利用图支撑的预训练与提示增强低资源文本分类

Zhihao Wen 新加坡管理大学 新加坡
zhwen.2019@smu.edu.sg

Yuan Fang 新加坡管理大学 新加坡 yfang@smu.edu.sg

arXiv:2305.03324v1 [cs.IR] 5 May 2023

摘要

文本分类是信息检索中的一项基础任务，在现实世界中有许多应用，例如预测在线文章的主题，以及电子商务产品描述的类别。然而，低资源文本分类缺少或只有少量标注样本，这给监督学习带来了严峻挑战。同时，许多文本数据天然依托于网络结构，例如在线文章的超链接／引文网络，以及电子商务产品的用户—商品购买网络。这些图结构捕捉了丰富的语义关系，有望增强低资源文本分类。本文提出一种名为图支撑的预训练与提示（G2P2）的新模型，通过双管齐下的方法解决低资源文本分类问题。在预训练阶段，我们提出三种基于图交互的对比策略，共同对图文模型进行预训练；在下游分类阶段，我们探索如何通过提示应用联合预训练的模型，以实现低资源分类。在四个真实世界数据集上进行的大量实验，证明了 G2P2 在零样本和少样本低资源文本分类任务中的优势。

CCS 概念

• **信息系统** → **内容分析与特征选择**；**聚类与分类**。

关键词

文本分类、图神经网络、低资源学习、预训练、提示调优

ACM 参考格式：
Zhihao Wen 和 Yuan Fang。2023。利用图支撑预训练和提示增强低资源文本分类。载于第*46*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集（*SIGIR '23*），*2023*年*7*月*23*日至*27*日，中国台湾台北。ACM，纽约，美国纽约州，11页。https://doi.org/10.1145/3539618.3591641

允许免费制作本作品全部或部分内容的数字版或纸质副本，用于个人或课堂用途，但不得出于盈利或商业目的复制或分发，且副本必须附有本声明及首页上的完整引文。对于本作品中由作者以外的其他人拥有版权的组件，必须尊重其版权。注明出处后可以摘录。如需以其他方式复制、再版、发布到服务器或分发至列表，须事先获得具体许可和/或支付费用。请向 permissions@acm.org 申请许可。*SIGIR '23, 2023*年*7*月*23*日至*27*日，中国台湾台北 © 2023 版权所有者/作者所有。出版权已授权给 ACM。ACM ISBN 978-1-4503-9408-6/23/07…$15.00
https://doi.org/10.1145/3539618.3591641

1 引言

文本分类是信息检索中的一项基础研究问题，在许多重要应用中发挥着作用。例如，预测在线文章的主题可以帮助读者在网站或门户网站 [29], 中轻松搜索和浏览；对电子商务商品描述进行类别分类，则能帮助企业高效地整理库存，并改善用户的搜索体验 [60]。过去十年间，监督式深度学习取得了显著进展，在文本分类方面尤为成功，尤其是在拥有大规模、高质量标注数据的情况下。然而，数据标注往往成本高昂且耗时，因此，在没有或只有少量标注样本可用的情况下，低资源分类成为一种颇具吸引力的替代方案。

为了解决低资源文本分类问题，一种方法是利用预训练语言模型（PLM） [16, 37], ，其中许多模型基于 Transformer架构 [48] ，因为该架构具有强大的文本编码能力。PLM 可以通过将模型参数微调到任务特定目标上，使其适配不同任务。尽管“预训练、微调”范式所需的标注数据少于传统监督学习，但它存在两个缺点。首先，当前最先进的 PLM 通常具有庞大的模型规模，例如，GPT-3 拥有 1750 亿个参数 [3], ，使得微调成本高昂得令人难以承受 [19]。其次，由于预训练目标与微调目标之间存在差距，微调仍然需要相当数量的标注数据，因此难以应对包括零样本和少样本分类在内的低资源场景。为解决预训练与微调之间的问题，提示 [3] 应运而生。该方法使用自然语言指令或“提示”来提供下游任务的线索，同时冻结大型 PLM 的参数。换言之，对于新任务，完全不需要微调或额外训练。然而，与微调相比，离散自然语言提示可能难以设计，并且可能导致性能欠佳 [18]。最近，提示调优 [18, 24]将连续提示构建为可学习嵌入，在任务适配过程中对其进行优化，而无需更新 PLM。

与此同时，文本数据通常依托于网络结构，例如在线文章的超链接网络或引文网络，以及电子商务中的用户-商品交互图。这些图结构揭示了文章或商品之间的宝贵关系，可用于增强低资源文本分类。现有的预训练语言模型和提示并未利用这些关系，而图神经网络 (GNN) [58] 旨在基于消息传递架构从图结构中学习。然而，传统的图神经网络端到端训练严重依赖大量任务特定标签，因此催生了自监督图神经网络 [57] ，它们在无标签图 [14, 15, 50]上采用精心设计的前置任务。遗憾的是，对文本特征的处理在

图神经网络仍处于初级阶段。通常，简单的词袋表示 [62]或浅层词嵌入向量的聚合[30] 会作为初始节点特征输入图神经网络，并沿图结构进一步传播。因此，图神经网络对文本的建模较为粗粒度，无法充分捕捉文本内部细微的语义差异与相似性。

挑战与当前工作。为克服现有基于文本和图的方法的局限性，我们必须解决以下两个尚待解答的问题。

首先，我们如何在利用图结构信息的同时，联合捕获细粒度的文本语义？一种简单的方法是使用语言模型从原始文本中生成特征作为输入，然后训练图神经网络。然而，这种方式只是松散地耦合文本和图，缺乏明确的配对来使二者互为补充。本文提出图支撑的对比预训练，基于三种图交互——文本-节点、文本-摘要和节点摘要交互——最大化文本表示与图表示之间的对齐。

其次，在拥有联合预训练的图文模型时，我们如何增强低资源文本分类？我们提出一种新颖的方法：对联合预训练的图文模型进行“提示”，而非微调。这样，我们便能利用预训练模型中最相关的结构和语义信息，使整个过程更适用于低资源场景。具体而言，我们在零样本分类中使用手工设计的离散提示，在少样本场景中则使用基于自动提示调优的连续提示。由于涉及的参数显著更少，提示调优比微调预训练模型更节省标注和计算资源。此外，我们还提出一种用于提示调优的基于上下文的初始化方法，该方法考虑文本之间的图结构，以提供信息更丰富的起点。

贡献。 总而言之，本研究的贡献如下。(1) 据我们所知，这是首次尝试联合预训练文本编码器和图编码器，以用于低资源文本分类。(2) 我们提出了一种名为图支撑的预训练与提示（G2P2）的新模型，其中包含三种基于图交互的预训练对比策略，以及一种面向下游任务中联合预训练图文模型的提示方法。(3) 我们在四个真实世界数据集上进行了大量实验，证明了 G2P2 在零样本和少样本文本分类中的优势。

2 相关工作

图神经网络。 受计算机视觉中卷积网络成功的启发，图神经网络应运而生，用于处理非欧几里得关系数据 [58], , 从早期的半监督模型（如图卷积网络 [17], 、GAT [49] 和GIN [61], ）到近期的自监督预训练范式 [14, 15, 27, 50]。除了在图任务中广泛取得成功外，它们还通过知识图谱 [4] 、异构图 [20],或多模态学习 [26]，被用于提升基于文本的任务表现。然而，这些方法要么对文本进行粗粒度处理，要么将图编码器与文本编码器分离，未能充分利用二者之间的内在联系。尽管近期一种名为 GLEM 的方法 [68] 通过融合语言模型和图神经网络，整合了文本与图结构信息，但它并非为低资源学习而设计。

语言预训练与提示。 预训练语言模型 [10] 已成为自然语言处理（NLP）领域最受欢迎的骨干模型。早期的预训练语言模型，如 GPT [37], 、BERT[16], 、XLNet [63] 和 RoBERTa [25] ，模型规模仍在可承受范围内；而近期推出的模型，如 T5 [38]和 GPT-3 [3]，则拥有数十亿个参数，规模庞大。为了避免在这些大型模型上进行微调的高昂成本，提示 [22] 开始受到学界越来越多的关注。提示是一种用于填充任务输入的特殊模板，旨在从预训练语言模型中提取有用知识，以灵活适配下游任务。在 GPT-3 成功的推动下，出现了许多提示方法，包括离散自然语言提示 [8, 40, 42] 和连续提示[18, 19, 24, 35, 69] 。提示的优势已在广泛的自然语言处理应用中得到验证，包括文本分类 [11, 13, 31, 45, 66], 、机器翻译 [46] 和关系抽取 [6, 39]。近期，提示也被应用于图神经网络，以进行节点分类 [44]。

零样本或少样本范式。 广义而言，我们的设定也与其他学习范式有关。例如，在半监督学习中 [5, 32, 59], , 每个类别可能只有少量样本，但训练期间必须见过所有类别，因此无法在测试时处理任何新类别。元学习 [1, 2, 7,12, 52, 54, 55, 65, 70] 是另一种支持少样本学习的流行范式。然而，仍然需要在所谓的“元训练”阶段使用大规模标注数据，以支持在“元测试”阶段对新类别进行少样本学习。相比之下，我们只需使用无标签数据进行预训练，无需消耗大规模标注数据的元训练阶段。此外，也有研究采用对比预训练策略，联合考虑图像和文本数据，以实现零样本或少样本分类 [36]。在我们的工作中，图数据与图像有显著区别，它能提供文本之间多种类型的交互信息。关于图上的零样本节点分类，也已有相关研究 [53]。该方法高度依赖维基百科页面或其他辅助信息，以生成类别原型嵌入。然而，寻找并整理合适的辅助信息非常耗费人力，尤其是在类别数量庞大和/或新类别频繁出现时。

3 提出的方法

本节将介绍我们用于低资源文本分类的方法 G2P2。我们首先介绍一些预备知识和总体概述，然后详细说明所提出的方法。

3.1 预备知识

图结构关联文本语料库。 考虑一组文档 D，它们以图 G = (D, E,X) 为基础构建，其中每个文档 $d_i \in$ D 都是图中的一个节点 v_i。文档通过 E 中的边相连，这些边根据具体应用构建（例如，如果每个文档代表一篇文章，那么边可以是文章之间的引文）。每个节点 v_i 还关联一个特征向量 $\mathbf{x}_i$，由输入特征矩阵 $\mathbf{X}$给出。最后，每个文档/节点[1] 都有一个类别标签（例如，文章的主题）。

[1]鉴于在我们的语境中二者是一一对应的，我们将交替使用“节点”和“文档”这两个术语。

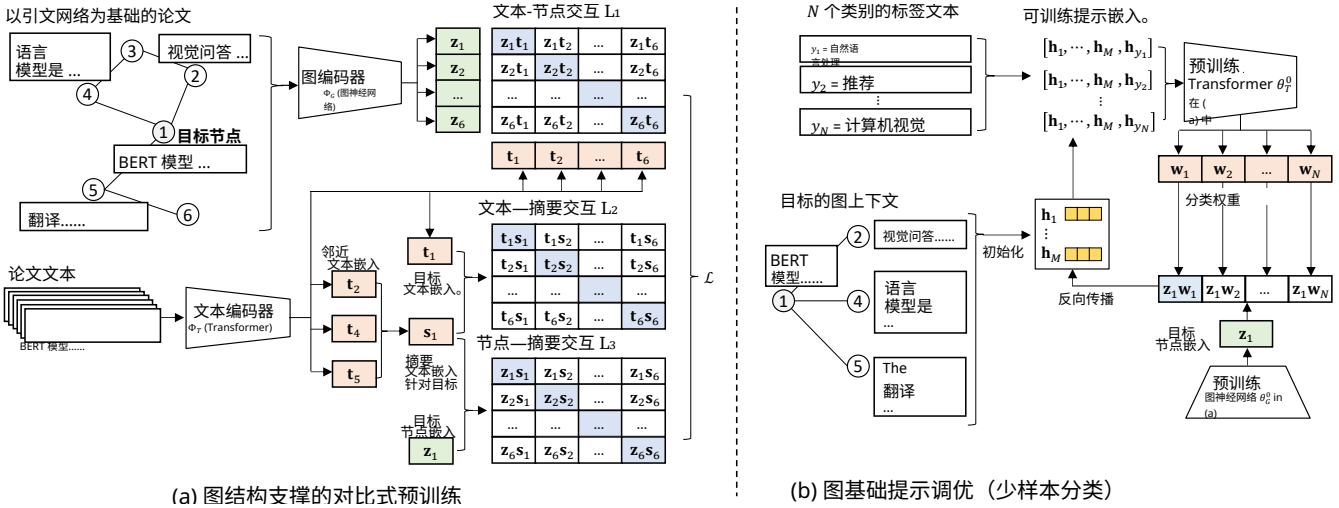


图 1：G2P2 的总体框架。(a) 在预训练期间，通过三种对比策略联合训练文本编码器和图编码器。(b) 在测试期间，执行提示辅助的零样本或少样本分类（图中仅展示了少样本分类的提示调优，而零样本推理采用简化方案）。

低资源分类。 低资源任务由一个支持集 S 和一个查询集 Q 组成。支持集 S 包含 N个类别，每个类别有 K 个带标签示例，其中 K 是一个较小的数值（例如，1 或 5），这种分类方式称为 N-way K-shot 分类。查询集 Q 包含一个或多个属于支持集中 N个类别的无标签实例。我们的目标是根据支持集中的带标签示例，对查询集中的实例进行分类。与同时包含训练任务和测试任务的情景式少样本元学习 [7] 不同，我们只有测试任务；在训练阶段，我们仅在无标签数据上进行自监督预训练。作为一种特殊情况，具有 $K = 0$ 的任务称为零样本分类，这意味着完全没有带标签示例，我们只能依赖类别元数据（例如，类别标签文本）。

3.2 G2P2 概述

如图 1 所示，我们的模型由两个阶段组成：(a) 基于图的对比预训练，以及 (b) 面向低资源分类的基于图的提示调优。

在预训练期间，我们通过以自监督方式联合训练文本编码器和图编码器，学习一个双模态嵌入空间，因为文档也作为节点存在于图中。具体而言，我们使用基于Transformer的文本编码器和基于GNN的图编码器。Transformer 将每个节点上的文本（即，文档）作为输入，并输出节点 v_i 的文本嵌入向量 $\mathbf{t}_i$。另一方面，GNN 将图和节点特征作为输入，并生成节点 v_i 的节点嵌入向量 $\mathbf{z}_i$。随后，在双模态嵌入空间中，我们通过三种基于图中不同类型交互的对比策略，对齐相同或相关节点上的文本和图表示。

在下游测试中，我们对联合预训练的图文模型使用提示，以进行零样本或少样本分类。对于零样本分类，我们将手工设计的离散提示与标签文本结合使用。对于少样本分类，我们使用连续提示来填充标签文本。具体而言，在提示调优中，我们根据图上下文初始化连续提示嵌入。

3.3 图支撑的对比预训练

图支撑预训练基于底层图上的三种交互，通过联合训练文本编码器和图编码器来学习双模态嵌入空间。

双编码器。 文本编码器是一个 Transformer [48],，记为 Φ_T。给定文档 d_i，文本编码器[2] 会输出d维的嵌入向量，记为 $d_i\mathbf{t}_i \in \mathbb{R}^d$：

$$\mathbf{t}_i = \Phi_T(d_i; \theta_T), \tag{1}$$

其中，θ_T 表示 Transformer 的参数集。相应地，令 $\mathbf{T} \in \mathbb{R}^{|\mathcal{D}| \times d}$ 表示所有文档的文本嵌入矩阵。

同时，文档 d_i 也是图中的一个节点 v_i。我们选择一种经典的 GNN，即图卷积网络（ GCN ）[17]，作为图编码器，记为 Φ_Z。对于给定节点 v_i，它同样会输出一个嵌入向量 $\mathbf{z}_i \in \mathbb{R}^d$：

$$\mathbf{z}_i = \Phi_Z(v_i; \theta_G), \tag{2}$$

其中，θ_G 表示 GCN 的参数集。同样地，令$\mathbf{Z} \in \mathbb{R}^{|\mathcal{D}| \times d}$ 表示所有节点的图嵌入矩阵。

文本-节点交互。 我们的图关联文本天然地意味着节点与文本之间存在双射，其中每个文档 d_i 都对应图中的节点 v_i。受图像及其图像说明文本 [36] 配对方式以及内容与节点序列 [21]，映射方式的启发，我们设计了一种预训练策略，用于预测哪篇文本文档与图中的哪个节点相匹配。

具体而言，给定 n个文档及其对应的 n个节点，共有 n^2种可能的文档-节点对 $\{(d_i, v_j) \mid i, j = 1, \ldots, n\}$。

[2]从技术上讲，文本编码器的输入是一系列连续嵌入；文档中的词元首先被转换为词嵌入。

其中，只有 n 个满足 $i = j$ 的配对是真正匹配的，其余 $n^2 - n$ 个配对均为错误匹配。作为我们的第一种对比策略，我们利用图中文本与节点之间的一一对应关系，最大化 n 匹配对的余弦相似度，同时最小化 n^2 非匹配对的余弦相似度。为计算这些配对的余弦相似度，我们首先对嵌入矩阵 $\mathbf{T}$ 和 $\mathbf{Z}$ 按行进行 L2归一化，分别得到 $\tilde{\mathbf{T}}$ 和 $\tilde{\mathbf{Z}}$。然后，我们计算节点-文本相似度矩阵 $\mathbf{\Lambda}_1 \in \mathbb{R}^{n \times n}$ 以获取成对余弦相似度，具体如下。

$$\mathbf{\Lambda}_1 = \left(\tilde{\mathbf{Z}}\tilde{\mathbf{T}}^\top\right) \cdot \exp(\tau), \tag{3}$$

其中，$\tau \in \mathbb{R}$ 是用于缩放相似度值的可训练温度参数 [36]。

备注。尽管 $\mathbf{\Lambda}_1 \in \mathbb{R}^{n \times n}$ 是一个稠密矩阵，但在实际实现中会按批次级构建。也就是说，n 并非文档总数，而是相对较小的批次大小，因此开销可以忽略不计。$\mathbf{\Lambda}_2$ 和 $\mathbf{\Lambda}_3$ 稍后也会采用相同的处理方式进行介绍。□

为了基于文本与节点的双射交互构建对比损失，我们对多类*N*对损失 [43, 67], 进行调整，同时考虑相对于行索引或列索引的逐行和逐列交叉熵损失。例如，i行的 $\mathbf{\Lambda}_1$ 表示节点 v_i 与每篇文档之间的相似度分数，其中行索引 i 指示与 v_i匹配的真实文档 d_i。

$$\mathcal{L}_1 = \frac{1}{2}\left(\mathrm{CE}(\mathbf{\Lambda}_1, \mathbf{y}) + \mathrm{CE}(\mathbf{\Lambda}_1^\top, \mathbf{y})\right), \tag{4}$$

其中，$\mathbf{y} = (1, 2, \ldots, n)^\top$ 是用于对比训练的标签向量，而 CE 表示按行应用于输入矩阵 $\mathbf{\Lambda}_1$ 或 $\mathbf{\Lambda}_1^\top$的交叉熵损失。

文本-摘要交互。除了双射的文本-节点交互外，我们还进一步利用图上的高阶交互。具体而言，每个文档都具有由图拓扑定义的一组邻近文档。鉴于这些文档之间存在语义相关性，可以将邻近文档视为目标文档的摘要。例如，在电子商务网络中，用户购买的产品自然可以描绘该用户的摘要，反之亦然。在不失一般性的情况下，我们采用简单的均值池化来生成摘要嵌入 $\mathbf{s}_i \in \mathbb{R}^d$，如下所示。

$$\mathbf{s}_i = \frac{1}{|\mathcal{N}_i|}\sum_{j \in \mathcal{N}_i} \mathbf{t}_j. \tag{5}$$

为提高效率，我们仅采样固定数量的邻近文档来生成摘要。接着，令$\mathbf{S} \in \mathbb{R}^{n \times d}$ 表示所有文档的摘要文本嵌入矩阵。

因此，作为第二种对比策略，我们旨在根据从图邻域中获得的文本-摘要交互，对齐每个文档的文本嵌入及其对应摘要文本的嵌入。换言之，我们最大化 n 匹配的文档与其基于邻域的摘要之间的余弦相似度，同时最小化 $n^2 - n$ 不匹配项之间的余弦相似度。具体而言，我们首先按照公式（3）构建文本-摘要相似度矩阵 $\mathbf{\Lambda}_2 \in \mathbb{R}^{n \times n}$:

$$\mathbf{\Lambda}_2 = \left(\tilde{\mathbf{T}}\tilde{\mathbf{S}}^\top\right) \cdot \exp(\tau). \tag{6}$$

算法1 G2P2预训练流程

输入： 图结构关联文本语料库 G = (D, E, X)。
输出： 文本编码器 θ_T^0 和图编码器 θ_G^0 的预训练权重。

```
θ_T^0, θ_G^0 ← 参数初始化;
当 尚未收敛时 执行
    从 D 中采样文档批次;
    对于每个批次执行
        对于 批次中的每个节点 v_i/文档 d_i 执行
            计算 d_i的文本嵌入 t_i;                ▷ 式（1）
            计算 v_i的节点嵌入 z_i;                ▷ Eq. (2)
            计算 v_i的摘要嵌入s_i;                 ▷公式(5)
        结束循环
        计算 S imil性矩阵 ces Λ1, Λ2, Λ3;          ▷ E公式 (3)、(6)、(8)
        计算对比损失 L1, L2, L3;                   ▷ 公式 (4)、(7)、(9)
        更新总体损失 L;                            ▷ 公式 (10)
        θ_T^0, θ_G^0 ← 通过反向传播更新
    结束循环
结束循环
return θ_T^0, θ_G^0.
```

随后，我们按照公式（4）应用相同的对比损失，如下所示。

$$\mathcal{L}_2 = \frac{1}{2}\left(\mathrm{CE}(\mathbf{\Lambda}_2, \mathbf{y}) + \mathrm{CE}(\mathbf{\Lambda}_2^\top, \mathbf{y})\right), \tag{7}$$

节点-摘要交互。文档 d_i 的基于邻域的摘要也充当节点 v_i的语义描述。与文本-摘要交互类似，作为我们的第三种对比策略，我们旨在对齐节点嵌入及其基于邻域的摘要文本嵌入。接下来，我们同样计算节点-摘要相似度矩阵 $\mathbf{\Lambda}_3 \in \mathbb{R}^{n \times n}$，并构建相应的对比损失 L_3。

$$\mathbf{\Lambda}_3 = \left(\tilde{\mathbf{Z}}\tilde{\mathbf{S}}^\top\right) \cdot \exp(\tau), \tag{8}$$

$$\mathcal{L}_3 = \frac{1}{2}\left(\mathrm{CE}(\mathbf{\Lambda}_3, \mathbf{y}) + \mathrm{CE}(\mathbf{\Lambda}_3^\top, \mathbf{y})\right). \tag{9}$$

总体预训练总体预训练总体预训练目标。最后，我们基于文本-总体预训练目标。最后，我们整合基总体预训练目标。最后，我们基于文本-节点、文本-摘要和节点-摘要之间的交互。将这三种对比损失整合起来。总体预训练目标。最后，我们整合基于文本-节点、文本-摘要以及节点-摘要交互的三种对比损失。由双编码器的参数组成。我们得到一个预训练模型 θ^0 =({v总体预训练目标。最后，我们基于文本-节点、文本-摘要和节点-摘要之间的交互。将这三种对比损失整合起来。由此，我们得到一个预训练模型 $\theta^0 = (\theta_T^0, \theta_G^0)$。该模型由双编码器的参数组成。其定义如下。总体预训练目标。最后，我们整合基于文本-节点、文本-摘要以及节点-摘要交互的三种对比损失。由双编码器的参数组成。我们得到一个预训练模型 $\theta^0 = (\theta_T^0, \theta_G^0)$。其表示为总体预训练目标。最后，我们基于文本-节点、文本-摘要和节点-摘要之间的交互，将这三种对比损失整合起来。由此，我们得到一个预训练模型 $\theta^0 = (\theta_T^0, \theta_G^0)$。该模型由双编码器的参数组成，其定义如下。总体预训练目标。最后，我们整合基于文本-节点、文本-摘要以及节点-摘要交互的三种对比损失。由双编码器的参数组成。我们得到一个预训练模型 $\theta^0 = (\theta_T^0, \theta_G^0)$。其表示为总体预训练目标。最后，我们基于文本-节点、文本-摘要和节点-摘要之间的交互。将这三种对比损失整合起来。由此，我们得到一个预训练模型 $\theta^0 = (\theta_T^0, \theta_G^0)$。该模型由双编码器的参数组成，其定义如下。

$$\theta^0 = \arg\min_{\theta_T, \theta_G} \mathcal{L}_1 + \lambda(\mathcal{L}_2 + \mathcal{L}_3), \tag{10}$$

其中，$\lambda \in \mathbb{R}^+$ 是一个超参数，用于平衡基于摘要的交互所产生的贡献。

预训练流程概述于算法1，其每轮的复杂度如下。设 $|\mathbb{D}|$ 为文档数量，η 为生成式（5）中摘要嵌入时采样的邻居数量，β 为批次大小。首先，每轮生成三种嵌入（第5–8行）的开销为 $O(|\mathbb{D}|\eta)$，因为计算摘要嵌入需要遍历 η 个邻居。其次，每个批次计算三个相似度矩阵的开销为 $O(\beta^2)$，考虑到每轮有 $|\mathcal{D}_\beta|$ 个批次，每轮的总开销为 $O\left(|\mathcal{D}_\beta|\beta^2\right) = O(|\mathbb{D}|\beta)$。因此，总体复杂度为 $O(|\mathbb{D}|(\eta + \beta))$，关于文档数量呈线性增长，因为 η 和 β 都是较小的常数。在我们的实现中，设定 $\eta = 3$ 和 $\beta = 64$。

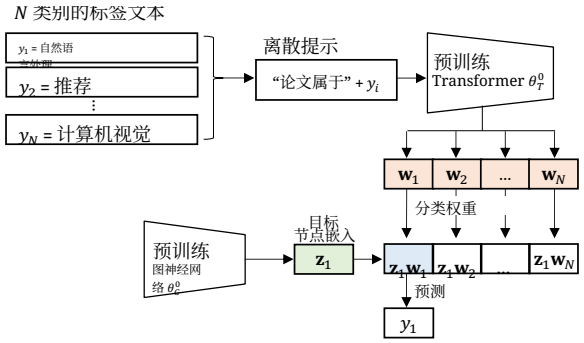


图 2：零样本分类示意图。预训练模型 θ_G^0 和 θ_T^0 取自图 1(a)。

3.4 联合图-文本模型的提示

对图文模型进行预训练后，将其应用于低资源分类并非易事。为了缩小预训练与下游任务之间的差距，传统的“预训练、微调”范式通常会为下游任务引入一个新的投影头，并与整个预训练模型一同进行微调。然而，在低资源场景下，更新包含大量参数的整个模型既无效也低效。近年来，提示已成为自然语言处理中微调的一种强大替代方案，无需更新庞大的预训练语言模型 [22]。不过，对于结构信息与文本信息经过联合预训练的图文模型，提示方法尚未得到探索。下面，我们将详细介绍用于零样本和少样本分类的提示策略。

零样本分类。 在零样本设置下，我们只能使用手工设计的离散提示，因为零样本任务中缺少标注数据，无法支持可学习的提示。

在 N 路零样本分类中，对于 N 个类别，我们预测与给定节点相似度最高的类别。如图 2 所示，文本编码器可以根据类别标签文本 [51], 生成分类权重，而无需为该分类任务提供任何标注样本。具体而言，类别 $y \in \{1、2、\ldots\ldots、N\}$ 的权重向量 $\mathbf{w}_y$ 是预训练文本编码器的输出，即

$$\mathbf{w}_y = \phi_T(\text{“prompt [CLASS]”}; \theta_T^0). \tag{11}$$

这里的“prompt [CLASS]”是一个提示模板，其中 [CLASS] 指目标类别的标签文本 y （例如，“NLP”表示论文领域分类），而 prompt 是由人工设计的自然语言词元序列，用于表明标签文本的相关性（例如，“NLP 论文”有助于聚焦论文的主题）。在最简单的情况下，“prompt”可以是空字符串，因此我们只依赖标签文本。根据节点表示 $\mathbf{z}_i$ 得到的类别分布预测为

$$p(y \mid \mathbf{z}_i) = \frac{\exp\left(\langle \mathbf{z}_i, \mathbf{w}_y \rangle\right)}{\sum_{y=1}^{N} \exp\left(\langle \mathbf{z}_i, \mathbf{w}_y \rangle\right)}, \tag{12}$$

其中，$\cdot,\cdot$ 表示余弦相似度。

少样本分类。离散提示的问题在于难以优化，因为预训练语言模型本身就……

连续的。用可学习的连续提示替代离散自然语言提示后，在有部分标注数据可用的情况下，提示调优 [18, 23, 24] 可以自动优化提示。因此，在少样本设置下，我们探索使用提示调优，从联合预训练的图文模型中引导提取相关的结构信息和语义信息。

具体而言，我们不使用离散词元序列，而是采用连续嵌入序列 $[\mathbf{h}_1, \cdots, \mathbf{h}_M, \mathbf{h}_{\text{CLASS}}]$ 作为提示，其中 M 是一个超参数，表示上下文词元的数量；每个 $\mathbf{h}_m$ $(m \le M)$ 都是可训练向量，而 $\mathbf{h}_{\text{CLASS}}$ 是目标类别标签的词嵌入序列。将连续提示作为文本编码器的输入，为每个类别生成分类权重 y:

$$\mathbf{w}_y = \phi_T([\mathbf{h}_1, \cdots, \mathbf{h}_M, \mathbf{h}_{\text{CLASS}}]; \theta_T^0), \tag{13}$$

其中，每个 $\mathbf{h}_m$ $(m \le M)$ 的维度都与文本编码器的输入词嵌入维度相同。

使用公式（12）中的同一 softmax 层，我们进一步利用少样本任务的带标签支持集，通过最小化交叉熵损失来更新连续提示嵌入，同时冻结双编码器的参数。由于提示中可学习参数的数量较少，这一提示调优过程在数据和计算方面都很高效。

此外，现有的提示调优方法要么随机初始化提示嵌入 [18, 23]，要么使用手工设计的离散提示的词嵌入 [71]。随机初始化缺乏信息，且更容易陷入局部最优；而要挑选合适的离散提示用于初始化也并非易事。因此，我们利用图结构来初始化提示嵌入。

具体而言，给定一个节点 v_i，我们将其图上下文定义为其邻居集合 $\{v_j \mid j \in \mathrm{N}_i\}$。由于底层的语义相关性，少样本示例的图上下文包含关于任务的丰富信号，可用于改进初始化。对于任务支持集中的每个文档/节点 v_i，我们从其图上下文中采样 η 个节点。对于节点 v_i 自身以及采样到的每个上下文节点，我们将其对应的文档截断至 M 个词，并将其转换为由 M 个词嵌入向量组成的序列，其中每个向量的维度都与连续提示中的向量 $\mathbf{h}_m$ $(m \le M)$ 相同。因此，对于每个支持节点，我们会得到 $\eta+1$ 个这样的序列；在一个 N 路 K 样本任务中，总共有 $NK(\eta+1)$ 个序列。我们对这些嵌入序列取平均值，以初始化可学习的提示向量 $\mathbf{h}_1, \ldots, \mathbf{h}_M$。这些向量源自图上下文，因此比随机初始化能提供信息更丰富的起点。

4 实验

我们进行了大量实验，以评估所提出的 G2P2 方法，并与最先进的基线方法进行比较和模型分析。

4.1 实验设置

数据集。 使用了四个公开的图关联文本语料库，汇总见表 1。

• **Cora**：被称为“Cora研究论文分类”数据集[28], ，它是一个由相互链接的研究论文组成的集合。

表1：数据集统计信息。

数据集	Cora	Art	工业	M.I.
# 文档	25,120	1,615,902	1,260,053	905,453
# 链接	182,280	4,898,218	3,101,670	2,692,734
# 平均文档长度	141.26	54.23	52.15	84.66
# 平均节点度	7.26	3.03	2.46	2.97
# 类别总数	70	3,347	2,462	1,191

彼此通过引文相连。论文的摘要被视为文本文档。论文被划分为包含 73 个叶节点的主题层级。移除没有内容或标签的论文后，所得层级包含 70 个叶主题。请注意，我们使用的是更全面的 Cora 数据集版本，该版本比其他地方使用的版本更大，且类别更多 [17]。

• **Art、工业和乐器（M.I.）**是三个亚马逊评论数据集 [33], , 分别来自三个大类领域，即艺术、工艺与缝纫（Art）、工业与科学（工业）以及乐器（M.I.）。每种产品的描述均视为一个文档，而用户的所有评论则合并为一个文档，以反映用户的偏好。如果用户评论过某种产品，就在两者之间建立一条连接。大类领域下的产品子类别代表各个类别，这些类别粒度较细，可能多达数千种，且彼此差异细微。分类仅针对产品描述进行，用户评论则仅用于丰富文本语义。

对于所有数据集，我们使用 word2vec 算法 [30] 获取文本文件中每个词的 128 维词嵌入。然后，对于每个节点，我们对其文档中所有词的嵌入向量取平均值，并将所得平均向量用作基于图神经网络的方法的节点输入特征。

任务构建。我们进行零样本或少样本文本分类。我们采用*5-way*设置，即，从所有类别中抽取五个类别来构建一个任务。在每个任务中，我们进一步从每个类别中抽取 K 个样本，为 $K \in \{0, 1、\ldots\ldots、5\}$ 构建一个 K-shot 支持集，并构建一个与支持集大小相同的验证集。其余样本构成查询集。请注意，支持集带有标签，用作任务训练数据；而查询集不带标签，用于评估。还要注意，在我们的实验中会使用所有类别——只是每个任务包含五个类别，并且测试期间会进行多个任务，以覆盖所有类别。这是一种典型的任务设置 [7], , 可在不同类别组合下进行全面评估。报告的结果是每个数据集上所有任务结果的平均值。

少样本分类的基线方法。我们考虑了来自四个类别的具有竞争力的基线方法。

(1) 端到端图神经网络，即通过监督式学习，以随机初始化为起点，采用端到端方式训练的图神经网络。

• GCN [17]：一种在图上运行的卷积神经网络扩展。

• SAGEsup [9]：GraphSAGE 的监督式版本，是一种归纳式图神经网络，通过对节点局部邻域中的特征进行采样和聚合来生成节点嵌入。

• TextGCN [64]：一种基于图卷积网络的模型，运行于由词共现关系和文档-词关系构建的文本图上，可联合学习词和文档的嵌入。

(2) 预训练/自监督图神经网络，这些图神经网络使用无需标注数据的前置任务进行预训练，随后在冻结模型参数的同时进行微调或拟合分类头。

• GPT-GNN [15]：一种基于自监督图生成任务的图神经网络预训练方法，包括节点属性生成和边生成。它遵循"预训练、微调"的范式。

• DGI [50]：一种图神经网络预训练方法，通过最大化局部和全局表示之间的互信息来实现预训练。作为一种无监督方法，它还会冻结模型参数，并在预训练后，为下游少样本分类拟合一个简单的逻辑回归模型。

• SAGEself [9]：GraphSAGE 的自监督版本，通过鼓励相邻节点具有相似的嵌入、非相邻节点具有不同的嵌入来进行学习。预训练后，它采用与 DGI 相同的方法进行下游分类。

(3) 预训练变换器，通过掩码语言建模进行预训练 [16], , 然后与随机初始化的分类头（例如，一个全连接层）一起微调，以完成下游少样本分类任务。

• BERT [16]：一种通过掩码语言建模进行双向训练的 Transformer，在所有层中同时以左右文脉为条件，从无标注文本中学习。

• RoBERTa [25]：BERT 的复现版本，在训练过程中仔细衡量许多关键超参数和训练数据集大小的影响。

• BERT* 和 RoBERTa*：BERT 和 RoBERTa 的变体，分别通过在我们的数据集上使用掩码语言建模对预训练 BERT 和 RoBERTa 进行微调得到，以缓解我们的数据集与用于预训练 BERT 和 RoBERTa 的数据集之间的领域差距。

(4) 提示调优：P-Tuning v2 [23], 是一种针对自然语言优化和调整的前缀调优 [19] 版本。它采用深度提示调优，为预训练语言模型的每一层应用连续提示。

请注意，我们的设置不同于元学习范式下的少样本学习 [7], , 因为元训练阶段不存在少样本任务。因此，我们无法使用最先进的元学习模型进行比较。此外，我们比较的两个基线方法 DGI 和 SAGEself 均采用了线性探测，这种方法被认为是一种强大的少样本学习器 [47]。

零样本分类的基线方法。我们仅与预训练语言模型进行比较，因为所有其他方法至少需要一个样本才能运行。对于每种方法，我们使用离散提示 [CLASS]（即，仅使用标签文本）。我们还评估手工设计提示"prompt [CLASS]"，其中 prompt 是通过提示工程找到的一系列词元，并在模型名称后标注"+d"。本质上，我们计算目标文档与每个类别的标签文本（可附加或不附加其他词元）之间的相似度，并按照图 2 预测最相似的类别。

G2P2 和基线方法的设置。对于 G2P2，文本编码器是一个 Transformer [48]。按照 CLIP [36], , 我们使用一个拥有 6300 万参数、12 层、宽度为 512 且具有 8 个注意力头的模型。它采用

表 2：五样本分类性能（百分比），含 95% 置信区间。在每一列中，所有方法中的最佳结果以**粗体**显示，基线方法中的最佳结果加下划线。G2P2 的提升是相对于最佳基线方法计算的。 * 表示根据双尾 t检验，($p < 0.05$)我们的模型显著优于最佳基线方法。

	Cora		Art		工业		M.I.	
	准确率	宏平均 F1	准确率	宏平均 F1 值	准确率	宏平均 F1 值	准确率	宏平均 F1 值
GCN	41.15±2.41	34.50±2.23	22.47±1.78	15.45±1.14	21.08±0.45	15.23±0.29	22.54±0.82	16.26±0.72
$SAGE_{sup}$	41.42±2.90	35.14±2.14	22.60±0.56	16.01±0.28	20.74±0.91	15.31±0.37	22.14±0.80	16.69±0.62
TextGCN	59.78±1.88	55.85±1.50	43.47±1.02	32.20±1.30	53.60±0.70	45.97±0.49	46.26±0.91	38.75±0.78
GPT-GNN	76.72±2.02	72.23±1.17	65.15±1.37	52.79±0.83	62.13±0.65	54.47±0.67	67.97±2.49	59.89±2.51
DGI	78.42±1.39	74.58±1.24	65.41±0.86	53.57±0.75	52.29±0.66	45.26±0.51	68.06±0.73	60.64±0.61
$SAGE_{self}$	77.59±1.71	73.47±1.53	76.13±0.94	65.25±0.31	71.87±0.61	65.09±0.47	77.70±0.48	70.87±0.59
BERT	37.86±5.31	32.78±5.01	46.39±1.05	37.07± 0.68	54.00±0.20	47.57±0.50	50.14±0.68	42.96±1.02
BERT*	27.22±1.22	23.34±1.11	45.31±0.96	36.28±0.71	49.60±0.27	43.36±0.27	40.19±0.74	33.69±0.72
RoBERTa	62.10±2.77	57.21±2.51	72.95±1.75	62.25±1.33	76.35±0.65	70.49±0.59	70.67±0.87	63.50±1.11
RoBERTa*	67.42±4.35	62.72±3.02	74.47±1.00	63.35±1.09	77.08±1.02	71.44±0.87	74.61±1.08	67.78±0.95
P-Tuning v2	71.00±2.03	66.76±1.95	76.86±0.59	66.89±1.14	79.65±0.38	74.33±0.37	72.08±0.51	65.44±0.63
G2P2-p	79.16±1.23	74.99±1.35	79.59±0.31	68.26±0.43	80.86±0.40	74.44±0.29	81.26±0.36	74.82±0.45
G2P2 （提升。）	**80.08*** ±1.33 (+2.12%)	**75.91*** ±1.39 (+1.78%)	**81.03*** ±0.43 (+5.43%)	**69.86*** ±0.67 (+4.44%)	**82.46*** ±0.29 (+3.53%)	**76.36*** ±0.25 (+2.7%)	**82.77*** ±0.32 (+6.53%)	**76.48*** ±0.52 (+7.92%)

文本的小写字节对编码（BPE）表示，词汇表大小为 49,152 [41]。最大序列长度设为 128。图编码器采用图卷积网络 [17], , 由两层组成 [9] ， 每层均使用带泄漏的修正线性单元激活函数, 维度为 128 [34]。我们的模型从头开始预训练，不使用先前预训练的权重初始化图编码器和文本编码器。λ 在公式（10）中设为 0。1（用于 Cora），在三个亚马逊评论数据集上则设为 10；这些取值是根据验证数据上的准确率，从 {0 1, 1, 10, 100 01, 0. . } 中选出的。公式（13）中的可学习提示词元数量 M 设为 4，该取值是根据验证数据从 {2, 4, 8, 16, 32} 中选出的。我们使用 Adam优化器，学习率为 2×10^{-5}，预训练 2 轮，批次大小为 64，设置参考了 Hugging Face 的 [56] 示例。文本嵌入维度为 128，与图编码器的输出维度相同。为生成摘要嵌入和基于上下文的提示初始化，采样的相邻节点数量设为 3。对于提示调优，学习率设为 0。01，该值是根据验证数据上的准确率，从 {0.0001,0.001,0.01,0.1} 中选出的。

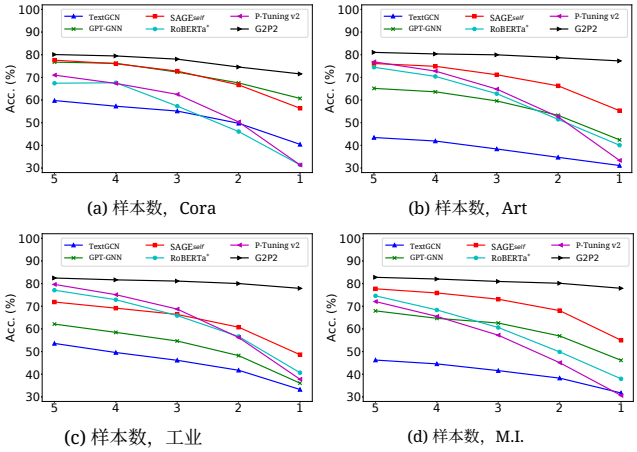
对于所有图神经网络方法（包括 G2P2 中的图神经网络组件），我们使用 128 维的 word2vec 嵌入 [30] ，并对原始文本中的词进行平均，作为输入节点特征。我们采用两层架构，并将隐藏维度设为 128；但对于图卷积网络和 SAGEsup，隐藏维度设为 32 [17] ，因为这样在实证中表现更好。对于所有图神经网络预训练基线方法，我们将学习率设为 0.01。对于 BERT、RoBERTa 和 G2P2，我们采用 0.00002 的学习率。我们的 BERT 、RoBERTa 及其掩码语言建模实现基于 Hugging Face 的 transformers [56]。对于 BERT 和 RoBERTa，我们均使用基础版，因为我们的模型 G2P2 仅有一个 6300 万参数的小型模型，这与先前的工作一致 [36]。对于 P-Tuning v2，我们使用基于 RoBERTa 骨干网络的原始代码，并采用推荐的 0.005 作为提示调优的学习率。对于 G2P2，提示调优的学习率设为 0.01。

所有实验均在一台配备 4 块 GeForce RTX 3090 GPU 的服务器上进行。在单块 GPU 上，G2P2 在 Cora/M.I./Industrial/Art 数据集上的预训练分别耗时约 0.5/6/9/10 小时。推理（采用提示调优）基于由五个随机种子 {1,2, 4, 8,16} 生成的五种不同数据划分进行。

4.2 低资源分类的性能我们评估了不同样本数下的分类性能。

五个样本。 在表 2 中，我们首先比较了 G2P2 与基线方法在 *5* 个样本设置下的性能。G2P2 始终表现最佳，其性能比最优基线方法高出约 2–8%，且具有统计显著性。

我们还得出了一些其他观察结果。首先，在图神经网络中，预训练/自监督模型往往优于端到端方法，因为后者严重依赖标注数据。在前者中，DGI 和 SAGEself 表现更好，因为它们属于线性探测，而线性探测被认为是强大的少样本学习器 [47]。请注意，除了使用原始文本的 word2vec 嵌入 [30] 作为节点特征外，我们还尝试使用预训练 RoBERTa [25] 为 DGI 和 SAGEself 生成节点特征。然而，这样做并未带来任何改进，说明简单地以解耦方式结合语言模型和图神经网络并无成效。相比之下，我们提出的模型通过三种以图为基础的对比策略，联合学习文本编码器和图编码器。其次，预训练语言模型通常优于图神经网络，这凸显了以细粒度方式利用文本的重要性。此外，RoBERTa 凭借改进的预训练流程 [25], 表现优于 BERT。然而，在我们的文本数据上进一步微调预训练语言模型，结果喜忧参半：RoBERTa* 略优于 RoBERTa，但 BERT* 远逊于 BERT。换言之，仅靠简单的方法并不能轻易缩小领域差距。



(a) 样本数，Cora (b) 样本数，Art

(c) 样本数，工业 (d) 样本数，M.I.

图 3：不同样本数下的性能。

使用领域文本进行微调。第三，连续提示方法 P-Tuning v2 的效果与微调相比具有竞争力，同时成本远低于微调。然而，我们的模型 G2P2 仍显著优于它。此外，不进行提示调优的 G2P2-p 表现逊于 G2P2，这表明连续提示具有优势。

样本数较少。 除了 5样本设置外，我们还在图 3 中研究了较少样本数对 G2P2 和若干有代表性的基线方法的影响。总体而言，在不同样本数下，G2P2 的表现始终最佳。一般来说，随着样本数减少，所有方法的性能都会下降。然而，在极低资源（例如，1 样本或 2 样本）设置下，基线方法的性能会大幅下降。相比之下，G2P2 依然稳健，即使只有 1 或 2 个样本，性能下降幅度也相对较小。

结果表明，我们提出的模型具有实际价值，尤其是在标注数据难以或需要付出高昂成本才能及时获取时。另一方面，在动态且开放的环境中，传统方法始终面临无法跟上新兴类别快速增长的挑战 [53]。例如，为在线文章中的新主题或开放式电子商务平台中的新产品类别标注大量文本，可能会造成显著的时间延迟。

零样本。最后，我们在表3中报告了零样本性能，其中我们的模型 G2P2 和 G2P2+d 显著优于基线方法。结果尤其证明了我们的图支撑的对比预训练在缺少标注数据时的有效性；在许多现实场景中，这对于处理没有任何标注样本的不断演化的类别至关重要。此外，手工设计的离散提示（即，BERT*+d 和 G2P2+d）可能优于仅使用标签文本（即，BERT* 和 G2P2），展现了额外提示词元的有效性。

然而，寻找最优的离散提示通常需要大量的提示工程工作。具体而言，对于采用离散提示的三种方法，即 RoBERTa*+d、BERT*+d 和 G2P2+d，我们在每个数据集上探索了 10 多种人工设计的提示模板。这些模板通常与相应的数据集相关，设计时也需要一定的领域知识。虽然离散

<escape>**表 3：零样本分类准确率（百分比）。**有关条目样式的说明，请参见表 2。</escape>

	Cora	Art	工业	M.I.
RoBERTa	30.46±2.01	42.80±0.94	42.89±0.97	36.40±1.20
RoBERTa*	39.58±1.26	34.77±0.65	37.78±0.32	32.17±0.68
RoBERTa*+d	45.53±1.33	36.11±0.66	39.40±1.22	37.65±0.33
BERT	23.58±1.88	35.88±1.44	37.32±0.85	37.42±0.80
BERT*	23.38±1.96	54.27±1.85	56.02±1.22	50.19±0.72
BERT*+d	26.65±1.71	56.61±1.76	55.93±0.96	52.13±0.88
G2P2	63.52±2.89	76.52±0.59	76.66±0.31	74.60±0.62
G2P2+d	**65.28*** ±3.12	**76.99*** ±0.60	**77.43*** ±0.27	**75.86*** ±0.69
（改进。）	(+45.38%)	(+36.00%)	(+38.22%)	(+45.52%)

提示通常有助于零样本分类，但其有效性各不相同。在表 3 中，我们仅报告了每种方法在每个数据集上表现最佳的人工设计模板。还值得注意的是，同一提示有时会在不同模型上产生截然相反的结果。例如，在 Cora 数据集中，“a model of [CLASS]" 是 RoBERTa*+d 的最佳提示，却不适用于 G2P2+d。此外，有些不具备任何语义含义的提示，例如 “a [CLASS]"，有时反而是最佳选择。这些观察表明，提示工程需要投入大量人工劳动，而且最优离散提示的选择存在很大的不确定性。因此，仅使用标签文本仍然是一个相当不错的选择。

4.3 模型分析

我们对 G2P2 进行了更深入的研究。除非另有说明，我们报告的是准确率，采用*5* 样本设置。

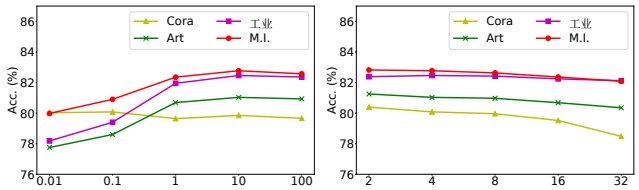
消融研究。 我们首先通过采用所提出的损失项 L_1、L_2 和 L_3的不同组合，评估三种基于图交互的对比策略各自的贡献。如表 4 所示，不包含 L_1 的策略表现相当糟糕，这表明双射文本 - 节点交互是我们预训练的基础组成部分。尽管如此，在 L_1的基础上进一步加入 L_2 或 L_3 时，我们仍然观察到明显的性能提升，说明为文本数据引入额外的基于图的交互是有益的。最后，采用全部三个损失项的 G2P2 优于所有由 1 个或 2 个损失项组成的组合，表明这三种对比策略都很有用，并且能够得到良好整合。总体而言，结果表明图信息对于低资源文本分类至关重要，因为图结构揭示了文档之间丰富的关系。

接下来，我们评估提示调优方法的贡献。具体而言，我们将 G2P2 与两个消融变体进行比较：仅使用标签文本而不采用可训练的提示向量，以及对提示向量进行随机初始化。如表 4 所示，仅使用标签文本会明显降低分类性能，这说明通过提示调优学习连续提示的重要性。此外，与随机初始化相比，采用基于上下文的提示向量初始化方法的 G2P2 具有小幅但稳定的优势，这表明在提示调优中考虑图结构是有益的。

超参数研究。 我们首先研究图 4(a) 中交互系数 λ 的影响，该系数平衡高阶

表4：消融研究。

	Cora	Art	工业	M.I.
仅 L_3	74.66±1.80	52.56±1.09	45.97±0.81	49.05±0.54
仅 L_2	77.01±1.30	58.90±0.55	52.99±0.46	59.41±0.85
仅 L_1	79.50±1.19	77.37±0.72	78.10±0.34	79.70±0.56
L_2+L_3	70.04±2.89	49.91±1.57	50.07±0.50	56.14±1.01
L_1+L_3	79.73±0.89	78.60±0.40	79.97±0.43	80.42±0.45
L_1+L_2	79.42±1.04	80.55±0.52	81.06±0.33	82.39±0.41
仅标签文本	79.16±1.23	79.59±0.31	80.86±0.40	81.26±0.36
随机初始化	80.03±0.99	80.85±0.43	82.43±0.33	82.64±0.21
G2P2	**80.08**±1.33	**81.03**±0.43	**82.46**±0.35	**82.77**±0.32



(a) 交互系数，λ (b) 提示长度，M

图 4：超参数研究。

对比损失（L_2，L_3）。当 λ 略大时（例如，≥ 10），模型性能通常更好且更稳定，这表明高阶文本-摘要交互和节点-摘要交互的重要性。接下来，我们研究图 4(b) 中的提示长度 M，即第 3.4 节中可训练提示向量的数量。提示长度对性能的影响相对较小，因此选择较小的 {v12>（例如，4）可提高效率且具有较强的稳健性。

提示调优的效率。 在我们的工作中，连续提示通过提示调优进行优化 [23, 71]，无需更新预训练模型。在本实验中，我们研究了 G2P2 中提示调优的效率，并将其与传统微调的效率进行比较。由于 G2P2 包含一个 Transformer 组件，我们将其与四种基于 Transformer 的模型进行比较，这些模型都遵循经典的"预训练、微调"范式 [16]。

如表5所示，"每个任务的调优时间"指的是 G2P2 中进行提示调优或基线方法中进行微调时每个任务所需的平均时间，而"参数规模"指的是需要更新的参数数量。结果表明，G2P2 中的提示调优比基线方法中的微调高效得多，实现了 2.1~18.8 倍的加速。原因在于，提示调优更新的参数少得多。在 G2P2 中，我们使用了 4 个可训练的 512 维提示向量，总计仅 2048 个参数；而基线方法中的微调需要更新整个预训练模型，涉及超过 1 亿个参数。需要注意的是，由于数据加载器和优化器的开销，加速比与参数规模并非线性相关。总体而言，我们的提示调优不仅在低资源设置下有效，而且在参数和计算方面都十分高效。

泛化研究。 我们之前的实验可视为"传导式"，因为文本编码器的预训练

表5：调优时间和参数规模。

	每项任务的调优时间（秒）				参数。
	Cora	Art	工业	M.I.	size
RoBERTa	45.47±2.38	64.22±3.62	43.46±2.99	44.99±2.58	123 M
RoBERTa*	39.38±2.01	59.56±3.55	35.10±2.75	38.84±2.39	123 M
BERT	32.23±1.71	51.77±2.00	31.72±1.77	33.55±2.39	110 M
BERT*	34.82±1.68	55.16±2.32	31.11±1.74	29.00±2.23	110 M
G2P2	**2.42**±0.41	**22.03**±1.39	**14.63**±1.26	**12.72**±1.17	**2048**

表 6：文本分类的归纳性能。

	Art	工业	M.I.
BERT*	43.66±0.90	48.35±0.25	39.24±0.88
RoBERTa*	69.55±1.14	73.65±0.86	71.96±1.44
G2P2	**79.81**±0.22	**81.29**±0.32	**81.85**±0.33

和下游分类都是在整个语料库上进行的。为了进一步评估模型的泛化能力，我们采用"归纳式设置"：仅使用语料库的一个子集预训练文本编码器，并在另一个不重叠的子集上进行下游分类。具体而言，在这三个亚马逊数据集中，由于用户文本没有标签，而物品文本带有标签，因此我们自然可以仅使用用户文本进行预训练，并仅对物品文本进行下游分类。我们还仅使用用户文本对 BERT 和 RoBERTa 进行掩码语言建模，得到 BERT* 和 RoBERTa*。如表 6 所示，G2P2 在归纳式设置下仍表现出色，说明我们的预训练模型具有很强的泛化能力。

5 结论

本文研究了低资源文本分类问题。鉴于许多文本文件通过底层网络彼此关联，我们提出了一种名为图驱动预训练与提示（G2P2）的新模型。该模型由预训练阶段的三种基于图交互的对比策略，以及一种用于下游分类的联合预训练图文模型提示机制构成。我们进行了大量实验，结果表明，G2P2 在零样本和少样本文本分类中具有优势。

这项工作的一项局限是需要借助图来补充文本。尽管图在信息检索应用中无处不在，但在缺少自然图的情况下，一种可行的解决方案是基于词语共现或其他关系构建合成图，例如，将时间和地点相近的新闻文章连接起来。我们将进一步探索留待未来工作。

致谢

本研究／项目由新加坡教育部通过其学术研究基金第二级（提案编号：T2EP20122-0041）资助。本材料中表达的任何观点、研究结果、结论或建议均属于作者本人，并不代表新加坡教育部的观点。

参考文献

[1] Trapit Bansal、Rishikesh Jha、Tsendsuren Munkhdalai 和 Andrew McCallum。2020。用于少样本自然语言分类任务的自监督元学习。载于 EMNLP。522–534。[2] Yujia Bao、Menghua Wu、Shiyu Chang 和 Regina Barzilay。2020。利用分布特征进行少样本文本分类。载于 ICLR。[3] Tom Brown、Benjamin Mann、Nick Ryder、Melanie Subbiah、Jared D Kaplan、Prafulla Dhariwal、Arvind Neelakantan、Pranav Shyam、Girish Sastry、Amanda Askell 等。2020。语言模型是少样本学习器。NeurIPS 33（2020），1877–1901。[4] Xianshuai Cao、Yuliang Shi、Han Yu、Jihu Wang、Xinjun Wang、Zhongmin Yan 和 Zhiyong Chen。2021。DEKR：用于机器学习方法推荐的描述增强知识图谱。载于 SIGIR。203–212。[5] Jiaao Chen、Zichao Yang 和 Diyi Yang。2020。MixText：用于半监督文本分类的语言学信息引导隐空间插值。载于 ACL。2147–2157。[6] Xiang Chen、Ningyu Zhang、Xin Xie、Shumin Deng、Yunzhi Yao、Chuanqi Tan、Fei Huang、Luo Si 和 Huajun Chen。2022。Knowprompt：用于关系抽取的知识感知提示调优与协同优化。载于 ACM Web Conference 2022 论文集。2778–2788。[7] Chelsea Finn、Pieter Abbeel 和 Sergey Levine。2017。用于深度网络快速适应的模型无关元学习。载于 ICML。1126–1135。[8] Tianyu Gao、Adam Fisch 和 Danqi Chen。2021。让预训练语言模型成为更出色的少样本学习器。载于 ACL。3816–3830。[9] Will Hamilton、Zhitao Ying 和 Jure Leskovec。2017。在大型图上进行归纳式表示学习。载于 NeurIPS。[10] Xu Han、Zhengyan Zhang、Ning Ding、Yuxian Gu、Xiao Liu、Yuqi Huo、Jiezhong Qiu、Yuan Yao、Ao Zhang、Liang Zhang 等。2021。预训练模型：过去、现在与未来。AI Open 2（2021），225–250。[11] Xu Han、Weilin Zhao、Ning Ding、Zhiyuan Liu 和 Maosong Sun。2021。Ptr：利用规则进行文本分类提示调优。arXiv 预印本 arXiv:2105.11259（2021）。[12] Xu Han、Hao Zhu、Pengfei Yu、Ziyun Wang、Yuan Yao、Zhiyuan Liu 和 Maosong Sun。2018。FewRel：大规模监督式少样本关系分类数据集及先进评测。载于 EMNLP。4803–4809。[13] Shengding Hu、Ning Ding、Huadong Wang、Zhiyuan Liu、Jingang Wang、Juanzi Li、Wei Wu 和 Maosong Sun。2022。知识增强提示调优：将知识融入文本分类的提示词表述器。载于 ACL。2225–2240。[14] Weihua Hu、Bowen Liu、Joseph Gomes、Marinka Zitnik、Percy Liang、Vijay Pande 和 Jure Leskovec。2019。图神经网络的预训练策略。载于 ICLR。[15] Ziniu Hu、Yuxiao Dong、Kuansan Wang、Kai-Wei Chang 和 Yizhou Sun。2020。GPT-GNN：图神经网络的生成式预训练。载于 KDD。1857–1867。[16] Jacob Devlin Ming-Wei Chang Kenton 和 Lee Kristina Toutanova。2019。BERT：用于语言理解的深度双向 transformers 预训练。载于 NAACL。4171–4186。[17] Thomas N. Kipf 和 Max Welling。2017。利用图卷积网络进行半监督分类。载于 ICLR。OpenReview.net。[18] Brian Lester、Rami Al-Rfou 和 Noah Constant。2021。规模对参数高效提示调优的作用。载于 EMNLP。3045–3059。[19] Xiang Lisa Li 和 Percy Liang。2021。前缀调优：优化用于生成的连续提示。载于 ACL。4582–4597。[20] Hu Linmei、Tianchi Yang、Chuan Shi、Houye Ji 和 Xiaoli Li。2019。用于半监督短文本分类的异构图注意力网络。载于 EMNLP。4821–4830。[21] Jie Liu、Zhicheng He、Lai Wei 和 Yalou Huang。2018。内容到节点：自翻译网络嵌入。载于 KDD。1794–1802。[22] Pengfei Liu、Weizhe Yuan、Jinlan Fu、Zhengbao Jiang、Hiroaki Hayashi 和 Graham Neubig。2021。预训练、提示与预测：自然语言处理中的提示方法系统综述。arXiv 预印本 arXiv:2107.13586（2021）。[23] Xiao Liu、Kaixuan Ji、Yicheng Fu、Weng Tam、Zhengxiao Du、Zhilin Yang 和 Jie Tang。2022。P-Tuning：提示调优在不同规模和任务上的表现可与微调相媲美。载于 ACL。61–68。[24] Xiao Liu、Yanan Zheng、Zhengxiao Du、Ming Ding、Yujie Qian、Zhilin Yang 和 Jie Tang。2021。GPT 也能理解。arXiv 预印本 arXiv:2103.10385（2021）。[25] Yinhan Liu、Myle Ott、Naman Goyal、Jingfei Du、Mandar Joshi、Danqi Chen、Omer Levy、Mike Lewis、Luke Zettlemoyer 和 Veselin Stoyanov。2019。RoBERTa：一种稳健优化的 BERT 预训练方法。arXiv 预印本 arXiv:1907.11692（2019）。[26] Yong Liu、Susen Yang、Chenyi Lei、Guoxin Wang、Haihong Tang、Juyong Zhang、Aixin Sun 和 Chunyan Miao。2021b。利用多模态辅助信息对图 Transformer 进行预训练，以用于推荐。载于 ACMMM。2853–2861。[27] Yuanfu Lu、Xunqiang Jiang、Yuan Fang 和 Chuan Shi。2021。学习如何预训练图神经网络。载于 AAAI，第35卷。4276–4284。[28] Andrew McCallum、Kamal Nigam、Jason Rennie 和 Kristie Seymore。2000。利用机器学习自动构建互联网门户网站。《信息检索期刊》3（2000），127–163。www.research.whizbang.com/data。[29] Andrew Kachites McCallum、Kamal Nigam、Jason Rennie 和 Kristie Seymore。2000。利用机器学习自动构建互联网门户网站。《信息检索》3，2（2000），127–163。[30] Tomas Mikolov、Kai Chen、Greg Corrado 和 Jeffrey Dean。2013。在向量空间中高效估计词表示。arXiv 预印本 arXiv:1301.3781（2013）。[31] Sewon Min、Mike Lewis、Hannaneh Hajishirzi 和 Luke Zettlemoyer。2022。用于少样本文本分类的噪声信道语言模型提示。载于 ACL。5316–5330。[32] Takeru Miyato、Andrew M. Dai 和 Ian J. Goodfellow。2017。半监督文本分类的对抗训练方法。载于 ICLR。OpenReview.net。[33] Jianmo Ni、Jiacheng Li 和 Julian McAuley。2019。利用远程标注的评论和细粒度方面为推荐提供依据。载于 EMNLP。188–197。[34] Bryan Perozzi、Rami Al-Rfou 和 Steven Skiena。2014。Deepwalk：社交表示的在线学习。载于 KDD。701–710。[35] Guanghui Qin 和 Jason Eisner。2021。学习如何提问：利用软提示混合查询语言模型。载于 NAACL。5203–5212。[36] Alec Radford、Jong Wook Kim、Chris Hallacy、Aditya Ramesh、Gabriel Goh、Sandhini Agarwal、Girish Sastry、Amanda Askell、Pamela Mishkin、Jack Clark 等。2021。通过自然语言监督学习可迁移的视觉模型。载于 ICML。PMLR，8748–8763。[37] Alec Radford、Karthik Narasimhan、Tim Salimans、Ilya Sutskever 等。2018。通过生成式预训练改进语言理解。（2018）。[38] Colin Raffel、Noam Shazeer、Adam Roberts、Katherine Lee、Sharan Narang、Michael Matena、Yanqi Zhou、Wei Li、Peter J Liu 等。2020。探索统一文本到文本 Transformer 的迁移学习极限。《机器学习研究期刊》21，140（2020），1–67。[39] Oscar Sainz、Oier Lopez de Lacalle、Gorka Labaka、Ander Barrena 和 Eneko Agirre。2021。用于有效零样本和少样本关系抽取的标签词汇化与蕴含。载于 EMNLP。1199–1212。[40] Timo Schick 和 Hinrich Schütze。2021。利用完形填空问题进行少样本文本分类和自然语言推理。载于 EACL。255–269。[41] Rico Sennrich、Barry Haddow 和 Alexandra Birch。2016。利用子词单元对稀有词进行神经机器翻译。载于 ACL。计算语言学协会（ACL），1715–1725。[42] Taylor Shin、Yasaman Razeghi、Robert L Logan IV、Eric Wallace 和 Sameer Singh。2020。AutoPrompt：利用自动生成的提示从语言模型中 eliciting 知识。载于 EMNLP。4222–4235。[43] Kihyuk Sohn。2016。采用多类N对损失目标改进深度度量学习。NeurIPS 29（2016）。[44] Mingchen Sun、Kaixiong Zhou、Xin He、Ying Wang 和 Xin Wang。2022。Gppt：图预训练和提示调优以实现图神经网络的泛化。载于 KDD。1717–1727。[45] Yi Sun、Yu Zheng、Chao Hao 和 Hangping Qiu。2021。NSP-BERT：通过原始预训练任务——下一句预测实现的基于提示的零样本学习器。arXiv 预印本 arXiv:2109.03564（2021）。[46] Zhixing Tan、Xiangwen Zhang、Shuo Wang 和 Yang Liu。2022。MSP：通过多阶段提示让预训练语言模型成为更好的翻译器。载于 ACL。6131–6142。[47] Yonglong Tian、Yue Wang、Dilip Krishnan、Joshua B Tenenbaum 和 Phillip Isola。2020。重新思考少样本图像分类：良好的嵌入就是你所需要的一切？载于 ECCV。Springer，266–282。[48] Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Łukasz Kaiser 和 Illia Polosukhin。2017。注意力机制就是你所需要的一切。NeurIPS 30（2017）。[49] Petar Veličković、Guillem Cucurull、Arantxa Casanova、Adriana Romero、Pietro Liò 和 Yoshua Bengio。2018。图注意力网络。载于 ICLR。[50] Petar Velickovic、William Fedus、William L Hamilton、Pietro Liò、Yoshua Bengio 和 R Devon Hjelm。2019。深度图互信息最大化。ICLR 2，3（2019），4。[51] Guoyin Wang、Chunyuan Li、Wenlin Wang、Yizhe Zhang、Dinghan Shen、Xinyuan Zhang、Ricardo Henao 和 Lawrence Carin。2018。用于文本分类的词与标签联合嵌入。载于 ACL。2321–2331。[52] Ning Wang、Minnan Luo、Kaize Ding、Lingling Zhang、Jundong Li 和 Qinghua Zheng。2020。利用属性匹配进行图少样本学习。载于 CIKM。1545–1554。[53] Zheng Wang、Jialong Wang、Yuchen Guo 和 Zhiguo Gong。2021。利用分解式图原型网络进行零样本节点分类。载于 KDD。1769–1779。[54] Zhihao Wen。2023。跨图和时间泛化图神经网络。载于 WSDM。1214–1215。[55] Zhihao Wen、Yuan Fang 和 Zemin Liu。2021。跨图元归纳节点分类。载于 SIGIR。1219–1228。[56] Thomas Wolf、Lysandre Debut、Victor Sanh、Julien Chaumond、Clement Delangue、Anthony Moi、Pierric Cistac、Tim Rault、Rémi Louf、Morgan Funtowicz 等。2020。Transformers：最先进的自然语言处理。载于 EMNLP。38–45。[57] Lirong Wu、Haitao Lin、Cheng Tan、Zhangyang Gao 和 Stan Z Li。2021。图上的自监督学习：对比式、生成式还是预测式。《TKDE》（2021）。

[58] Zonghan Wu、Shirui Pan、Fengwen Chen、Guodong Long、Chengqi Zhang 和 S Yu Philip。2020。图神经网络综合综述。*TNNLS* 32，1（2020），4–24。[59] Qizhe Xie、Zihang Dai、Eduard Hovy、Thang Luong 和 Quoc Le。2020。用于一致性训练的无监督数据增强。*NeurIPS* 33（2020），6256–6268。[60] Hu Xu、Bing Liu、Lei Shu 和 P Yu。2019。开放世界学习及其在产品分类中的应用。载于 万维网大会。3413–3419。[61] Keyulu Xu、Weihua Hu、Jure Leskovec 和 Stefanie Jegelka。2019。图神经网络的能力有多强？载于 *ICLR*。OpenReview.net。[62] Zhilin Yang、William Cohen 和 Ruslan Salakhudinov。2016。重新审视利用图嵌入进行半监督学习。载于 *ICML*。PMLR，40–48。[63] Zhilin Yang、Zihang Dai、Yiming Yang、Jaime Carbonell、Russ R Salakhutdinov 和 Quoc V Le。2019。XLNet：用于语言理解的广义自回归预训练。*NeurIPS* 32（2019）。[64] Liang Yao、Chengsheng Mao 和 Yuan Luo。2019。用于文本分类的图卷积网络。载于 *AAAI*，第 33 卷。7370–7377。[65] Mo Yu、Xiaoxiao Guo、Jinfeng Yi、Shiyu Chang、Saloni Potdar Yu Cheng Gerald Tesauro、Haoyu Wang Bowen Zhou 和 AI Foundations-Learning。2018。多样

少样本文本分类与多种度量指标。在 *NAACL*。1206–1215。[66] Wenxuan Zhang、Yang Deng、Xin Li、Yifei Yuan、Lidong Bing 和 Wai Lam。2021。方面情感四元组预测作为释义生成。在 *EMNLP*。9209–9219。[67] Yuhao Zhang、Hang Jiang、Yasuhide Miura、Christopher D Manning 和 Curtis P Langlotz。2020。通过配对图像和文本进行医学视觉表征的对比学习。***arXiv*** 预印本 ***arXiv:2010.00747***（2020）。[68] Jianan Zhao、Meng Qu、Chaozhuo Li、Hao Yan、Qian Liu、Rui Li、Xing Xie 和 Jian Tang。2023。通过变分推断在大规模文本属性图上学习。在 *ICLR*。https://openreview.net/forum?id=q0nmYciuuZN [69] Zexuan Zhong、Dan Friedman 和 Danqi Chen。2021。事实探测是 [MASK]：学习与学习回忆。在 *NAACL*。5017–5033。[70] Fan Zhou、Chengtai Cao、Kunpeng Zhang、Goce Trajcevski、Ting Zhong 和 Ji Geng。2019。Meta-GNN：关于图元学习中的少样本节点分类。在 *CIKM*。2357–2360。[71] Kaiyang Zhou、Jingkang Yang、Chen Change Loy 和 Ziwei Liu。2022。学习为视觉语言模型设计提示。*IJCV* 130，9（2022），2337–2348。