 8b 所示，不含 BN 的 SGT 的标记值呈尖锐分布：标记值主要分布在零点附近，其标准偏差（std）小于 0.35。这个小的标准偏差问题可能是由 GNN 的平滑效应造成的[78]。表达式神经网络（*即*图自动编码器）可以快速拟合这种尖锐的目标分布，并将损失最小化到可忽略不计的值，从而导致损失消失问题。但是，如果使用 BN 层，就会迫使标记符输出的每个维度的 std 都为 1.00，从而 "分散 "SGT 标记符的分布。这些具有较大 std 的新 SGT 标记更难匹配。它们将 MSE 损失保持在 0.10 ~ 0.15 的合理范围内（图 8a）。

屏蔽率。我们在整个实验过程中采用了随机节点屏蔽 [12]。图 9 显示了 SimSGT 对掩码比率的敏感度。SimSGT 对掩码比率并不敏感，因此在很宽的比率范围（0.25 0.45）内都能产生有竞争力的性能。0.35 的比率实现了最佳性能。该比率远低于图像的比率，在图像中，0.75 的比率可以产生很好的性能[15]。

平衡重建目标的分布。如图 11 所示，常用的 ZINC15 数据集包含 12 种原子类型，其中 95% 的原子分布在前三种原子类型上。这种倾斜分布使得节点级标记重构成为一项简单的

预训练任务[10]。图 10 显示，节点级标记预测的准确率收敛得很快。现有的 SSL 文献[79, 80]指出，这种简单的预训练任务可能会导致次优性能。图 11 显示，单层 SGT 的诱导子图（*即单*跳根子树）的分布比节点分布更均衡。SGT 标记的词汇量也更大：ZINC15 包含 555 种单跳根子树。因此，预测单层 SGT 标记的准确性需要更多的历时才能收敛（图 10）。

子图表示的池化方法。在之前的实验中，我们按照 [4, 12] 中获取图表示的方法，使用均值池法获取基于图案的标记符的子图表示。在此，我们添加了 MGSSL 标记符号生成器使用总和和最大值的结果。

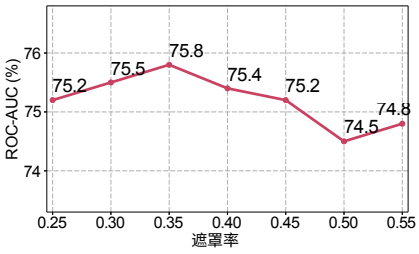


图 9：不同节点屏蔽率的平均 ROC-AUC 分数（%）。在 MoleculeNet 的八个分类数据集上对性能进行了评估。

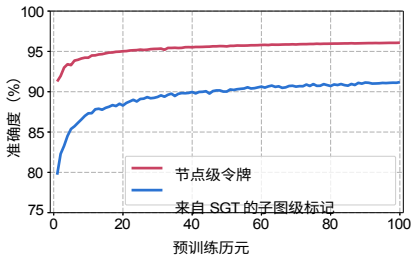


图 10：标记预测准确率。SGT 标记预测是通过计算自动编码器输出与所有 SGT 标记词汇之间的欧氏距离来实现的。

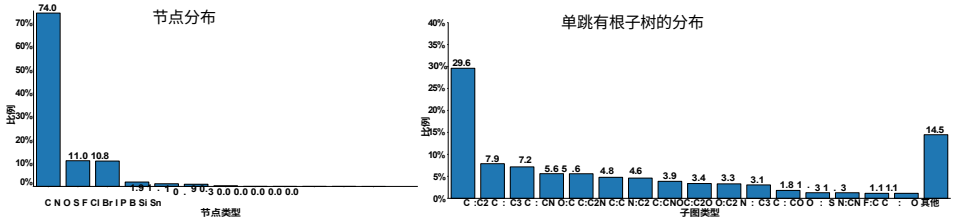


图 11：MGM 中图形片段的分布。统计数据来自 ZINC15 [42] 中的分子。对于子图类型，中心节点和相邻节点之间用冒号": "分隔。例如，C:CN 表示以碳为中心、以碳和氮为邻居的子图。

表 13 列出了平均值池计算的结果。结果表明，均值集合的性能最高，这证明了我们之前的实验是正确的。

第 4 节的全部结果我们在第 4 节提供了全部实验结果。表 14 包含表 3a 和表 4 的全部结果。表 15 包含表 3b 和图 6 的全部结果。

表 13：八个 MoleculeNet 数据集的 ROC-AUC (%) 分数。比较方法使用 GTS 编码器和 GTS-Small 解码器以及 remask-v2 解码。

数据集	BBBP	Tox21	ToxCast	SIDER	ClinTox	MUV	艾滋病毒 BACE	平均值	增益
Motif、MGSSL、平均值	72.5±0.9	77.5±0.4	65.2±0.6	60.7±0.9	85.0±3.5	79.9±1.5	78.0±1.5	83.0±1.0	75.2	5 .3
Motif、MGSSL、Max	71.5±0.9	75.8±1.2	66.2±0.7	60.7±1.3	82.6±2.3	78.9±1.8	76.5±1.4	83.8±1.6	74.5	4 .6
Motif、MGSSL、Sum	71.7±1.4	75.9±0.6	66.1±0.7	60.4±1.4	83.4±1.5	79.5±1.0	76.8±1.2	84.2±1.1	74.8	4 .8

表 14：MoleculeNet 中八个分类数据集的迁移学习 ROC-AUC 分数（%）。

编码器	BACE	解码器重掩码	BBBP	Tox21	ToxCast	SIDER	ClinTox	MUV	艾滋病毒	平均值
GINE	线性	-	72.5±0.8	76.0±0.4	63.7±0.5	60.1±0.7	81.2±2.6	74.2±1.6	78.0±0.8	79.6±1.4	73.2
GINE	GINE-Small	-	70.9±0.6	75.1±0.5	63.5±0.4	61.0±0.4	79.1±2.6	76.0±0.5	76.3±0.5	82.5±0.9	73.0
GINE	GINE-Small	v1	70.2±1.0	76.4±0.6	64.2±0.4	61.9±0.8	80.9±1.8	77.8±1.1	78.1±1.1	83.6±1.1	**74.1**
GTS	线性	-	72.9±1.0	76.6±0.8	63.8±0.8	58.3±1.3	81.9±5.5	78.5±1.5	78.0±2.0	83.1±0.9	74.1
GTS	小号 GTS	-	72.0±0.6	74.7±0.4	63.7±0.4	58.9±0.6	86.0±2.0	78.9±1.2	77.3±0.7	81.0±0.7	74.1
GTS	小号 GTS	v1	71.3±1.0	77.0±1.2	66.2±0.6	60.6±1.4	84.5±3.4	81.5±1.5	77.0±1.6	83.5±1.2	75.2
GTS	小号 GTS	v2	72.2±0.9	76.8±0.9	65.9±0.8	61.7±0.8	85.7±1.8	81.4±1.4	78.0±1.9	84.3±0.6	**75.8**
GTS	GTS-Tiny	v2	71.9±1.2	77.2±1.1	65.6±0.5	61.7±1.4	82.9±2.4	79.6±1.4	76.8±1.3	82.1±1.5	74.7
GTS	GTS	v2	70.7±1.2	76.4±0.9	66.1±0.4	60.3±0.9	84.7±4.6	79.6±0.7	76.8±1.9	84.5±0.8	74.9

表 15：MoleculeNet 中八个分类数据集的迁移学习 ROC-AUC 分数（%）。

代币转换器	BBBP	Tox21	ToxCast	SIDER	ClinTox	MUV	艾滋病毒 BACE	平均值
节点	70.3±0.9	76.4±1.0	65.7±0.7	61.7±0.9	81.9±3.5	79.8±0.7	77.4±1.8	84.6±1.1	74.7
Motif、MGSSL	72.5±0.9	77.5±0.4	65.2±0.6	60.7±0.9	85.0±3.5	79.9±1.5	78.0±1.5	83.0±1.0	75.2
Motif, RelMole	71.4±1.3	77.1±0.4	66.3±0.6	58.9±1.2	80.7±2.7	79.2±1.4	78.0±1.0	83.6±1.0	74.4
预训练，GraphCL，1 层 GIN	72.2±1.4	76.8±0.4	66.0±0.8	60.8±1.2	81.4±2.5	81.1±1.5	78.4±1.3	83.8±0.9	75.1
预训练、GraphCL、2 层 GIN	70.8±0.6	76.7±0.8	66.3±0.4	60.6±1.2	84.3±3.3	77.2±1.4	76.3±2.1	83.8±1.2	74.5
预训练、GraphCL、3 层 GIN	70.6±0.8	77.1±0.6	65.4±0.5	59.3±1.4	81.2±3.8	79.4±2.3	76.4±1.9	84.2±1.1	74.2
预训练、GraphCL、4 层 GIN	72.1±0.9	76.9±0.6	65.7±0.7	59.6±0.9	77.6±3.5	81.7±1.2	77.6±2.2	81.1±1.1	74.0
预训练、GraphCL、5 层 GIN	71.4±0.7	76.9±0.7	66.5±0.7	60.0±1.2	80.8±2.3	81.0±1.1	77.7±1.2	82.5±1.5	74.6
预训练、VQ-VAE、1 层 GIN	72.2±0.9	77.0±0.6	66.5±0.6	61.3±1.8	82.8±3.7	79.1±2.0	77.4±1.5	84.2±0.8	75.1
预训练、VQ-VAE、2 层 GIN	71.5±0.8	76.6±0.6	65.9±0.7	60.3±0.7	82.1±2.0	81.5±2.4	77.2±1.9	84.3±1.1	74.9
预训练、VQ-VAE、3 层 GIN	71.9±0.9	76.7±0.9	65.8±0.8	61.2±1.8	79.5±2.5	80.0±0.9	78.0±1.3	81.7±0.9	74.4
预训练、VQ-VAE、4 层 GIN	72.5±1.0	77.0±0.6	66.3±0.3	61.7±1.5	86.7±2.2	80.3±1.6	77.6±1.3	82.7±0.9	75.6
预训练、VQ-VAE、5 层 GIN	72.0±0.9	76.8±0.6	65.6±0.6	61.5±1.1	84.3±1.3	80.6±1.0	78.1±1.4	81.9±0.8	75.1
预训练，GraphMAE，1 层 GIN	72.0±1.1		66.3±0.3	60.9±1.5	83.7±2.7	79.9±1.4	76.3±2.2		75.1

	77.3±0.6				84.0±1.6		
预训练、GraphMAE、2 层 GIN	71.9±0.8	77.6±0.6	66.0±0.5	60.8±1.8	81.9±3.9	79.0±1.6	76.5±2.4	74.9
						85.3±0.7		
预训练、GraphMAE、3 层 GIN	71.4±0.7	77.6±0.8	65.8±0.3	61.1±1.7	82.2±2.9	79.2±1.5	77.4±2.1	74.9
						84.3±0.9		
预训练、GraphMAE、4 层 GIN	72.3±0.7	76.6±0.7	66.1±0.8	62.0±1.2	83.3±2.1	80.1±2.2	77.9±1.7	75.4
						85.0±0.9		
预训练、GraphMAE、5 层 GIN	72.6±0.6	76.4±0.5	65.7±0.6	62.4±1.3	84.0±2.8	80.0±1.3	78.7±1.3	75.2
						81.5±1.3		
SGT, 1 层 GIN	72.2±0.9		65.9±0.8	61.7±0.8	85.7±1.8	81.4±1.4	78.0±1.9	75.8
	76.8±0.9					84.3±0.6		
中士，2 层 GIN	71.3±0.7	77.0±0.9	66.2±0.8	61.5±0.8	84.9±2.0	80.7±2.0	78.1±1.1	75.5
						84.5±0.8		
中士，3 层 GIN	71.7±0.8	77.6±0.8	66.2±0.6	61.2±2.4	85.8±2.6	80.4±1.1	77.7±2.1	75.6
						84.1±1.4		
中士，4 层 GIN	72.0±0.9	77.1±1.2	65.4±1.0	61.7±1.5	83.8±2.2	80.1±1.9	77.5±1.2	75.3
						84.7±0.9		
中士，5层 GIN	70.7±0.7	77.0±0.7	65.9±0.8	61.1±1.6	83.1±1.6	79.9±1.5	77.6±1.6	74.9
						83.9±1.4		
SGT，1 层 GCN	71.9±0.9	77.8±1.0	66.5±0.7	62.0±0.8	85.2±1.6	79.2±1.4	77.9±2.3	75.6
						84.4±2.1		
SGT，3 层 GCN	71.1±1.0		66.3±0.4	61.6±1.1	84.4±2.5	78.7±1.1	77.7±2.4	75.3
	77.4±0.8					85.1±1.3		
SGT, 5 层 GCN	71.0±1.0		65.6±0.3	61.9±1.3	83.7±2.1	78.3±2.0	76.7±1.2	74.8
	76.5±0.6					84.9±1.0		
SGT，1 层 GraphSAGE	72.5±0.7	77.1±0.8	66.7±0.5	61.3±1.0	86.2±1.8	80.7±1.5	76.6±1.7	75.6
						83.5±1.0		
SGT，3 层 GraphSAGE	70.3±1.1		65.4±0.6	60.3±1.0	87.4±3.7	79.2±1.9	77.6±2.1	75.4
	77.9±0.7					84.9±0.5		
SGT，5 层 GraphSAGE	71.6±1.0		66.4±0.7	60.8±1.4	85.6±2.4	78.6±1.3	77.0±1.6	75.1
	76.5±0.7					84.4±0.9		