

基于两阶段提示学习的少样本命名实体识别

邵佳兴 黄琪 肖聪 刘璟 罗文兵 王明文*
江西师范大学 计算机信息工程学院 江西 南昌 330022
Email: {shaojiaxing, huangqi, lwb, mwwang}@jxnu.edu.cn

摘要

少样本命名实体识别旨在用少量的标注数据来识别命名实体。近年来受提示学习在少样本场景中表现良好性能的启发，本文探索了基于提示的少样本命名实体识别的方法。已有的基于提示学习的方法是通过列举所有可能的跨度来进行实体识别，这导致了计算成本高以及对实体边界信息未充分利用的问题。本文提出一种基于提示学习的两阶段框架TSP-Few，在不使用源域数据的情况下，进行少样本命名实体识别。第一阶段对种子跨度进行增强、过滤和扩展，其中种子增强模块能够让种子跨度捕获到更丰富的语义信息，种子过滤器能够减少大量的无关跨度，种子扩展模块能够充分利用实体的边界信息，为实体类型分类提供高质量的候选实体跨度。第二阶段利用提示学习方法预测候选跨度的相应类别。此外，为了缓解跨度检测阶段的错误累积，在实体分类阶段引入了负采样策略。跨度检测和实体类型分类任务的独立训练更容易在少样本情况下取得优异的性能。在三个基准数据集上的实验表明，与先进的方法相比，本文提出的方法在性能上有了进一步的提升，并且实验结果也表明了该文模型各个模块的有效性。

关键词： 少样本学习；命名实体识别；提示学习

Two-Stage Prompt Learning for Few-Shot Named Entity Recognition

Jiaxing Shao Qi Huang Cong Xiao Jing Liu Wenbing Luo Mingwen Wang*
School of Computer and Information Engineering, Jiangxi Normal University,
Nanchang, Jiangxi 330022, China
Email: {shaojiaxing, huangqi, lwb, mwwang}@jxnu.edu.cn

Abstract

Few-Shot Named Entity Recognition (FS-NER) is designed to identify named entities using a limited amount of annotated data. Inspired by the effective performance of prompt-based learning in few-shot scenarios, This research examines methodologies for FS-NER that utilize prompting techniques. Conventional prompt-based methods involve enumerating all potential spans for entity recognition, which leads to high computational costs and insufficient use of boundary information about the entities. This

©2024 中国计算语言学大会

根据《Creative Commons Attribution 4.0 International License》许可出版

基金项目：国家自然科学基金(62266023); 江西省教育厅研究生创新基金项目(YC2023-s297); 江西省教育厅科学技术研究项目(GJJ210325, GJJ2200354)

通讯作者：王明文

paper proposes a two-stage framework based on prompt learning for FS-NER that operates without source domain data. In the first stage, seed spans are enhanced, filtered, and expanded: enhancement allows seed spans to capture richer semantic information, filtering eliminates a substantial number of irrelevant spans, and expansion utilizes the boundary information of entities effectively. The second stage employs prompt-based learning methods to predict the categories of the candidate spans. Moreover, to alleviate the accumulation of errors during the span detection phase, a negative sampling strategy is introduced in the entity classification stage. Independent training of span detection and entity type classification tasks proves to be more effective in few-shot settings. Experimental results on three benchmark datasets indicate that the proposed method shows further improvement over advanced methods, and the experiments also demonstrate the effectiveness of each module within the model.

Keywords: few-shot learning , named entity recognition , prompt learning

1 引言

命名实体识别 (Named Entity Recognition, NER) 作为一项基本的信息抽取任务, 广泛应用于自然语言处理的下游任务。其目的是识别文本中的实体跨度, 并将其分类为预先定义的实体类型类别, 比如地名、组织名、时间、人名。尽管它具有挑战性, 当前研究人员基于深度学习的方法已经取得了良好的性能 (Lample et al., 2016; Ma and Hovy, 2016; Chiu and Nichols, 2016; Peters et al., 2017)。然而, 基于深度学习的方法依赖于大量的标注训练数据, 这需要耗费大量时间和高度依赖于劳动密集度。除此之外, 在一些特定的领域, 需要特定领域知识, 这是相当昂贵的。因此, 近年来, 在资源有限的情况下, 进行命名实体识别, 引起了学者们的广泛关注。

少样本命名实体识别的方法可以分为两类: 基于跨度的方法和基于序列标注的方法。基于序列标注的方法通常是将命名实体识别任务看作是序列标注任务, 给每句话中的单词都分配一个标签。(Hou et al., 2020; Ballesteros et al., 2022)通过计算每个查询标记与支持标记之间的距离, 在标记层面上直接对实体类型进行分类。然而, 这些方法受到非实体标记 (“O”类) 和标记的标签依赖性的干扰。另一方面, 基于度量学习的方法已经被应用到少样本命名实体识别。(Snell et al., 2017)提出原型网络来学习一个度量空间, 其中特定类别的一个查询例子聚集在一个原型周围。由于原型表示只是从几个例子中学习的, 而other类往往是有非常多的噪声, 所以无法在少样本命名实体识别达到令人满意的表现。

上述方法, 都是从单词级别上进行命名实体识别, 除此之外, 命名实体识别任务还可以被视为跨度级别上的分类任务。大部分少样本命名实体识别均采用端到端度量学习的方法 (Huang et al., 2020), 此类方法需要同时完成跨度检测和跨度分类两个任务。因此, 在少样本场景下, 仅依靠少数样本难以同时对两个任务充分学习, 从而导致此类方法会产生实体边界预测错误, 难以获得较好性能。(Sohrab and Miwa, 2018)枚举出文本中所有的可能跨度, 然后进行跨度分类。然而, 这忽略了实体的边界信息, 并且会产生大量的低质量的候选跨度, 这严重影响模型性能。

最近, 一些基于提示学习的工作被提出来用于少数命名实体识别。提示学习旨在通过将目标任务重新表述为完形填空问题 (Schick and Schütze, 2021)来提高预训练语言模型 (Pretrained Language Model, PLM) 的性能, 这是PLM的预训练任务。例如, 在执行情感分类任务时, 使用带有[MASK]标记的模板将输入“我喜欢这部电影”转换为“我喜欢这部电影, 它是[MASK]”。然后, PLM直接为[MASK]标记预测一个词“great”, 并通过预测的词将输入归入正类。目标任务的转换减少了预训练和微调之间的差距, 并有助于在少样本场景下工作。

为了解决基于跨度方法对实体边界信息利用不充分和枚举跨度带来的大量低质量的候选跨度的问题, 受文献 (Yang et al., 2022)提出方法的启发, 为了提高命名实体识别性能, 本文提出了一种基于提示的两阶段框架, 用于少样本命名实体识别。具体来说, 在第一阶段, 候选种子模块选择单词或双词作为初始种子, 种子增强模块对初始种子进行语义增强, 增强后的种子经过过滤, 过滤掉低质量的种子。扩展模块是通过调整高质量种子的边界来定位实体, 缩小了实

体类型预测的范围，提高候选跨度的质量。在第二阶段，使用提示学习方法，将候选跨度分类任务转化为掩蔽语言建模（MLM）问题。将实体类型预测建模为计算屏蔽位置的词汇概率分布问题。此外，跨度检测模块和实体类型分类模块的单一训练更能够在少样本情况下取得优异的性能。

本文的主要贡献有以下几点：

(1) 为了提高命名实体识别性能，本文提出了一种基于提示学习的两阶段框架TSP-Few，用于少样本命名实体识别，它由跨度检测模块和实体类型分类模块组成。在跨度检测阶段，为了增强初始种子的语义信息，设计了一个种子增强模块。

(2) 为了缓解将少样本命名实体识别分解为跨度检测和实体类型分类所造成的错误累积，本文将负向采样策略纳入到模型中。

(3) 在三个基准数据集上的不同设置下的实验表明，与最先进的方法相比，本文提出的方法的性能取得了明显的改善。

2 相关工作

少样本命名实体识别任务旨在少样本场景下对命名实体跨度进行识别并分类为特定的实体类别。以前关于少样本命名实体识别的研究大多是基于度量学习的。具体地，(Yi and Katiyar, 2020)等人提出Structshot算法，该算法在源域上训练命名实体识别模型作为特征提取器，推理时利用最近邻算法对实体类型分类并使用维特比解码器学习标签的依赖性。(Fritzler et al., 2019)首先将原型网络调整应用于少样本命名实体识别，然后(Tong et al., 2021)等人通过训练分类器学习多个原型表示，用于表示“O”类嘈杂的语义，弥补了现有的原型方法不能区分“O”类词中丰富的语义信息的不足。然而，在少样本场景下令牌级基于度量学习的方法难以充分学习两个不同的任务。

为了解决基于度量学习方法存在的问题，有研究者提出基于跨度的方法用于少样本命名实体识别，(Ma et al., 2022)将命名实体识别任务分解为跨度检测和实体分类，将跨度预测当作序列标注问题并使用MAML算法训练跨度检测器，在实体分类，利用MAML算法增强原型网络，找到好的嵌入空间来更好的分辨不同实体类的跨度。最近，(Wang et al., 2015)等人提出ESD模型，该模型利用多个不同注意力机制对支持集和查询集之间的跨度信息进行交互，并且使用跨度级原型网络，利用查询集跨度对原型进行增强，同时，将“O”类型跨度分为3个子类，以缓解其复杂的语义问题。(Wang et al., 2022)等人在跨度提取阶段，将顺序标签转化为全局边界矩阵，使模型专注于明确的边界信息，并提出基于边缘的损失来缓解两阶段方法中假阳性的问题，并且利用原型学习在语义空间中调整跨度的表示。

随着GPT (Brown et al., 2020)的引入，提示学习被广泛讨论，并在少量学习中显示出巨大的性能(Gao et al., 2021)。最近的一些研究试图引入提示学习，并使其适应少样本命名实体识别。(Cui et al., 2021)等人是第一个提出使用生成式预训练语言模型来解决少样本序列标注问题，通过提示模板列举输入文本的所有候选跨度，模型根据相应的模板分数对每个候选跨度进行分类。(Ma et al., 2021)等人提出了一种无模板的方法，对每个输入只使用一个预测，这个模型减少了预训练和微调之间的差异，并且没有修改预训练模型的输出层。(Hou et al., 2022)等人为了提高基于提示学习预测速度，提出了一种反向的提示范式，在给定槽位标签类型的情况下，反向预测槽位值，而不是列举槽位值来预测标签。

然而，这些研究大多同时学习跨度检测和类型分类任务，这在少样本场景下很难取得优异的表现。此外，这些方法遵循情节训练的方法，或者假设有一个丰富的源域，这不是一个更实用的少样本场景。为了处理上述问题，本文提出了一种基于提示的两阶段少样本命名实体识别框架，它由一个跨度检测模块和一个实体类型分类模块组成。

3 模型

本文提出一种基于提示的两阶段框架TSP-Few，用于少样本命名实体识别（Few-shot NER）。如图1所示，该框架由两个预先训练好的语言模型组成，其中一个用于解决实体跨度检测，输入文本首先经过候选种子模块产生候选种子，然后候选种子进入种子增强模块，对种子进行增强，之后通过计算出的阈值过滤出高质量的候选种子，最后经过种子扩展得到候选跨度。跨度检测模型是与类别无关的，可以在不同的实体类别之间共享信息。另一个用于候选跨度分类，本文将经过扩展得到的候选跨度填入预先定义的模板并送入实体分类模型，该模型缩

小了实体类型预测的范围，旨在通过提示学习为每个跨度分配一个特定的实体类别。下面我们分别讨论每个模块的细节。

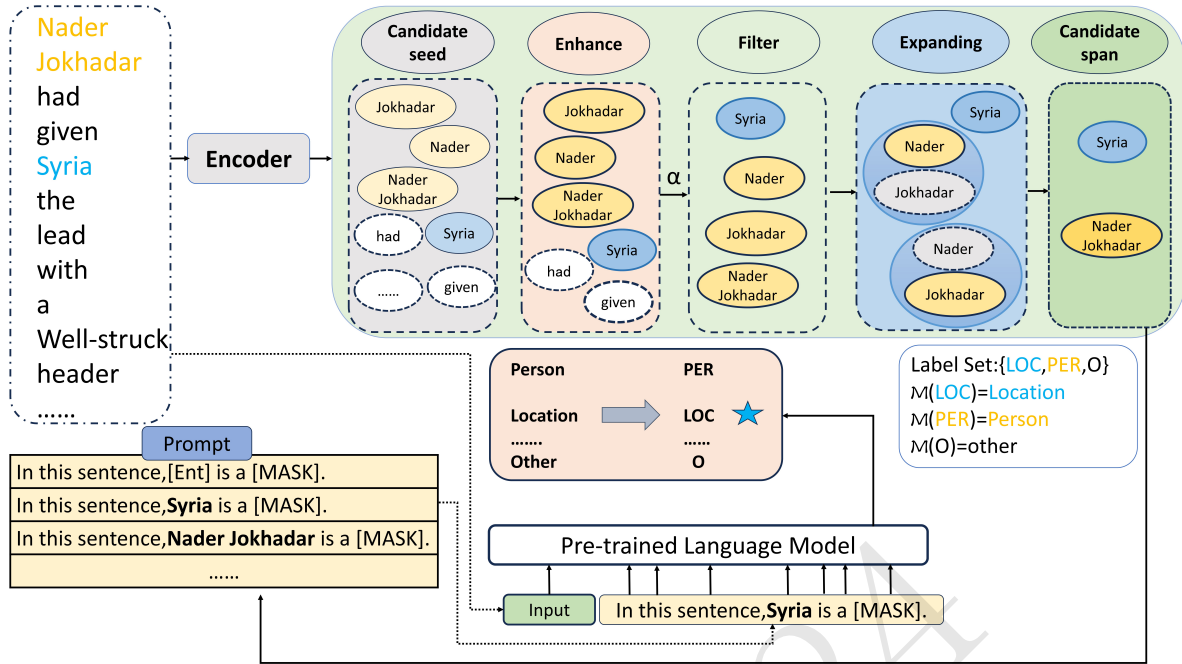


图 1: 模型框架

3.1 问题陈述

本文将命名实体识别任务分解为两个子任务：跨度检测和实体类型分类。给定一个长度为 n 的输入文本 $X = \{x_1, \dots, x_n\}$ ，跨度检测的目的是提取每个候选跨度 $s = [x_i, \dots, x_j] (1 \leq i \leq j \leq n)$ ，并从输入文本中获得一个候选跨度集 $S = \{s_1, \dots, s_m\}$ ，实体类型分类模块的目标是为每个候选跨度 $s \in S$ 分配一个实体类型 $y \in Y$ 。

3.2 第一阶段

3.2.1 产生候选种子

候选种子是从输出文本中抽取的子序列。通过对过滤之后的候选种子进行扩展，能够得到候选跨度。输入文本 $X = \{x_1, \dots, x_n\}$ ，列举出长度 L 不超过2的所有跨度。把得到的候选种子跨度的集合表示为 $C = \{c_1, \dots, c_{2n-1}\}$ ，其中 $c_i = (st_i, ed_i)$ 表示第 i 个候选种子跨度， st_i 、 ed_i 分别表示该跨度的左边界和右边界。这里使用BERT作为编码器，将输入文本送入BERT，获得隐藏表示 $h \in R^{n \times d}$ ，这里 d 是BERT的隐藏层状态的维度。

为了训练候选种子过滤器，筛选出高质量的候选种子，这更有可能扩展为正确的命名实体词，需要计算出候选种子的分数。具体来说，本文将 C 中的每个候选种子跨度和真实实体配对，计算出候选种子分数。计算候选种子分数一般有两种方式：(1) Intersection over Foreground (IoF)，(2) Intersection over Union (IoU)。在命名实体识别中，IoU被定义为 $IoU(A, B) = \frac{A \cap B}{A \cup B}$ ，其中 A 、 B 是两个跨度 (Zheng et al., 2019; Shen et al., 2021)。IoU衡量跨度之间的重叠程度，这对嵌套命名实体识别更有作用。然而，这里很少涉及嵌套命名实体，所以本文使用IoF计算候选种子分数。在命名实体识别中，IoF被定义为 $IoF(A, B) = \frac{A \cap B}{A}$ 。(Yang et al., 2022)考虑到一个由五个词组成的命名实体，“International Conference on Computational Linguistics”，候选种子为“International Conference”与该实体之间的IoU为0.4，并不显著。而该候选种子与实体之间的IoF为1，其中， A 是候选种子， B 是命名实体。这表明该候选种子是实体的一部分，很有可能扩展到整个实体。这里，使用IoF计算候选种子得分更加合适。计算每个候选种子与其真实实体之间的IoF，作为候选种子分数 y^{cw} 。

3.2.2 种子增强模块

本文认为，种子跨度的含义能够从其他种子跨度中推断出来。因此，本文设计了一个种子注意力机制。给出一个句子 $X \in R^{K \times d}$ (K 是跨度的数量) 的所有跨度表示，对于候选种子跨度 $c_i = (st_i, ed_i)$ ， h_i^p 代表跨度 c_i 的平均池化表示。对于种子 c_i ，得到它的增强跨度表示 $\hat{h}_i^p = \sum_{j=1}^K \alpha_i^j c_j$ ，这里 $\alpha_i = \text{softmax}(c_i X^T)$ ，然后，使用带有残差连接和层归一化的前馈神经网络来获得最终的种子增强表示 \hat{h}_i^p ，

$$h_i^p = \text{MeanPooling}(h_{st_i}, \dots, h_{ed_i}) \quad (1)$$

$$\hat{h}_i^p = \text{LayerNorm}(h_i^p + \text{FFN}(\bar{h}_i^p)) \quad (2)$$

$$\text{FFN}(\bar{h}_i^p) = \text{GELU}(\bar{h}_i^p W^1) W^2 \quad (3)$$

其中， W^1 、 W^2 是权重矩阵，这里省略了偏置项。

候选种子过滤器：能够计算出所有候选种子的得分，并设置阈值 α ，选择得分大于 α 的候选种子进行扩展。对于候选种子跨度 $c_i = (st_i, ed_i)$ ， \hat{h}_i^p 代表经过种子增强的跨度表示，连接 \hat{h}_i^p 和 [CLS] 标记的 h^{CLS} 表示作为候选种子跨度的最终表示 h_i^{cw} 。候选种子分数 p_i^{cw} 的计算方法如下：

$$h_i^{cw} = \text{Concat}(\hat{h}_i^p, h^{[CLS]}) \quad (4)$$

$$p_i^{cw} = \text{Sigmoid}(\text{MLP}(h_i^{cw})) \quad (5)$$

这里 MLP 表示多层感知机，最后一层使用 GELU 激活函数。

3.2.3 候选种子扩展

尽管通过种子过滤器得到了质量较高的候选种子，但这些种子还不能够精准的定位命名实体，还需要对候选种子边界进行调整，这是一项回归任务。为了学习边界偏移量 δ 不仅需要跨度本身的信息，也需要外部边界词的信息。因此，扩展还需要考虑候选种子周围的一个窗口。

扩展是允许候选种子的左右边界分别偏移 λ ，这样能够得到最长的实体的长度是 $2 + 2\lambda$ ，对于候选种子 c_i ，最大扩展为：

$$b_i^{max} = (\max(1, st_i - \lambda), \min(n, ed_i + \lambda)) \quad (6)$$

为了能够区分最大扩展的边界，窗口应该大于 b_i^{max} ，定义为 w_i 。

$$w_i^l = \max(1, st_i - 2\lambda) \quad (7)$$

$$w_i^r = \min(n, ed_i + 2\lambda) \quad (8)$$

$$w_i = (w_i^l, w_i^r) \quad (9)$$

其中， w_i^l 、 w_i^r 分别表示窗口的左右边界。

将候选种子 c_i 的平均集合跨度 h_i^p 和窗口的平均集合跨度 h_i^w 连接起来，然后计算边界偏移量 o_i ，计算如下：

$$h_i^w = \text{MeanPooling}(h_{w_i^l}, \dots, h_{w_i^r}) \quad (10)$$

$$o_i = \lambda \cdot (2 \cdot \text{Sigmoid}(\text{MLP}(h_i^p; h_i^w)) - 1) \quad (11)$$

其中， $(;)$ 表示连接操作。 o_i 的第一个元素表示为 o_i^{st} ，第二个元素表示为 o_i^{ed} 。 o_i^{st} 表示为左边界偏移量， o_i^{ed} 表示右边界偏移量。这样，就可以得到新的左右边界的候选跨度。

$$l_i' = \max(1, st_i + \left\lfloor o_i^{st} + \frac{1}{2} \right\rfloor) \quad (12)$$

$$r_i' = \max(n, ed_i + \left\lfloor o_i^{ed} + \frac{1}{2} \right\rfloor) \quad (13)$$

其中 l_i' 、 r_i' 分别表示新的左右边界。重复的结果和无效的结果 $l_i' > r_i'$ 会被丢弃。这样，产生了一组候选跨度，用于跨度分类。

3.2.4 训练目标

产生候选种子损失 L_{cw} 和候选种子扩展损失 L_{exp} 定义为:

$$L_{cw} = \sum_i SmoothL1(y_i^{cw}, p_i^{cw}) \quad (14)$$

$$L_{exp} = \sum_i \sum_{j \in \{st, ed\}} SmoothL1(\hat{o}_i^y, o_i^j) \quad (15)$$

其中, y_i^{cw} 是真实的候选种子分数, \hat{o}_i^y 是原始边界偏移量。通过最小化综合损失 L_z 进行训练, 定义如下:

$$L_z = L_{cw} + L_{exp} \quad (16)$$

3.3 第二阶段

在实体类型分类模块中, 预测由上述跨度检测模块从候选种子扩展得到的候选跨度集 $S = \{s_1, \dots, s_m\}$ 中每个候选跨度的实体类别。为避免引入新的参数并减少预训练和微调之间的差距, 本文遵循基于提示的调整, 将分类任务转化为掩蔽语言建模 (MLM) 问题。基于提示的方法包括构建言语器和模板。在接下来的章节中, 将详细介绍实体类型预测模块的言语器和模板的构建。

3.3.1 言语器设计

言语器的作用是将标签词汇映射到一个特殊的类别, 这有助于预训练语言模型理解每个类别。为了丰富实体类别的内涵, 实体类型分类模块的言语器设计是将每个实体类别 $y \in Y$ 映射到一个词 ν_y , 即 $M: y \rightarrow \nu$ 。例如, 标签集合 $Y = \{PER, LOC, ORG, O\}$ 被转换为相应的词 $\nu = \{person, location, organization, other\}$ 。

3.3.2 模板设计

为了统一预训练任务和微调任务的形式, 提示模板将原始输入 x 修改为提示输入 $T(x)$, 其中包括[MASK]标记。在本文的框架中, 用自然语言构建一个模板 $T(x)$, 如下所示:

$$T(x) = x.In\ this\ sentence,\ [Ent]\ is\ a\ [MASK]. \quad (17)$$

其中 x 代表原始句子输入, $[Ent]$ 表示句子 x 中的候选实体跨度。因此, 计算标签 $y \in Y$ 概率的实体类型预测任务被建模为计算被屏蔽位置的词汇的概率分布, 其形式化为:

$$p(y|x) = p([MASK] = M(y)|T(x)) \quad (18)$$

3.3.3 训练和推理

为了缓解两阶段框架带来的误差累积问题, 本文采用了负向采样策略, 在训练过程中对训练数据进行采样。具体来说, 随机抽取非实体的跨度作为实体类型分类的负面样本, 并利用“O”来表示负面样本 $O \in Y$ 的类型。采用负面抽样策略使实体分类模块有能力预测非实体类别。当跨度检测模块出现错误时, 实体分类模块能够将错误跨度分配“other”, 从而缓解了错误的积累。此外, 在实体分类模块中, 本文选择BERT作为预训练模型, 并利用交叉训练损失作为目标函数来微调预训练模型。该函数的形式如下:

$$\mathcal{L} = - \sum \log p(y|x) \quad (19)$$

最后, 将跨度检测模块得到的候选跨度填入 $T(x)$ 中的 $[Ent]$, 并将其送入微调后的预训练模型, 在推理过程中预测跨度的实体类型。

4 实验设置

4.1 少样本设置

与之前假设有源域数据可用的少样本命名实体识别不同，本文考虑了没有源域训练的设置，这更具有挑战性。这个实际场景只有一个测试集，每个类别只有 N 个标签（ N -way）的 K 个例子（ K -shot）。为了更好的评估基线和本文提出的模型的性能，本文用 $K \in \{5, 10, 20\}$ 进行了实验。

4.2 数据集的构建

为了给所提出的模型提供有效性的经验证据和为了进行公平的比较，本文在三个数据集上进行了实验。在 $K \in \{5, 10, 20\}$ 的情况下，本文完全遵循 (Yang and Katiyar, 2020) 中的贪心抽样策略。具体来说，(1) 训练集中的每个类别至少有 K 个对应的命名实体。(2) 从训练集中删除任何句子都会导致至少有一个类别的命名实体少于 K 。为了保证实验的有效性和可靠性，所使用的三个数据集中的抽样数据是由 (Yang et al., 2022) 所提供公开的。本文使用的三个不同领域的命名实体识别数据集分别是：CoNLL03数据集 (Sang and Meulder, 2003) 在新闻领域、MIT-Restaurant数据集 (Liu et al., 2013) 在评论领域和WikiGold数据集 (Balasuriya et al., 2009) 一般领域。表1显示了这些数据集的原始信息。

Dataset	Domain	Language	Class	Train	Test
CoNLL03	News	English	4	14,987	3,684
MIT-Restaurant	Review	English	8	7,660	1,521
WikiGold	General	English	4	1,017	339

表 1: 原始数据集的统计信息

4.3 实施细节

本文使用PyTorch (Paszke et al., 2019) 和Openprompt (Paszke et al., 2022) 实现模型。在跨度检测模块，英文数据集使用“bert-base-uncased”预训练模型作为编码器，epochs设置为35，CoNLL03、MIT-Restaurant和WikiGold数据集的阈值 α 分别设置为0.5, 0.6, 0.7。 λ 设置为5。采用AdamW优化器，学习率为 $3e-0.5$ 。对于实体类型分类模块，本文使用“bert-base-uncased”预训练模型作为编码器，并设置epochs的数量为30，该模块使用AdamW作为优化器，学习率为 $5e-5$ 。对于所有的基线，本文使用原始论文或者官方实现提供的推荐参数。

4.4 基线

本文将提出的模型与六个竞争性基线进行比较。

Prototype (Snell et al., 2017) 一种基于原型网络的方法，将实体类别表示为单个标记的同一表示空间中的向量，并利用最近的邻居标准来分配实体类别。

LC-BERT (Devlin et al., 2019) 一个基于BERT的序列标签模型，它直接在BERT模型中加入一个线性分类器，用于序列标签。

NNShot和StructShot (Yang and Katiyar, 2020) 提出了一个基于度量的少样本学习方法NNShot用于命名实体识别，它采用最近邻分类器进行预测。在此基础上，他们进一步提出了在解码过程中引入维特比算法的少样本学习方法StructShot。

TemplateNER (Cui et al., 2021) 一种基于模板的提示学习方法，用于少样本命名实体识别，它在推理过程中列举所有的跨度来生成预定义的模板作为BART的输入。

SEE-Few (Yang et al., 2022) 将命名实体识别分为两部分，跨度检测和跨度分类。在跨度检测模块生成种子，直接经过筛选，再对种子扩展为候选跨度。对于跨度分类模块，它将跨度分类重新定义为文本蕴含任务，同时利用上下文和实体类型信息进行类型预测。

5 实验结果

5.1 结果分析

表2显示了本文所提出的方法和基线在CoNLL03、MIT-Restaurant和WikiGold数据集

上不同少样本设置中的表现，有以下重要观察：1) 本文的模型在上述三个数据集上表现出优越性。具体来说，在K=5样本设置下，本文提出的模型的F1分数在CoNLL03、MIT-Restaurant和WikiGold比基线模型分别增加了2.27%、12.75%、6.77%；在K=10样本设置下分别增加了：6.91%、13.24%、3.94%，这验证了本文提出的模型在少样本设置下表现更加优异，对少样本数据下的情况更加适应。2) 与基线相比，本文提出的模型能够在5样本设置下取得比基线在20样本设置下表现相当甚至更优的性能表现。3) 本文的方法优于以前的提示方法，与TemplateNER模型在不同样本设置对比下更加明显。4) 在本文的少样本设置中，优于缺乏丰富的源域数据进行预训练，基于度量的方法NNShot和StructShot并没有显示出它们的优势，这些结果表明这些方法不适合更实际的少样本场景。

Datasets	Methods	K=5			K=10			K=20		
		P	R	F1	P	R	F1	P	R	F1
CoNLL03	LC-BERT	42.83	30.72	35.06 _(6.09)	50.36	52.13	51.20 _(6.39)	56.33	63.85	59.84 _(1.43)
	Prototype	38.26	43.14	40.37 _(8.06)	45.08	64.02	52.82 _(3.22)	43.94	69.72	53.89 _(1.95)
	NNShot	32.11	38.42	34.92 _(3.30)	34.10	40.98	37.18 _(5.82)	38.43	47.85	42.61 _(2.23)
	StructShot	30.04	21.33	23.43 _(4.52)	38.62	19.72	26.09 _(7.23)	44.96	28.59	34.87 _(1.30)
	TemplateNER	26.90	23.46	23.13 _(8.40)	44.51	43.99	44.01 _(4.82)	52.16	56.46	54.01 _(5.09)
	SEE-Few	60.45	51.27	55.21 _(3.93)	66.19	58.68	61.99 _(1.73)	69.49	67.07	68.21 _(2.60)
	Ours	63.63	52.42	57.48 _(5.10)	70.82	67.08	68.90 _(1.94)	76.89	76.59	76.74 _(3.17)
MIT-Restaurant	LC-BERT	41.21	38.65	39.88 _(3.79)	43.60	48.93	46.08 _(3.75)	56.24	60.04	58.07 _(1.50)
	Prototype	27.77	46.79	34.84 _(1.63)	30.37	50.64	37.97 _(2.29)	37.91	59.31	46.25 _(1.62)
	NNShot	28.15	34.81	31.11 _(2.30)	30.28	37.56	33.56 _(1.48)	36.72	45.55	40.66 _(1.26)
	StructShot	45.13	25.00	31.93 _(4.32)	43.94	28.19	34.30 _(2.56)	52.08	36.18	42.69 _(1.12)
	TemplateNER	23.11	20.78	21.53 _(4.66)	39.45	28.77	32.71 _(8.14)	46.93	37.00	41.26 _(6.80)
	SEE-Few	53.08	39.47	45.25 _(3.18)	57.19	46.41	51.20 _(1.48)	64.79	57.22	60.75 _(2.07)
	Ours	71.34	49.85	58.69 _(3.44)	74.61	56.71	64.44 _(2.15)	76.17	68.16	71.94 _(1.88)
WikiGold	LC-BERT	36.02	8.02	12.57 _(7.81)	43.13	8.95	37.72 _(7.20)	50.68	50.73	50.68 _(5.94)
	Prototype	20.55	21.46	19.28 _(8.12)	23.31	45.21	30.59 _(3.95)	27.31	56.22	36.56 _(8.65)
	NNShot	27.81	34.16	30.63 _(1.91)	26.36	37.92	30.93 _(4.89)	28.33	39.07	32.81 _(5.41)
	StructShot	49.00	13.37	20.88 _(4.61)	43.21	14.19	21.28 _(2.96)	43.51	15.94	23.16 _(2.18)
	TemplateNER	18.45	19.45	17.26 _(12.73)	38.33	45.37	41.04 _(13.19)	57.39	56.00	56.00 _(3.22)
	SEE-Few	61.23	41.01	48.87 _(8.01)	63.36	48.74	54.98 _(3.24)	69.06	58.25	63.19 _(1.28)
	Ours	74.94	44.24	55.64 _(5.84)	67.19	58.46	58.92 _(5.80)	76.58	64.52	70.03 _(2.65)

表 2: 三个数据集上不同场景下的实验结果

5.2 消融实验

为进一步验证所提方法中不同组件的有效性，本文用不同的少样本设置进行了消融实验。本文构建了多个消融实验变体：

- 1) w/o sea: 移除跨度检测模块中的种子增强模块，直接将种子进行过滤；
- 2) w/o neg: 移除跨度分类模块中的负向采样策略；
- 3) w/o span: 移除跨度检测模块，在CoNLL03、MIT-Restaurant和WikiGold数据集上列举长度L=5的所有可能跨度，并将其填入实体分类模块；
- 4) w/o prompt: 移除提示学习模块，直接使用基线SEE-Few模型的跨度分类方法，将分类任务定义为文本蕴含任务。

Model	CoNLL03			MIT-Restaurant			WikiGold		
	5	10	20	5	10	20	5	10	20
Full model	57.48	68.90	76.74	58.98	64.57	71.94	55.64	58.92	70.03
w/o sea	54.72	65.06	75.50	58.21	60.84	70.67	52.06	56.30	68.39
w/o neg	54.98	67.10	76.24	57.91	64.26	71.18	54.78	57.69	69.73
w/o span	38.14	49.85	55.25	35.71	42.44	38.37	27.12	17.80	20.12
w/o prompt	56.05	55.88	68.42	49.92	51.86	59.49	50.30	45.85	66.77

表 3: 在不同的样本设置下，模型消融实验结果在三个数据集上的F1得分

本文对消融实验采用F1值作为实验结果，如表3所示。由实验结果可知：

Model	CoNLL03	MIT-Restaurant	WikiGold
Full model	5648	3151	730
w/o span	199068	56139	32073

表 4: 测试集上候选跨度的数量统计

(1) 对于移除种子增强模块，直接将种子进行过滤的设置。这意味着种子跨度的含义不能从其他种子跨度中推断出来，实验结果表明，在CoNLL03、MIT-Restaurant、WikiGold数据集上5样本设置下，模型的性能F1分数分别下降了2.76%、0.77%、3.58%，这表明，种子增强模块能够让特定的种子跨度从其他种子跨度推断出来，证明种子增强模块是有效的。

(2) 对于移除负向采样策略，模型不能够学习到“other”类别的信息，从实验结果来看，在所有的数据集和少样本设置中，取消负向采样策略后，F1分数都在下降。这表明，利用负向采样有助于缓解错误累积，能够识别出跨度检测阶段识别出的虚假跨度。

(3) 对于移除跨度检测模块，直接枚举跨度，进行分类。可以看到，与移除负采样相比，跨度检测模块对少样本命名实体识别有更大的影响，直接枚举跨度导致模型性能急剧下降。从表4的实验结果分析，这是因为直接枚举跨度会产生数量巨大的无关跨度，这些杂乱的语义信息对模型来说是灾难性的，这是导致模型性能下降的直接原因。这表明，跨度检测模块能够极大的减少大量的无关跨度，这对模型性能很重要，跨度检测模块有能力缩小命名实体识别的范围并提高性能。

(4) 对于移除提示学习模块，将跨度分类任务定义为文本蕴含任务。从实验结果来看，F1分数都有明显下降，这说明利用提示学习进行实体类型分类，模型能够更好的充分利用上下文的信息，能够更准确的识别实体类型。除此之外，模型在K=10的样本设置下，性能下降的更快，根据实验结果分析，在K=10的场景中，在第一阶段中的种子增强模块，可能给种子跨度引入了噪声，并没有对所有的种子跨度都有增强效果，反而影响最终的模型性能。

5.3 案例研究

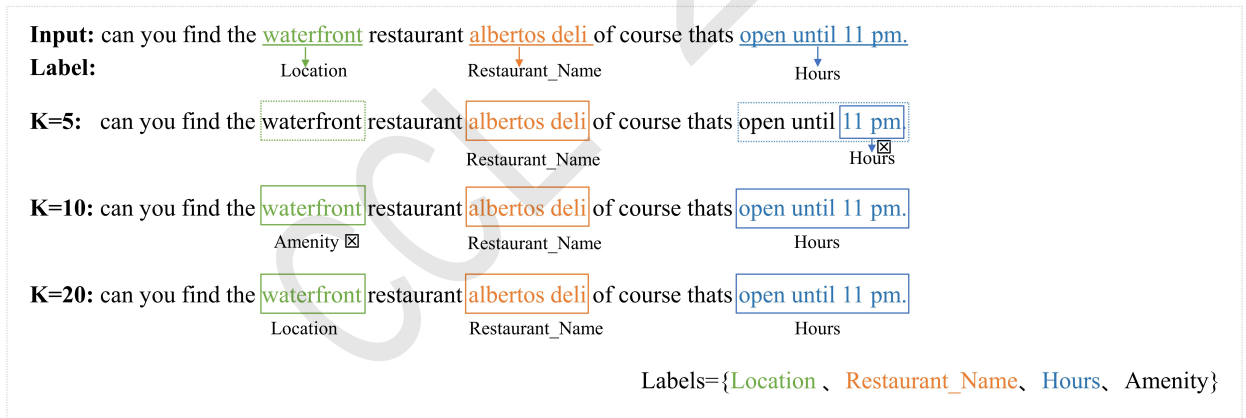


图 2: 案例研究。在数据集MIT-Restaurant下不同样本场景中进行的案例分析。实线框表示模型识别到的实体跨度，虚线框表示模型未识别到的实体跨度。

图2展示了在不同样本设置下，模型预测的例子。我们可以观察到在K=5样本设置下，模型只正确检测出实体跨度“albertos deli”，而模型没有检测出实体跨度“waterfront”，并且模型检测出的错误跨度“11 pm”只是真实实体跨度“open until 11 pm”的一部分，没有扩展成完整的实体跨度，原因是：在K=5样本设置下，训练样本信息太少，对种子扩展模块还不足以充分的训练。在K=10样本设置下，模型正确检测出实体跨度“waterfront”和“open until 11 pm”，然而，模型虽然能检测出在K=5样本中未识别出的跨度，但是，在对实体跨度“waterfront”进行分类时，模型分配了错误的实体类型“Amenity”，而正确的实体类型是“Location”。我们分析认为是模型对实体类型的信息捕获不足及对上下文信息的利用还不够充分。此外，模型能够准确检测

出实体跨度“open until 11 pm”，这表明模型具备将种子跨度扩展到完整实体跨度的能力。最后，在K=20样本设置下，模型不仅能够准确检测出实体跨度还能够为实体跨度分配准确的实体类型。这说明模型在训练样本数增加下，能够学习到更多的知识，取得较好的性能。

6 总结与未来工作

在本文中，提出一个两阶段框架TSP-Few，用于不使用源域数据的少样本命名实体识别。本文将少样本命名实体识别任务分解为跨度检测和实体分类两个阶段，这可以使模型在少样本情况下取得更出色的性能。种子增强模块和扩展模块对实体分类模块提供尽可能准确的候选跨度。实体分类模块将分类任务转化为屏蔽语言建模任务，将实体类型分配给候选跨度，通过减少预训练和微调之间的差距来提高少样本性能。在三个基准数据集上的大量实验表明，本文提出的模型在少样本命名实体识别基准上取得了有竞争力的结果。

在未来的工作中，我们将探索把提示学习与对比学习结合起来，以提高PLM的性能，并进一步提高少样本命名实体识别的性能。

参考文献

- Dominic Balasuriya, Nicky Ringland, Joel Nothman, Tara Murphy, and James R. Curran. 2009. Named Entity Recognition in Wikipedia. In *Proceedings of the 2009 Workshop on The People’s Web Meets NLP: Collaboratively Constructed Semantic Resources (People’s Web)*, pages 10–18.
- 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, 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. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS ’20)*, pages 1877–1901.
- Jason P.C. Chiu and Eric Nichols. 2016. Named Entity Recognition with Bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, pages 4:357–370.
- Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-Based Named Entity Recognition Using BART. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1835–1845.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *North American Chapter of the Association for Computational Linguistics(2019)*.
- Ning Ding, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Haitao Zheng, and Maosong Sun. 2022. OpenPrompt: An Open-source Framework for Prompt-learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 105–113.
- Alexander Fritzier, Varvara Logacheva, and Maksim KretoV. 2019. Few-shot classification in named entity recognition task. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing (SAC ’19)*, pages 993–1000.
- Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making Pre-trained Language Models Better Few-shot Learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830.
- Jiaxin Huang, Chunyuan Li, Krishan Subudhi, Damien Jose, Shobana Balakrishnan, Weizhu Chen, Baolin Peng, Jianfeng Gao and Jiawei Han. 2020. Few-Shot Named Entity Recognition: A Comprehensive Study. *arXiv preprint arXiv:2012.14978 (2020)*.
- Yutai Hou, Cheng Chen, Xianzhen Luo and Bohan Li. 2022. Inverse is better! fast and accurate prompt for few-shot slot tagging. *arXiv preprint arXiv:2204.00885 (2022)*.

- Yutai Hou, Wanxiang Che, Yongkui Lai, Zhihan Zhou, Yijia Liu, Han Liu, and Ting Liu. 2020. Few-shot Slot Tagging with Collapsed Dependency Transfer and Label-enhanced Task-adaptive Projection Network. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1381–1393.
- Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 260–270.
- Jingjing Liu, Panupong Pasupat, D. Scott Cyphers and James R. Glass. 2013. Asgard: A portable architecture for multilingual dialogue systems. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing(2013)*, pages 8386-8390.
- Jie Ma, Miguel Ballesteros, Srikanth Doss, Rishita Anubhai, Sunil Mallya, Yaser Al-Onaizan, and Dan Roth. 2022. Label Semantics for Few Shot Named Entity Recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1956–1971.
- Ruotian Ma, Xin Zhou, Tao Gui, Yi Chern Tan, Qi Zhang and Xuanjing Huang. 2021. Template-free Prompt Tuning for Few-shot NER. *arXiv preprint arXiv:2109.13532 (2021)*.
- Tingting Ma, Huiqiang Jiang, Qianhui Wu, Tiejun Zhao, and Chin-Yew Lin. 2022. Decomposed Meta-Learning for Few-Shot Named Entity Recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1584–1596.
- Xuezhe Ma and Eduard Hovy. 2016. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074.
- Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in neural information processing systems 32 (2019)*.
- Matthew E. Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Powe. 2017. Semi-supervised sequence tagging with bidirectional language models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1756–1765.
- Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.
- Mohammad Golam Sohrab and Makoto Miwa. 2021. Deep Exhaustive Model for Nested Named Entity Recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2843–2849.
- Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*, pages 4080–4090.
- Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*, pages 4080–4090.
- Timo Schick and Hinrich Schütze. 2021. Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 255–269.
- Yongliang Shen, Xinyin Ma, Zeqi Tan, Shuai Zhang, Wen Wang, and Weiming Lu. 2021. Locate and Label: A Two-stage Identifier for Nested Named Entity Recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2782–2794.
- Meihan Tong, Shuai Wang, Bin Xu, Yixin Cao, Minghui Liu, Lei Hou and Juan-Zi Li. 2021. Learning from Miscellaneous Other-Class Words for Few-shot Named Entity Recognition. *Annual Meeting of the Association for Computational Linguistics(2021)*.

- Jianing Wang, Chengyu Wang, Chuanqi Tan, Minghui Qiu, Songfang Huang, Jun Huang, and Ming Gao. 2022. SpanProto: A Two-stage Span-based Prototypical Network for Few-shot Named Entity Recognition. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3466–3476.
- Peiyi Wang, Runxin Xu, Tianyu Liu, Qingyu Zhou, Yunbo Cao, Baobao Chang, and Zhifang Sui 2015. An Enhanced Span-based Decomposition Method for Few-Shot Sequence Labeling. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5012–5024.
- Yang, Yi, and Arzoo Katiyar. 2020. Simple and Effective Few-Shot Named Entity Recognition with Structured Nearest Neighbor Learning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2540–2550.
- Yi Yang and Arzoo Katiyar. 2020. Simple and Effective Few-Shot Named Entity Recognition with Structured Nearest Neighbor Learning. *arXiv preprint arXiv:2010.02405 (2020)*.
- Zeng Yang, Linhai Zhang, and Deyu Zhou. 2022. SEE-Few: Seed, Expand and Entail for Few-shot Named Entity Recognition. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2540–2550.
- Changmeng Zheng, Yi Cai, Jingyun Xu, Ho-fung Leung, and Guandong Xu. 2019. A Boundary-aware Neural Model for Nested Named Entity Recognition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 357–366.