

# 基于跨度边界感知的嵌套命名实体识别\*

蔡宇翔<sup>1</sup>, 骆 姐<sup>1</sup>, 甘洋镭<sup>1</sup>, 侯 睿<sup>1</sup>, 刘雪怡<sup>1</sup>, 刘 峤<sup>1</sup>, 石晓军<sup>2</sup>

<sup>1</sup>(电子科技大学 计算机科学与工程学院, 四川 成都 611731)

<sup>2</sup>(中国电子科技集团公司 信息科学研究所, 北京 100086)

通信作者: 刘峤, E-mail: [qliu@uestc.edu.cn](mailto:qliu@uestc.edu.cn)



**摘 要:** 命名实体识别任务是信息抽取领域中的一个基础任务, 旨在定位句子中实体所在位置的边界, 并对该实体进行分类. 针对现有基于跨度检测的模型存在的嵌套实体边界模糊问题, 提出一种基于跨度边界感知的嵌套命名实体识别模型. 首先, 利用双向仿射注意力机制, 捕获词元间的语义相关性, 进而生成跨度语义表示矩阵; 其次, 通过设计一种二阶对角邻域差分算子, 建立跨度语义差分机制, 以提取跨度间的语义差异信息. 此外, 引入一种跨度边界感知机制, 利用滑动窗口的局部特征提取能力, 强化跨度的边界语义差异, 从而准确定位实体跨度位置. 为验证模型的有效性, 在 3 个基准数据集上进行测试, 包括 ACE04、ACE05 和 Genia 数据集. 实验结果表明, 提出的模型在实体识别准确率的表现优于相关工作. 此外, 还设计消融实验和案例分析以验证提出的语义差分机制和跨度边界感知机制的有效性, 为进一步研究命名实体识别问题提供新的思路和实验证据.

**关键词:** 嵌套命名实体识别; 跨度分类; 差分算子; 双向仿射注意力机制; 跨度语义感知

**中图法分类号:** TP391

中文引用格式: 蔡宇翔, 骆姐, 甘洋镭, 侯睿, 刘雪怡, 刘峤, 石晓军. 基于跨度边界感知的嵌套命名实体识别. 软件学报, 2024, 35(11): 5149–5162. <http://www.jos.org.cn/1000-9825/7040.htm>

英文引用格式: Cai YX, Luo D, Gan YL, Hou R, Liu XY, Liu Q, Shi XJ. Nested Named Entity Recognition Based on Span Boundary Perception. Ruan Jian Xue Bao/Journal of Software, 2024, 35(11): 5149–5162 (in Chinese). <http://www.jos.org.cn/1000-9825/7040.htm>

## Nested Named Entity Recognition Based on Span Boundary Perception

CAI Yu-Xiang<sup>1</sup>, LUO Da<sup>1</sup>, GAN Yang-Lei<sup>1</sup>, HOU Rui<sup>1</sup>, LIU Xue-Yi<sup>1</sup>, LIU Qiao<sup>1</sup>, SHI Xiao-Jun<sup>2</sup>

<sup>1</sup>(School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China)

<sup>2</sup>(Institute of Information Science, China Electronics Technology Group Corporation, Beijing 100086, China)

**Abstract:** Named entity recognition (NER) is a fundamental task in information extraction and aims to locate the boundaries of entities in a sentence and classify them. In response to the fuzzy boundaries of nested entities based on span detection models, this study proposes a nested NER model based on span boundary perception. Firstly, it utilizes a bidirectional affine attention mechanism to capture the semantic relevance among word tokens and then generates a span semantic representation matrix. Secondly, it designs a second-order diagonal neighborhood difference operator and establishes a span semantic difference mechanism to extract semantic difference information among spans. Additionally, a span boundary perception mechanism is introduced to employ the local feature extraction ability of sliding windows to enhance the span boundary semantic differences, thereby accurately locating the entity span. The model is validated on three benchmark datasets of ACE04, ACE05, and Genia. The experimental results show that the proposed model outperforms related work in entity recognition accuracy. Additionally, the study conducts ablation experiments and case studies to verify the effectiveness of the proposed semantic difference mechanism and span boundary perception mechanism, providing new ideas and empirical evidence for further research on NER.

**Key words:** nested named entity recognition; span classification; difference operator; bidirectional affine attention mechanism; span semantic cognition

\* 基金项目: 国家自然科学基金 (U19B2028, U22B2061); 国家重点研发计划 (2022YFB4300603); 四川省科技计划 (2023YFG0151)

蔡宇翔和骆姐为共同第一作者.

收稿时间: 2023-04-04; 修改时间: 2023-05-29; 采用时间: 2023-08-24; jos 在线出版时间: 2024-01-10

CNKI 网络首发时间: 2024-01-12

信息抽取中命名实体识别 (named entity recognition, NER) 任务, 是指通过计算机辅助手段, 识别文本中出现的实体对象, 如人名、地名、组织机构名、日期等, 从而提取出文本中所包含的重要信息<sup>[1]</sup>, 为下游关系抽取<sup>[2-4]</sup>、事件抽取<sup>[5,6]</sup>、问答系统<sup>[7,8]</sup>等任务提供数据基础. 然而, 根据 Finkel 等人<sup>[9]</sup>的研究表明, 命名实体通常以嵌套的形式存在于句子中. 嵌套的形式会使句子中的实体结构更复杂, 从而增加了机器理解实体的难度. 因此, 嵌套命名实体识别 (nested named entity recognition) 是 NER 任务面临的挑战之一.

本文研究嵌套命名实体识别的实体边界感知和类型识别任务, 即针对给定语句  $X$ , 感知该语句中所有命名实体位置的左/右边界 ( $s_i, e_i$ ), 并识别出每个命名实体的特定类型 ( $t_i$ ). 例如:

[The minister of [the department of education]<sub>1</sub>]<sub>2</sub> convened a meeting.

该语句摘自 ACE05 数据集, 其中包含两个嵌套的命名实体: the minister of the department of education 和 the department of education, 分别被识别为 PER 和 ORG 类型, 即人名和组织机构名.

随着深度学习的发展, 基于神经网络的语言模型被广泛应用于命名实体识别任务. 早期的工作采用序列标注的方法<sup>[10,11]</sup>, 将命名实体识别问题视为序列标注问题, 并利用句法结构信息来提高命名实体识别准确率. 然而, 由于嵌套实体在基准数据集中普遍存在 (ACE04 和 ACE05 数据集中有 30% 的语句存在嵌套实体)<sup>[9]</sup>, 导致传统的序列标注方法在处理嵌套实体时性能不稳定.

当前主流的嵌套命名实体识别建模方法主要有以下 4 种: 基于分层的方法、基于序列生成的方法、基于超图神经网络的方法以及基于跨度枚举的方法. 其中, 基于分层的方法<sup>[12-15]</sup>通过在层级结构中嵌入多层实体信息, 实现多级别的嵌套命名实体识别. 基于序列生成的方法<sup>[16,17]</sup>将嵌套命名实体识别视为序列生成任务, 并采用基于 Transformer 的模型结构, 逐个预测每个位置的实体标签. 基于超图神经网络的方法<sup>[18-20]</sup>通过在超图中编码句子中所有可能的实体提及和它们的嵌套关系, 实现嵌套命名实体的识别. 基于跨度枚举的方法<sup>[21-23]</sup>则直接枚举文本中所有可能的实体跨度, 根据语义信息预测各跨度的实体标签, 实现嵌套命名实体的识别.

当前的跨度嵌套命名实体识别方法通常采用穷举跨度并采用简单的聚合策略 (如池化或拼接头尾词元) 来构建跨度表示<sup>[22-24]</sup>. 然而, 由于嵌套实体之间存在词元高度重叠, 甚至共享相同头尾词元的现象, 导致现有方法生成的嵌套实体表示之间相似度较高, 使得分类器难以直接捕捉跨度之间的语义差异<sup>[22]</sup>, 从而引发嵌套实体边界模糊问题. 例如, 在图 1 所示的句子 1 中, 跨度“the Alabama-Florida line”“Florida”“Alabama”都是实体, 然而跨度“Alabama-Florida line”并非实体. 由于“Alabama-Florida”和“the Alabama-Florida line”共享大部分词元, 导致实体与非实体跨度在语义上非常接近. 同样, 在图 1 的句子 2 中, 由于实体“a man with a gun”与嵌套实体“a gun”共享头尾词元“gun”, 导致不同类型的命名实体语义相似度增加. 因此, 如果直接将跨度语义输入分类器, 将增加命名实体的边界感知和分类的困难性.

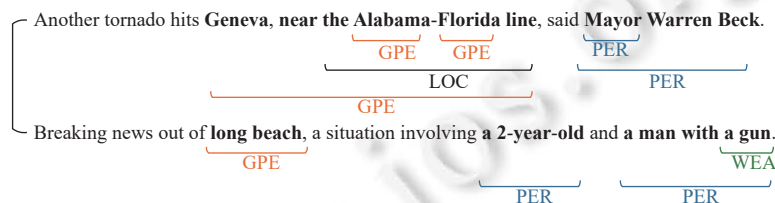


图 1 ACE 语料库中两个含嵌套实体的示例句子

为解决上述问题, 本文提出了一种基于跨度边界感知的嵌套命名实体识别模型. 模型将跨度间的语义差异作为显式输入特征. 通过梯度反向传播, 模型能够学习合适的内部表征<sup>[25]</sup>, 从而增大嵌套实体在跨度语义表示空间中的距离, 进而改善和提升嵌套命名实体的边界感知和分类准确率. 具体来说, 我们利用双仿射模型 (biaffine model) 生成跨度语义表示矩阵, 并设计一组二阶对角邻域差分算子来提取跨度间的语义差异信息. 同时, 我们提出了一种基于滑动卷积核 (convolution kernel) 的跨度边界感知机制, 扩大跨度间语义差异信息的感知范围, 从而能更准确地定位实体跨度位置索引. 最后, 使用线性层预测命名实体类型. 本文的主要贡献如下.

1) 本文提出了一种基于跨度边界感知的嵌套命名实体识别模型, 该模型能够有效建模嵌套实体边界的语义差异, 进而提升嵌套命名实体识别的准确率。

2) 本文提出了一种嵌套命名实体识别建模新范式, 通过引入跨度语义差分机制和跨度边界感知机制, 能够从嵌套实体中提取出语义差异, 从而有效克服嵌套实体间语义边界模糊的问题。

3) 本文使用了命名实体识别领域广泛使用的 ACE04, ACE05 以及 Genia 这 3 个数据集对模型性能进行验证, 并与当前主流的相关工作进行了性能比较。实验结果表明, 所提出的基于跨度边界感知的嵌套命名实体识别模型的识别准确率均优于相关工作。

## 1 相关工作

随着文本信息数量的快速增长, 命名实体识别已经成为自然语言处理领域中非常重要的任务之一。在传统的研究中, 命名实体识别通常被建模为序列标注任务, 使用序列模型来输出具有最大概率的标签序列, 然而, 传统的序列标注模型无法处理嵌套结构。嵌套命名实体识别作为命名实体识别的一种扩展形式, 旨在将由两个或多个相互嵌套的命名实体识别出来。因此, 嵌套命名实体识别是 NER 的一个更加复杂和具有挑战性的任务。本文针对近年来提出的不同方法, 按照不同的建模方式分为 4 类, 逐一进行分析和对比。

第 1 种是基于分层的方法。基于分层的方法旨在识别多层嵌套的命名实体, 通过自上而下的方式逐层标注实体。这种方法使用了一个特殊的层次结构, 将实体分为不同的层次, 每深一层表示更高一层次的实体。Wang 等人<sup>[13]</sup>设计了一种金字塔结构的序列标注框架, 通过卷积神经网络自下而上地抽取实体。Shibuya 等人<sup>[14]</sup>探索了次优路径解码方式, 以从外到内的方式从最外层到内层递进地提取实体。在上述研究基础上, 通过排除最佳路径的影响, Wang 等人<sup>[15]</sup>对解码方式进行改进。由于此类方法需要定义一个实体层次结构, 这个结构需要事先设计好, 并且需要适应各种不同的语境和领域。Zhang 等人<sup>[26]</sup>通过去除实体内部和实体间混淆因子来增强数据, 从而提高生成方法的性能。如果实体层次结构设计不当或者过于复杂, 可能会导致标注不准确或者无法识别某些实体。此外, 在实际应用中, 模型可能会遇到多层嵌套的复杂实体, 而基于分层的方法可能无法处理这种复杂的嵌套结构, 导致识别不准确。

第 2 种是基于序列生成的方法。基于序列生成的方法使用生成模型来预测每个位置的标签, 这种方法不需要任何外部标注数据。这种方法可以使用条件随机场或生成对抗网络来生成序列标记。Strakov 等人<sup>[17]</sup>提出一种序列到序列 (Seq2Seq) 架构来识别 BILOU 标注模式 (beginning, inside, outside, last, unit) 的嵌套实体。此外, Yan 等人<sup>[16]</sup>提出一种结合条件随机场 (conditional random field, CRF) 和生成式模型的统一 Seq2Seq 生成框架, 以提高命名实体识别的性能和鲁棒性。但当嵌套命名实体的层数较多时, 此类方法可能会出现长距离依赖问题, 即先前的预测错误会影响后续的预测, 导致错误的级联扩散, 且当训练数据中存在某些类型的命名实体的数量不足时, 模型容易产生偏差泄漏问题。

第 3 种是基于超图神经网络的方法。基于超图神经网络的方法将句子表示为超图, 将嵌套实体作为超边, 将实体的组成部分作为超点。这种方法可以将复杂的嵌套实体结构映射为超图中的子图, 使用图卷积网络 (graph convolution network, GCN) 来对子图进行特征提取和分类。Lu 等人<sup>[18]</sup>引入了一种用于捕获嵌套实体提及的超图表示。Katiyar 等人<sup>[19]</sup>提出了嵌套 NER 任务的超图表示, 并通过 LSTM 网络贪婪地学习超图结构。基于超图的嵌套命名实体识别方法需要构建超图, 超图的构建需要考虑实体之间的嵌套关系和依赖关系, 这使得算法的复杂度较高, 计算量较大, 需要耗费较长的时间。此外, 类似方法对于领域依赖性较强, 需要针对具体的领域进行定制化的处理和模型训练。

第 4 种是针对上述问题, 提出的基于跨度的方法。Sohrab 等人<sup>[21]</sup>提出了一种基于跨度的命名实体识别方法, 将嵌套命名实体识别任务看作一系列跨度预测问题。该方法在每个位置预测可能的实体跨度, 并对跨度进行过滤和合并, 从而得到最终的嵌套实体。在文献 [21] 的研究基础上, 一些研究通过研究图结构<sup>[27]</sup>、学习高质量跨度的模式<sup>[28,29]</sup>和注意力机制<sup>[30,31]</sup>改进了此类方法。其中, 最具代表性的工作之一是 Yu 等人所提出的模型<sup>[30]</sup>。他们采用

了双仿射注意力机制来加强头部和尾部词元之间的交互, 并使用解码器获得每个跨度的得分. 由于其简单的解码方法, 后续的工作都采用了这种范式. 例如, Zhu 等人<sup>[32]</sup>在原有模型<sup>[30]</sup>上采用了边界平滑技术, 以缓解预测器过度自信的问题. Yan 等人<sup>[33]</sup>使用卷积神经网络来建模跨度与其邻近跨度之间的关系. Yuan 等人<sup>[34]</sup>通过融合异构元素, 即词元、边界、标签和相关跨度, 来增强跨度表示.

与以往的研究不同, 本文提出了基于跨度边界感知的嵌套命名实体识别模型, 该模型强调了将跨度之间的语义差异信息作为一个显式特征输入的重要性. 在该模型中, 设计了语义差分机制和跨度边界感知机制. 通过表征和强化跨度之间的语义差异, 帮助模型更好地定位潜在的命名实体. 实验结果表明, 跨度语义差分机制可以获得跨度间语义差异信息. 进一步来说, 跨度边界感知机制通过建模跨度间语义差异信息, 强化嵌套实体的语义表征, 进而帮助模型更好地感知嵌套实体的边界, 从而提高嵌套命名实体识别的性能.

## 2 模型描述

本文提出了一种基于跨度边界感知的嵌套命名实体识别方法, 实现端到端的实体边界及实体类型识别. 如图 2 所示, 该模型的网络结构主要包含 4 个基础模块: 1) 跨度语义编码器, 在预训练语言模型 BERT<sup>[35]</sup>的词级特征编码的基础上, 引入双仿射机制, 构造跨度语义表征矩阵; 2) 语义差分模块, 通过设计二阶对角邻域差分算子, 将跨度语义表示转化为跨度间语义差异表示; 3) 边界感知模块, 构建一层可学习的滑动卷积核, 强化跨度间的语义差异, 从而感知潜在命名实体的边界; 4) 跨度语义解码器, 基于线性解码器对潜在命名实体进行分类. 具体来说, 图 2 从左到右依次为跨度语义编码器, 跨度语义差分模块, 跨度边界感知模块以及跨度语义解码器. 首先, 将句子“Another tornado hits Geneva, near the Alabama-Florida line, said Mayor Warren Beck.”输入跨度语义编码器中来构造句子的跨度表示矩阵  $M^0$ . 然后, 跨度语义差分模块利用固定参数的对角邻域差分算子  $D_p$ 、 $D_c$ 、 $D_{Rp}$  以及  $D_{Rc}$  将  $M^0$  转化为跨度之间的语义二阶差异表示  $M^2$  和  $M^{2'}$ . 接着, 为了强化不同方向上跨度间语义信息的差异性, 我们分别使用一个可学习的卷积核来扩大跨度间语义差异信息的感知范围, 得到跨度边界矩阵  $M^3$  和  $M^{3'}$ . 最后, 通过多个线性层和残差连接, 融合对角线上跨度边界矩阵以及跨度原始语义表示矩阵  $M^0$ , 输出跨度的类型概率矩阵  $p$ .  $\oplus$  表示矩阵的拼接操作.

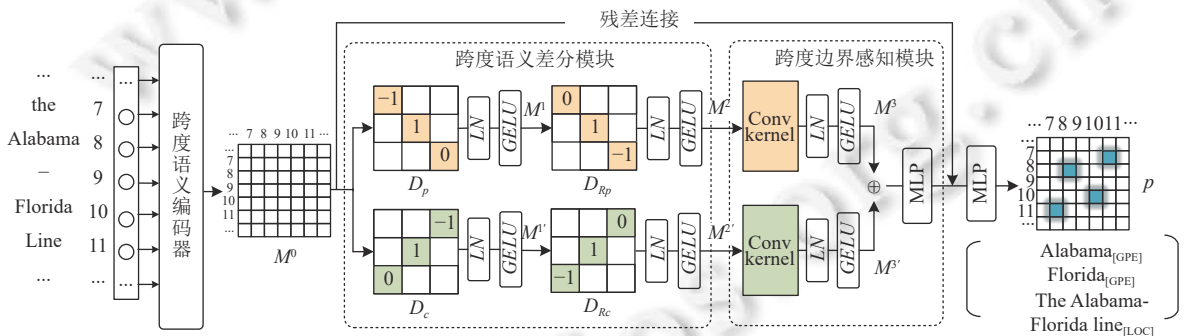


图 2 模型框架

### 2.1 任务定义

假设给定输入句子为  $X = \{x_1, x_2, \dots, x_n\}$ , 命名实体识别任务是识别句子中潜在的实体, 并将它们分类为预定义的实体类型集合  $T = \{t_1, t_2, \dots, t_k\}$  中的一个实体类型. 通常情况下, 一个实体可以表示为一个三元组  $(s_i, e_i, t_i)$ , 其中  $s_i$  是实体起始位置,  $e_i$  是实体结束位置,  $t_i \in T$  是实体类型.

### 2.2 跨度语义编码器

在基于跨度的实体识别方法中, 通过枚举句子中所有的跨度表示, 可以生成潜在实体. 为了获得更加丰富的语义特征, 本文设计了一个跨度语义编码器. 给定句子为  $X = \{x_1, x_2, \dots, x_i\}$ . 首先, 利用预训练模型 BERT 将词元  $x_i$  向

量化, 得到句子的词级特征表示向量  $H$ . 其次, 设计两个前馈神经网络 (feed-forward neural network, FNN) 进行词元映射变换, 得到跨度  $S$  的头词元和尾词元的语义表示向量  $(h_s, h_e)$ . 最后, 使用双仿射模型融合跨度头词元和尾词元  $(h_s, h_e)$  以及对应跨度的宽度表示向量  $w_{ij}$ , 捕获头尾词元语义表示之间的语义相关性, 从而构建跨度表示矩阵  $M^0 \in \mathbb{R}^{l \times l \times d}$ . 其中,  $l$  为句子的长度,  $d$  为跨度语义的特征向量维度. 相较于直接拼接或者池化头尾词元, 本文采用的双仿射机制通过计算头词元和尾词元之间的相关性, 可以有效地捕捉到头尾词元之间的语义联系<sup>[34]</sup>. 另外, 双仿射机制输出每个词元与其他词元所构成跨度的语义特征, 构成跨度语义矩阵, 便于后续边界信息提取以及实体解码. 对于头词元的位置索引为  $i$ , 尾词元的位置索引为  $j$  的跨度  $S_{ij}$ , 其表示向量  $M_{ij}^0$  计算公式如下:

$$\begin{cases} h_s = GELU(HW_s) \\ h_e = GELU(HW_e) \\ M_{ij}^0 = (h_s[i] \oplus h_e[j] \oplus w_{ij})W + h_s[i]Uh_e[j]^T \end{cases} \quad (1)$$

其中,  $h_s, h_e \in \mathbb{R}^{h}$ ,  $h$  为隐藏层数;  $w_{ij} \in \mathbb{R}^c$  为跨度  $S_{ij}$  宽度的嵌入,  $c$  为跨度宽度嵌入数;  $W_s, W_e \in \mathbb{R}^{h \times h}$ ,  $W \in \mathbb{R}^{(2h+c) \times r}$  和  $U \in \mathbb{R}^{h \times r \times h}$ , 均为可训练的参数矩阵,  $r$  为双仿射特征数;  $\oplus$  代表张量的拼接操作;  $GELU$  为 GELUs 激活函数. 值得注意的是, 当  $M_{ij}^0$  位于  $M^0$  矩阵的非对角线位置时, 跨度  $S_{ij}$  的表示具有两种形式, 并且沿对角线对称.

### 2.3 跨度语义差分模块

在跨度表示矩阵  $M^0$  中, 枚举生成跨度会带来大量的负样本. 因此如何过滤负样本并准确定位潜在实体是一项复杂且困难的任务. 受计算机视觉中 Roberts 算子<sup>[36]</sup>启发, 本文提出了一种跨度语义差分机制. 具体而言, 本机制通过设计一种特定的二阶对角邻域差分算子来计算跨度间的语义距离, 并将其表示为跨度语义差异信息. 最后, 网络模型通过梯度反向传播, 能够增大嵌套实体在跨度语义表示空间中的距离, 来降低模型的训练损失, 进而分辨出潜在实体.

根据调研发现, 实体的语义差异主要取决于头尾词元的语义表示. 如图 3 所示, 在编码器生成的跨度表示矩阵中, 以跨度  $S_{8,9}$  为中心的  $3 \times 3$  区域内, 与其头尾实体不同的跨度均分布于主副对角线上, 即  $S_{7,8}$ ,  $S_{9,10}$  等. 水平方向上的数值表示跨度头词元的位置, 而垂直方向上的数值则表示跨度尾词元的位置. 例如, 句子中跨度 “Alabama-Florida” 的头尾词元的位置分别为 8 和 10, 则用符号  $S_{8,10}$  来表示. 跨度表示矩阵中的实体跨度用下划线标识.

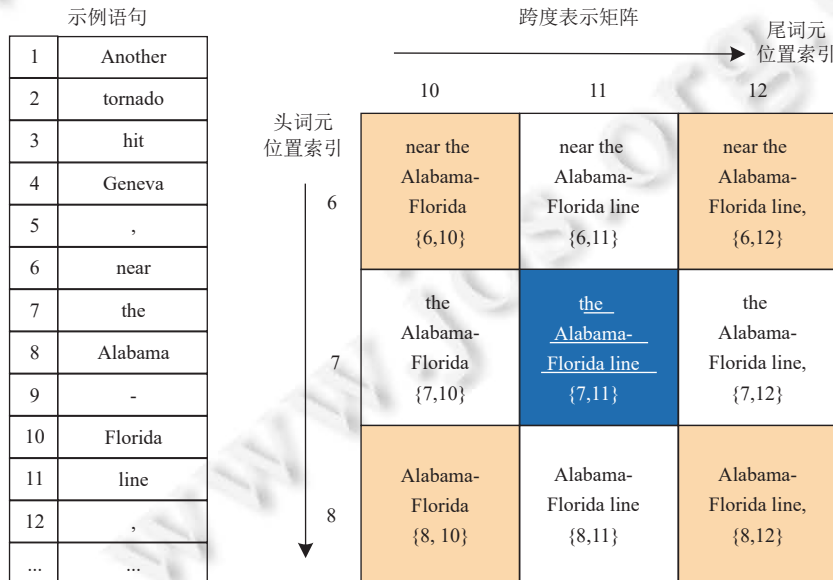


图 3 例句中的部分跨度表示矩阵

基于此, 本文设计了一种轻量级的对角邻域差分算子, 通过计算主对角线 (principal diagonal) 和副对角线 (counter diagonal) 方向跨度间语义距离  $M_{ij}^1$  和  $M_{ij}'^1$ , 构建跨度间语义差异表示矩阵. 模型中, 用卷积核具体实现算子的差分过程. 卷积核参数以及计算过程如下所示:

$$D_p = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad D_c = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (2)$$

$$\begin{cases} M_{ij}^1 = D_p(M_{ij}^0) = M_{ij}^0 - M_{(i-1)(j-1)}^0 \\ M_{ij}'^1 = D_c(M_{ij}^0) = M_{ij}^0 - M_{(i-1)(j+1)}^0 \end{cases} \quad (3)$$

此外, 选取主副对角线的反方向为计算方向 (在经过第 1 个差分算子之后, 跨度间的语义距离信息被提取到被差分的跨度位置上. 例如, 图 2 中位于 {8, 9} 和 {9, 10} 的跨度之间的语义差异信息, 被提取到 {9, 10} 上. 因此, 若计算位于 {8, 9} 的跨度的二阶差分信息, 需要采用反方向的差分算子, 即差分位于 {8, 9} 和 {9, 10} 上的语义差异信息), 进一步计算上述跨度的二阶跨度语义差表示  $M_{ij}^2$  和  $M_{ij}'^2$ . 具体卷积核参数以及计算过程如下所示:

$$D_{Rp} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \quad D_{Rc} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \quad (4)$$

$$\begin{cases} M_{ij}^2 = D_{Rp}(M_{ij}^1) = M_{ij}^1 - M_{(i+1)(j+1)}^1 \\ M_{ij}'^2 = D_{Rc}(M_{ij}'^1) = M_{ij}'^1 - M_{(i+1)(j-1)}^1 \end{cases} \quad (5)$$

较于 Roberts 算子, 本文提出的跨度语义差分机制所提出的对角邻域差分算子采用  $3 \times 3$  的模板, 避免了 Roberts 算子在计算时出现跨度特征漂移问题<sup>[37]</sup>. 其次, 为避免在计算高阶语义差异信息时出现差分对象漂移问题, 本文所提出的跨度语义差分机制还设计了反方向的差分算子, 而不是简单地重复使用原始的差分算子.

为了保证训练过程的平稳性, 我们采用规范化层 (layer normalization, LN)<sup>[38]</sup>和激活函数 (GELU) 对每个差分算子计算后的特征表示进行处理, 以获得分布稳定的语义差异特征, 进而构建用于表示跨度间语义差异信息的特征矩阵.  $M^1$  和  $M^2$  具体计算过程如公式 (6), 公式 (7) 所示,  $M'^1$  和  $M'^2$  计算过程与其类似.

$$M^1 = D_p \text{Block}(M^0) = \text{GELU}(\text{LN}(D_p(M^0))) \quad (6)$$

$$M^2 = D_{Rp} \text{Block}(M^1) = \text{GELU}(\text{LN}(D_{Rp}(M^1))) \quad (7)$$

其中,  $D_p \text{Block}$  表示主对角线语义差分块.  $D_{Rp} \text{Block}$  表示副对角线方向语义差分块. LN 为规范化层, 即该层在特征维度  $d$  进行标准化.

#### 2.4 跨度边界感知模块

在得到跨度间语义差异信息的特征矩阵后, 为进一步感知更大范围内的语义差异信息并定位潜在实体的边界位置, 本文提出一种基于可学习卷积核的跨度边界感知机制. 该机制使用滑动窗口对该跨度邻域内的语义差异信息进行动态加权求和, 使模型获得更全局的上下文语义差异信息, 进一步提高定位潜在实体位置索引的准确性. 相较于跨度语义差分模块所获得的语义差异信息, 该机制能有效地对边界位置进行感知和判断.

具体来说, 首先, 利用滑动卷积核处理语义差分模块沿主、副对角方向获得的跨度语义差异信息表示矩阵. 然后, 利用线性层融合主、副对角方向上的语义差异信息, 增强跨度间语义边界信息的表示, 获得跨度边界矩阵, 从而更有效地帮助模型定位潜在的命名实体. 具体计算过程如下:

$$\begin{cases} M^3 = \text{GELU}(\text{LN}(\text{Conv}(M^2))) \\ M^{3'} = \text{GELU}(\text{LN}(\text{Conv}(M^{2'}))) \end{cases} \quad (8)$$

$$M_{\text{fuse}}^3 = W_{\text{fuse}}(M^3 \oplus M^{3'}) + B_{\text{fuse}} \quad (9)$$

其中,  $W_{\text{fuse}} \in \mathbb{R}^{(d \times 2) \times d}$ ,  $B_{\text{fuse}} \in \mathbb{R}^d$ . 为保证训练结果的一致性, 本模块中所有卷积核均不包含偏置项, 并使用 0 填充<sup>[33]</sup>.

## 2.5 跨度语义解码器

由于跨度语义差分模块和跨度边界感知模块中存在差分和卷积操作, 其输出矩阵  $M^3$  缺失了跨度的原始语义信息. 为了保留跨度的完整语义信息, 我们将  $M^0$  作为残差项添加到  $M^3$  中, 从而得到保留了完整语义信息的特征表示. 最后, 通过一个线性解码器输出潜在实体的类别概率.

$$p = \text{Sigmoid}(W_p(M^0 + M_{\text{fuse}}^3) + B_p) \quad (10)$$

其中,  $p \in \mathbb{R}^{l \times l}$ ,  $W_p \in \mathbb{R}^{d \times d}$ ,  $B_p \in \mathbb{R}^l$ ,  $l$  为预定义的实体类别个数.  $W_d$  和  $B_d$  均为可训练的参数.

## 2.6 训练策略

我们使用二值交叉熵函数来计算损失:

$$\mathcal{L} = - \sum_{0 \leq i, j < l} y_{ij} \log(p_{ij}) + (1 - y_{ij}) \log(1 - p_{ij}) \quad (11)$$

其中,  $y_{ij}$  为跨度  $S_{ij}$  的实体类型.

在计算损失时, 由于在批训练的过程中, 本模型无法单独对矩阵的上三角部分计算. 所以我们将矩阵上三角和下三角部分所有跨度的类型预测概率都计入计算损失. 而且矩阵  $M$  中所表示的实体是对称的, 即  $p_{ij}$  与  $p_{ji}$  所对应的真实实体类型也相同.

在测试时, 我们将矩阵  $p$  的上下三角部分对称相加, 得到矩阵  $p'$ . 然后只使用  $p'$  的上三角部分来解码.

$$p'_{ij} = (p_{ij} + p_{ji}) / 2 \quad (12)$$

主要解码方法主要参考 Yu 等人<sup>[23]</sup>的工作. 首先过滤非实体的跨度 (对预测为所有实体类别的概率均没有超过 0.5), 然后根据预测概率对剩余的跨度进行降序排序. 我们根据顺序选择跨度, 如果当前跨度的边界与已经选定的跨度边界冲突, 则认为当前跨度为非实体跨度.

## 3 实验及分析

本节中, 我们首先介绍了数据集、基线模型以及实现细节. 接下来, 我们在 3 个基准数据集上验证了模型的先进性. 此外, 我们通过一系列的消融实验和案例分析研究, 进一步证明了本文提出的跨度语义差分机制和跨度边界感知机制能够捕获嵌套实体间的语义差异, 并感知和强化嵌套实体间的语义边界, 即潜在实体的位置索引.

### 3.1 数据集

我们在 3 个常用的命名实体识别数据集上进行了模型评估: ACE04 (<https://catalog.ldc.upenn.edu/LDC2005T09>), ACE05 (<https://catalog.ldc.upenn.edu/LDC2006T06>) 和 Genia<sup>[39]</sup>. 其中, ACE04 和 ACE05 数据集包含 7 个实体类型, 分别是: 人名 (PER)、组织机构名 (ORG)、地理政治实体名 (GPE)、地名 (LOC)、基础设施 (FAC)、武器名 (WEA) 和交通工具 (VEH). 根据统计分析, ACE04 和 ACE05 数据集中 30% 的语句存在嵌套实体, Genia 数据集中则有 17% 的语句存在嵌套实体. 对于 ACE 数据集, 我们使用了 Yan 等人<sup>[33]</sup>公开的数据预处理代码, 并按照 8:1:1 的比例将数据划分为训练集、验证集和测试集. 对于 Genia 数据集, 按照 Li 等人<sup>[40]</sup>的方法将数据集中的实体分为 5 种类型, 并按照 8.1:0.9:1.0 的比例将数据划分为训练集、验证集和测试集. 3 个基准数据集的统计信息如表 1 所示.

表 1 3 个基准数据集的统计信息

数据集	类别	训练集	验证集	测试集	平均长度
ACE04	句子	6297	742	824	23.52
	实体	22231	2514	3036	2.64
ACE05	句子	7178	960	1051	20.59
	实体	25300	3321	3099	2.4
Genia	句子	15023	1669	1854	25.41
	实体	45144	5365	5506	1.97

### 3.2 对比模型

为了验证所提出模型的性能,将在 3 个公开数据集上与如下相关工作中提出的模型进行对比.

**Biaffine:** Yu 等人<sup>[23]</sup>使用双仿射模型来识别嵌套命名实体任务,通过预测两个单词之间的依存关系来预测命名实体的边界.

**Second-Best:** Wang 等人<sup>[15]</sup>通过显式地排除概率图最优路径的影响来实现对嵌套实体的识别.

**Seq2Seq:** Yan 等人<sup>[16]</sup>使用了指针方式,将实体标注任务转化为一个序列生成任务.

**Sequence2Set:** Tan 等人<sup>[24]</sup>针对嵌套 NER 提出了一种新的集合预测的神经网络架构.

**W2NER:** Li 等人<sup>[40]</sup>将统一命名实体识别建模为词-词关系分类,避免了传统序列标注方法中标签间的冲突问题.

**De-bias:** Zhang 等人<sup>[26]</sup>从因果关系的角度分析实体生成过程中的错误偏差,并将其归因于两个混淆因素:前文混淆因素和实体顺序混淆因素.

**Locate and label:** Shen 等人<sup>[28]</sup>将嵌套命名实体识别任务建模为实体边界回归和跨度分类的联合任务,提高模型的训练和推断效率.

**BS:** Zhu 等人<sup>[32]</sup>提出了一种基于边界平滑的方法来改善 NER 模型的性能.该方法的核心思想是通过在实体边界周围插入一些模糊区域,使得模型更容易识别出实体的边界.

**CNNER:** Yan 等人<sup>[33]</sup>使用 CNN 来建模分数矩阵中的空间关系来解决嵌套命名实体识别任务.

**Triaffine:** Yuan 等人<sup>[34]</sup>通过获取词元、实体类型、跨度和边界等异质元素之间的各种交互信息,提升实体检测的性能.

**Span-Graph:** Wan 等人<sup>[27]</sup>利用基于跨度的图结构进行建模,其中每个跨度表示为一个节点,并通过边连接跨度,提升跨度语义表征能力.

### 3.3 实验细节

本文实验采用 Python 3.8 和 PyTorch 1.10.1 的环境.对 ACE 数据集,我们使用 RoBERTa-Base (<https://huggingface.co/roberta-base>) 和 BERT-large-cased (<https://huggingface.co/bert-large-cased>) 作为预训练模型,模型隐藏层数为 768.在 Genia 数据集上,采用 BioBERT-Base (<https://huggingface.co/dmis-lab/biobert-base-cased-v1.1>) 作为预训练模型,模型隐藏层数为 768.双仿射模型的超参数选取参考了先前同类研究<sup>[33]</sup>.

模型的跨度嵌入宽度为 25.我们使用 AdamW 优化器对模型进行训练<sup>[41]</sup>.为了控制过拟合,我们使梯度裁剪,将梯度的 L2 范数限制在 5 以内<sup>[42]</sup>.在前 10% 的训练步骤中,我们采用线性预热调度器逐渐增加学习率.预热期结束后,使用线性衰减调度器逐渐降低学习率.其他因数据集而有所变化的超参数详见表 2.

表 2 模型的其他超参数

参数	ACE04	ACE05	Genia
批次大小 (batchsize)	48	48	8
训练轮数 (epochs)	50	50	5
学习率 (learning rate)	3E-5	3E-5	6E-7
双仿射特征数 ( $r$ )	200	200	400
跨度语义特征向量维度 ( $d$ )	120	200	200
神经元丢弃率 (dropout)	0.1	0	0.1

### 3.4 实验结果

根据实验数据的特点,并参考近期相关工作的通行做法,实验结果同时报告 3 种分类评价指标,包括精准度 (P)、召回率 (R) 和  $F1$ -score ( $F1$ ) 来评估模型的性能.我们使用严格的评估指标,即当实体边界和实体类别同时预测正确时,才认为实体识别的结果是正确的.此外,为了得到稳定和可靠的评估结果,我们进行了多次不同随机种子的重复实验,然后对得到的  $F1$ -score 进行了统计学分析.具体来说,我们使用 T 检验的方法对实验结果进行显



著性分析, 设定显著性水平为 5%, 以确定不同实验结果之间是否存在显著差异.

Yan 等人<sup>[33]</sup>发现即使近期提出的工作<sup>[27,32,34,40]</sup>使用的数据集相同, 但是由于不同的预处理方法导致最后训练的数据集统计信息均不同. 因此, 无论直接采用任何数据集版本, 来比较模型性能都是不公平的. 于是, 我们采用 Yan 等人公开的代码来预处理数据集. 然后, 通过相同的预处理数据集和公开代码, 重新复现了 2022 年的部分基线模型, 并将模型性能指标记录在表 3 中. 其中, 由于 2022 年的其他基线模型 (Span-Graph 模型和 De-bias 模型) 未公开代码以及模型细节有限, 无法复现完整的模型.

表 3 在 ACE04 和 ACE05 数据集上命名实体识别准确率 (%)

模型	预训练模型	ACE04			ACE05		
		P	R	F1	P	R	F1
Biaffine <sup>[23]</sup> (2020)	BERT-base	87.30	86.00	86.7	85.20	85.60	85.40
Second-Best <sup>[14]</sup> (2021)	BERT-base	86.42	85.71	86.06	83.95	84.67	84.30
Locate-and-Label <sup>[28]</sup> (2021)	BERT-base	87.44	87.38	87.41	86.09	87.27	86.67
Seq2Seq <sup>[16]</sup> (2021)	BART-large	87.27	86.41	86.84	83.16	86.38	84.74
Sequence2Set <sup>[24]</sup> (2021)	BERT-large	88.46	86.10	87.26	87.48	86.63	87.05
Span-Graph <sup>[27]</sup> (2022)	BERT-base	86.70	85.93	86.31	84.37	85.87	85.11
De-bias <sup>[26]</sup> (2022)	T5-base	86.36	84.54	85.44	83.31	86.56	84.90
CNNER <sup>※[33]</sup> (2022)	RoBERTa-base	87.33	87.29	87.31	86.70	88.16	87.42
BS <sup>[43]※</sup> (2022)	RoBERTa-base	87.32	86.84	87.08	86.58	87.84	87.20
Triaffine <sup>*[34]</sup> (2022)	BERT-large	87.13	87.68	87.4	86.70	86.94	86.82
Triaffine <sup>※[34]</sup> (2022)	BERT-large	87.19	86.29	86.74	87.00	85.54	86.26
W2NER <sup>※[40]</sup> (2022)	BERT-large	87.17	87.70	87.43	85.78	87.81	86.77
本文模型	RoBERTa-base	87.58	<b>88.51</b>	88.04	<b>88.59</b>	<b>88.48</b>	<b>88.54</b>
	BERT-large	<b>88.13</b>	88.32	<b>88.22</b>	88.43	87.43	87.91

注: 用※标识的模型代表使用了与我们相同的预处理数据, 并使用公开的代码复现了模型<sup>[33]</sup>. 用\*标识的模型代表基于作者提供的预处理数据, 并使用公开的代码复现了模型

表 3 和表 4 展示了所提出的模型和基准模型在 ACE04、ACE05、Genia 上的性能表现. 我们的模型在 3 个 NER 数据集上的性能优于所有基准模型. 具体而言, 以 RoBERTa-base 为预训练模型, 我们的模型在 ACE04, ACE05 上的 F1 分数分别比之前的模型提高了 0.71% 和 1.22%. 以 BERT-large 为预训练模型, 我们的模型在 ACE04, ACE05 上的 F1 分数分别比之前的模型提高了 0.79% 和 0.86%. 我们的模型在 Genia 上的 F1 分数分别比之前的模型提高了 0.46%.

表 4 在 Genia 数据集上命名实体识别准确率 (%)

模型	P	R	F1
Biaffine <sup>[23]</sup> (2020)	81.8	79.3	80.5
Second-Best <sup>[14]</sup> (2021)	79.2	78.16	78.63
Seq2Seq <sup>[16]</sup> (2021)	78.57	79.3	78.93
Locate-and-Label <sup>[28]</sup> (2021)	80.19	80.89	80.54
Sequence2Set <sup>[24]</sup> (2021)	82.3	78.7	80.4
De-bias <sup>[26]</sup> (2022)	81.04	77.21	79.08
Span Graph <sup>[27]</sup> (2022)	77.92	80.74	79.3
W2NER <sup>※[40]</sup> (2022)	83.1	79.76	81.39
BS <sup>*[43]</sup> (2022)	82.53	78.69	80.56
BS <sup>※[43]</sup> (2022)	80.74	79.83	80.28

表 4 在 Genia 数据集上命名实体识别准确率 (%) (续)

模型	P	R	F1
CNNER <sup>※</sup> [33] (2022)	83.19	79.7	81.4
Triaffine <sup>※</sup> [34] (2022)	81.79	79.39	80.57
Triaffine <sup>[34]</sup> (2022)	80.42	<b>82.06</b>	81.23
所提出的模型	<b>82.93</b>	80.82	<b>81.86</b>

注: 表格中的所有模型均使用BioBERT-Base作为预训练模型. 另外, 用※标识的模型代表使用了与我们相同的预处理数据, 并使用公开的代码复现了模型<sup>[33]</sup>. 用\*标识的模型代表使用了作者提供的预处理数据, 并使用公开的代码复现了模型

### 3.5 消融实验

为了验证提出的模型和机制的有效性, 在 ACE04 和 ACE05 数据集上进行多组消融实验, 结果如表 5 所示. 首先, 将跨度语义差分模块和跨度边界感知模块移除. 实验结果表明, 在两个数据集上, 模型的性能都明显下降, 这验证了将跨度语义差作为显式特征输入模型的有效性. 接着, 我们尝试多种跨度的差分方向, 发现选取其他方向组合会导致模型性能下降, 这表明相较于其他方向上的跨度, 主副对角线更适合用于衡量邻域跨度与中心跨度之间的语义差异. 然后, 将所有的卷积核移除, 发现模型性能在两个数据集上都明显下降, 这证明了边界感知模块的有效性. 需要注意的是, 随着卷积核数的增加, 并不会提高模型识别命名实体的准确率. 我们认为这是因为跨度间的语义差异信息属于细粒度信息, 过多地堆叠卷积核会导致模型丢失中心跨度的位置和细节信息.

表 5 消融实验结果 (以 RoBERTa-base 为预训练语言模型)(%)

验证集	ACE04		ACE05	
	F1	$\Delta$	F1	$\Delta$
w/o 跨度语言差分模块和跨度边界感知模块	86.52	-1.71	85.71	-1.36
水平方向	87.79	-0.44	86.75	-0.32
垂直方向	88.15	-0.08	86.72	-0.35
水平方向和垂直方向	87.65	-0.58	86.96	-0.11
主对角线方向	88.15	-0.08	86.77	-0.30
副对角线方向	88.16	-0.07	87.06	-0.01
所有方向	87.67	-0.56	87.05	-0.02
w/o 跨度边界感知模块	87.69	-0.54	86.35	-0.72
2层卷积核	87.57	-0.66	86.78	-0.29
3层卷积核	87.85	-0.38	86.82	-0.25
所提出的模型	<b>88.23</b>	—	<b>87.07</b>	—

注:  $\Delta$ 列表示在此实验条件下的模型与我们提出的模型的性能指标 (F1-score) 之间的差值. w/o 代表缺失某一模块的模型. 此外, “\*\*方向”表示在语义差分模块中, 差分算子只沿\*\*方向计算跨度间的语义距离

### 3.6 案例研究

我们选取图 1 的例句 2 作为案例, 旨在研究不同机制下嵌套实体之间的语义差异变化, 并证明我们提出的语义差分机制和跨度边界感知机制能够增强跨度间的语义边界表示. 对有无语义差分模块和边界感知模块的模型生成的跨度语义表示矩阵  $M^0$  和  $\tilde{M}^0$ , 我们分别计算了实体跨度“a man with a gun”与句子中所有跨度的余弦相似度, 并展示了部分相似度矩阵, 如图 4 所示. 图 4(a) 表示在缺少语义差分模块和边界感知模块的条件下, 基于模型的跨度语义表示  $\tilde{M}^0$  生成的语义相似热力图. 图 4(b) 表示在含有语义差分模块和边界感知模块的条件下, 基于模型的跨度语义表示  $M^0$  生成的语义相似热力图. 每一个色块中的数值均代表跨度语义向量的余弦相似度. 在该例中, “a man with a gun”和“a man”的潜在实体类型为 PER, 而“a gun”的潜在实体类型为 WEA. 图 4(a) 显示, 当模型缺少语义差分机制和边界感知机制时, 实体“a man with a gun”与潜在嵌套实体“a man”之间的语义相似度明显低于实体“a

man with a gun”与潜在嵌套实体“a gun”之间的语义相似度。这表明, 由于“a man with a gun”与潜在嵌套实体“a gun”具有相同的头尾词元, 仅经过双仿射模型的跨度语义表示难以很好地区分此类不同实体类型的跨度。然而, 当加入语义差分模块和边界感知模块时, 如图 4(b) 所示, 实体“a man with a gun”与潜在嵌套实体“a man”之间的语义相似度从 0.58 提高到 0.72, 而实体“a man with a gun”与潜在嵌套实体“a gun”之间的语义相似度从 0.62 降低到 0.57。这表明, 合理建模跨度间语义差异信息能够增强模型对嵌套实体语义的表征能力。

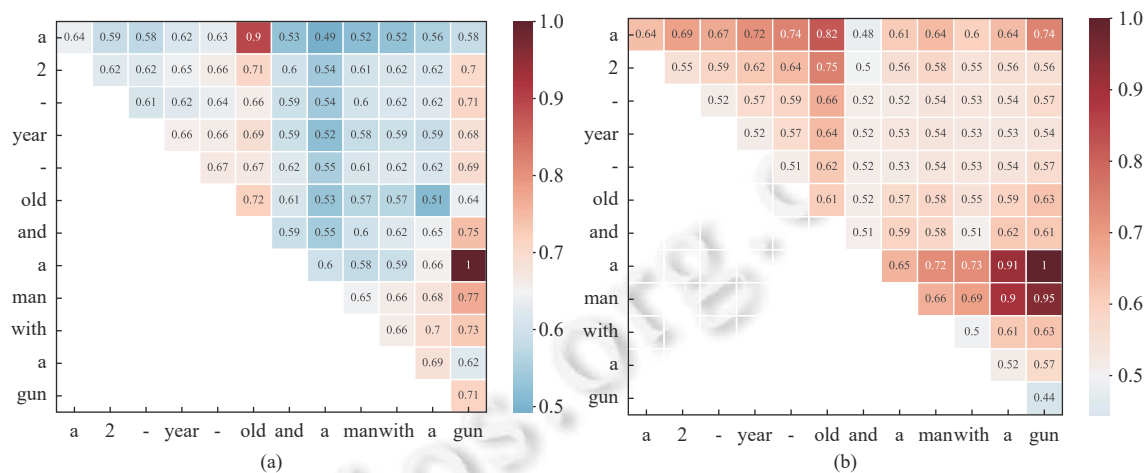


图 4 例句 2 的部分跨度与实体“a man with a gun”的语义相似热力图

## 4 总结

本文提出了一种基于跨度边界感知的神经网络模型, 旨在有效解决实体识别任务嵌套实体边界模糊的问题。为此, 该模型引入了跨度语义差分机制以提取跨度间的语义差异信息, 并设计了一个跨度边界感知机制感知和强化跨度间的边界语义差异。实验结果表明, 该模型在 3 个公开数据集上的性能表现优于现有相关工作, 消融实验和案例研究进一步验证了本文所提出模型的合理性。在后续的研究中, 我们考虑在事件抽取、关系抽取等下游任务中, 验证本文所提出的跨度语义差分机制和跨度边界感知机制的有效性。

## References:

- [1] Lample G, Ballesteros M, Subramanian S, Kawakami K, Dyer C. Neural architectures for named entity recognition. In: Proc. of the 2016 Conf. of the North American Chapter of the Association for Computational Linguistics. San Diego: Association for Computational Linguistics, 2016. 260–270. [doi: 10.18653/v1/N16-1030]
- [2] Miwa M, Sasaki Y. Modeling joint entity and relation extraction with table representation. In: Proc. of the 2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP). Doha: Association for Computational Linguistics, 2014. 1858–1869. [doi: 10.3115/v1/D14-1200]
- [3] Yuan Y, Zhou XF, Pan SR, Zhu QN, Song ZL, Guo L. A relation-specific attention network for joint entity and relation extraction. In: Proc. of the 29th Int'l Joint Conf. on Artificial Intelligence. Yokohama: Int'l Joint Conf. on Artificial Intelligence Organization, 2020. 4054–4060. [doi: 10.24963/ijcai.2020/561]
- [4] Chen Y, Zheng DQ, Zhao TJ. Chinese relation extraction based on Deep Belief Nets. Ruan Jian Xue Bao/Journal of Software, 2012, 23(10): 2572–2585 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/4181.htm> [doi: 10.3724/SP.J.1001.2012.04181]
- [5] Yang BS, Mitchell TM. Joint extraction of events and entities within a document context. In: Proc. of the 2016 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. San Diego: Association for Computational Linguistics, 2016. 289–299. [doi: 10.18653/v1/N16-1033]
- [6] Sha L, Qian F, Chang BB, Sui ZF. Jointly extracting event triggers and arguments by dependency-bridge RNN and tensor-based argument

- interaction. In: Proc. of the 32nd AAAI Conf. on Artificial Intelligence and the 30th Innovative Applications of Artificial Intelligence Conf. and the 8th AAAI Symp. on Educational Advances in Artificial Intelligence. New Orleans: AAAI Press, 2018. 5916–5923. [doi: [10.1609/aaai.v32i1.12034](https://doi.org/10.1609/aaai.v32i1.12034)]
- [7] Li XY, Yin F, Sun ZJ, Li XY, Yuan A, Chai D, Zhou MX, Li JW. Entity-relation extraction as multi-turn question answering. In: Proc. of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: Association for Computational Linguistics, 2019. 1340–1350. [doi: [10.18653/v1/P19-1129](https://doi.org/10.18653/v1/P19-1129)]
- [8] Du XY, Cardie C. Event extraction by answering (almost) natural questions. In: Proc. of the 2020 Conf. on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, 2020. 671–683. [doi: [10.18653/v1/2020.emnlp-main.49](https://doi.org/10.18653/v1/2020.emnlp-main.49)]
- [9] Finkel JR, Manning CD. Nested named entity recognition. In: Proc. of the 2009 Conf. on Empirical Methods in Natural Language Processing. Singapore: Association for Computational Linguistics, 2009. 141–150.
- [10] Ma XZ, Hovy E. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In: Proc. of the 54th Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Berlin: Association for Computational Linguistics, 2016. 1064–1074. [doi: [10.18653/v1/P16-1101](https://doi.org/10.18653/v1/P16-1101)]
- [11] Chiu JPC, Nichols E. Named entity recognition with bidirectional LSTM-CNNs. *Trans. of the Association for Computational Linguistics*, 2016, 4: 357–370. [doi: [10.1162/tacl\\_a\\_00104](https://doi.org/10.1162/tacl_a_00104)]
- [12] Ju MZ, Miwa M, Ananiadou S. A neural layered model for nested named entity recognition. In: Proc. of the 2018 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long Papers). New Orleans: Association for Computational Linguistics, 2018. 1446–1459. [doi: [10.18653/v1/N18-1131](https://doi.org/10.18653/v1/N18-1131)]
- [13] Wang J, Shou LD, Chen K, Chen G. Pyramid: A layered model for nested named entity recognition. In: Proc. of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2020. 5918–5928. [doi: [10.18653/v1/2020.acl-main.525](https://doi.org/10.18653/v1/2020.acl-main.525)]
- [14] Shibuya T, Hovy E. Nested named entity recognition via second-best sequence learning and decoding. *Trans. of the Association for Computational Linguistics*, 2020, 8: 605–620. [doi: [10.1162/tacl\\_a\\_00334](https://doi.org/10.1162/tacl_a_00334)]
- [15] Wang YR, Shindo H, Matsumoto Y, Watanabe T. Nested named entity recognition via explicitly excluding the influence of the best path. In: Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int'l Joint Conf. on Natural Language Processing (Vol. 1: Long Papers). Association for Computational Linguistics, 2021. 3547–3557. [doi: [10.18653/v1/2021.acl-long.275](https://doi.org/10.18653/v1/2021.acl-long.275)]
- [16] Yan H, Gui T, Dai JQ, Guo QP, Zhang Z, Qiu XP. A unified generative framework for various NER subtasks. In: Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int'l Joint Conf. on Natural Language Processing (Vol. 1: Long Papers). Association for Computational Linguistics, 2021. 5808–5822. [doi: [10.18653/v1/2021.acl-long.451](https://doi.org/10.18653/v1/2021.acl-long.451)]
- [17] Straková J, Straka M, Hajic J. Neural architectures for nested NER through linearization. In: Proc. of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: Association for Computational Linguistics, 2019. 5326–5331. [doi: [10.18653/v1/P19-1527](https://doi.org/10.18653/v1/P19-1527)]
- [18] Lu W, Roth D. Joint mention extraction and classification with mention hypergraphs. In: Proc. of the 2015 Conf. on Empirical Methods in Natural Language Processing. Lisbon: Association for Computational Linguistics, 2015. 857–867. [doi: [10.18653/v1/D15-1102](https://doi.org/10.18653/v1/D15-1102)]
- [19] Katiyar A, Cardie C. Nested named entity recognition revisited. In: Proc. of the 2018 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 1 (Long Papers). New Orleans: Association for Computational Linguistics, 2018. 861–871. [doi: [10.18653/v1/N18-1079](https://doi.org/10.18653/v1/N18-1079)]
- [20] Wang BL, Lu W. Neural segmental hypergraphs for overlapping mention recognition. In: Proc. of the 2018 Conf. on Empirical Methods in Natural Language Processing. Brussels: Association for Computational Linguistics, 2018. 204–214. [doi: [10.18653/v1/D18-1019](https://doi.org/10.18653/v1/D18-1019)]
- [21] Sohrab MG, Miwa M. Deep exhaustive model for nested named entity recognition. In: Proc. of the 2018 Conf. on Empirical Methods in Natural Language Processing. Brussels: Association for Computational Linguistics, 2018. 2843–2849. [doi: [10.18653/v1/D18-1309](https://doi.org/10.18653/v1/D18-1309)]
- [22] Tan CQ, Qiu W, Chen MS, Wang R, Huang F. Boundary enhanced neural span classification for nested named entity recognition. *Proc. of the AAAI Conf. on Artificial Intelligence*, 2020, 34(5): 9016–9023. [doi: [10.1609/aaai.v34i05.6434](https://doi.org/10.1609/aaai.v34i05.6434)]
- [23] Yu JT, Bohnet B, Poesio M. Named entity recognition as dependency parsing. In: Proc. of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2020. 6470–6476. [doi: [10.18653/v1/2020.acl-main.577](https://doi.org/10.18653/v1/2020.acl-main.577)]
- [24] Tan ZQ, Shen YL, Zhang S, Lu WM, Zhuang YT. A sequence-to-set network for nested named entity recognition. In: Proc. of the 30th Int'l Joint Conf. on Artificial Intelligence. Vienna: Int'l Joint Conf. on Artificial Intelligence Organization, 2021. 3936–3942. [doi: [10.1145/3455553](https://doi.org/10.1145/3455553)]

- 24963/ijcai.2021/542]
- [25] Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. *Nature*, 1986, 323(6088): 533–536. [doi: 10.1038/323533a0]
- [26] Zhang S, Shen YL, Tan ZQ, Wu YQ, Lu WM. De-bias for generative extraction in unified NER task. In: Proc. of the 60th Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Dublin: Association for Computational Linguistics, 2022. 808–818. [doi: 10.18653/v1/2022.acl-long.59]
- [27] Wan JC, Ru DY, Zhang WN, Yu Y. Nested named entity recognition with span-level graphs. In: Proc. of the 60th Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Dublin: Association for Computational Linguistics, 2022. 892–903. [doi: 10.18653/v1/2022.acl-long.63]
- [28] Shen YL, Ma XY, Tan ZQ, Zhang S, Wang W, Lu WM. Locate and label: A two-stage identifier for nested named entity recognition. In: Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int'l Joint Conf. on Natural Language Processing (Vol. 1: Long Papers). Association for Computational Linguistics, 2021. 2782–2794. [doi: 10.18653/v1/2021.acl-long.216]
- [29] Abdulkadhar S, Bhasuran B, Natarajan J. Multiscale Laplacian graph kernel combined with lexico-syntactic patterns for biomedical event extraction from literature. *Knowledge and Information Systems*, 2021, 63(1): 143–173. [doi: 10.1007/s10115-020-01514-8]
- [30] Jin N, Wu JX, Ma X, Yan K, Mo YC. Multi-task learning model based on multi-scale CNN and LSTM for sentiment classification. *IEEE Access*, 2020, 8: 77060–77072. [doi: 10.1109/ACCESS.2020.2989428]
- [31] Xu YX, Huang HY, Feng C, Hu Y. A supervised multi-head self-attention network for nested named entity recognition. *Proc. of the AAAI Conf. on Artificial Intelligence*, 2021, 35(16): 14185–14193. [doi: 10.1609/aaai.v35i16.17669]
- [32] Zhu EW, Li JP. Boundary smoothing for named entity recognition. In: Proc. of the 60th Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers). Dublin: Association for Computational Linguistics, 2022. 7096–7108. [doi: 10.18653/v1/2022.acl-long.490]
- [33] Yan H, Sun Y, Li XN, Qiu XP. An embarrassingly easy but strong baseline for nested named entity recognition. In: Proc. of the 61st Annual Meeting of the Association for Computational Linguistics. Toronto: Association for Computational Linguistics, 2023. 1442–1452. [doi: 10.18653/v1/2023.acl-short.123]
- [34] Yuan Z, Tan CQ, Huang SF, Huang F. Fusing heterogeneous factors with triaffine mechanism for nested named entity recognition. In: Proc. of Findings of the 2022 Association for Computational Linguistics: ACL 2022. Dublin: Association for Computational Linguistics, 2022. 3174–3186. [doi: 10.18653/v1/2022.findings-acl.250]
- [35] Dozat T, Manning CD. Deep biaffine attention for neural dependency parsing. arXiv:1611.01734, 2017.
- [36] Roberts LG. Machine perception of three-dimensional solids [Ph.D. Thesis]. Massachusetts: MIT, 1963.
- [37] Wu S, Wang GR, Tang P, Chen F, Shi LP. Convolution with even-sized kernels and symmetric padding. In: Proc. of the 33rd Int'l Conf. on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2019. 1194–1205.
- [38] Ba JL, Kiros JR, Hinton GE. Layer normalization. arXiv:1607.06450, 2016.
- [39] Kim JD, Ohta T, Tateisi Y, Tsujii J. GENIA corpus—A semantically annotated corpus for bio-textmining. *Bioinformatics*, 2003, 19(Suppl 1): i180–i182. [doi: 10.1093/bioinformatics/btg1023]
- [40] Li JY, Fei H, Liu J, Wu SQ, Zhang MS, Teng C, Ji DH, Li F. Unified named entity recognition as word-word relation classification. *Proc. of the AAAI Conf. on Artificial Intelligence*, 2022, 36(10): 10965–10973. [doi: 10.1609/aaai.v36i10.21344]
- [41] Loshchilov I, Hutter F. Decoupled weight decay regularization. arXiv:1711.05101, 2019.
- [42] Pascanu R, Mikolov T, Bengio Y. On the difficulty of training recurrent neural networks. In: Proc. of the 30th Int'l Conf. on Machine Learning. Atlanta: JMLR, 2013. III-1310–III-1318.
- [43] Wang HF, Du HF, Qi GL, Chen HJ, Hu W, Chen Z. Construction of a linked data set of COVID-19 knowledge graphs: Development and applications. *JMIR Medical Informatics*, 2022, 10(5): e37215. [doi: 10.2196/37215]

#### 附中文参考文献:

- [4] 陈宇, 郑德权, 赵铁军. 基于 Deep Belief Nets 的中文名实体关系抽取. *软件学报*, 2012, 23(10): 2572–2585. <http://www.jos.org.cn/1000-9825/4181.htm> [doi: 10.3724/SP.J.1001.2012.04181]



蔡宇翔(2001—), 男, 硕士生, 主要研究领域为事件抽取, 知识图谱, 时序推理.



刘雪怡(2001—), 女, 硕士生, 主要研究领域为情感计算, 知识图谱.



骆妲(1999—), 女, 硕士生, 主要研究领域为关系抽取, 知识图谱.



刘峤(1974—), 男, 博士, 教授, 博士生导师, CCF 专业会员, 主要研究领域为知识图谱构建与应用, 情感分析, 推荐系统.



甘洋镞(1995—), 男, 博士生, 主要研究领域为知识图谱, 推荐系统, 时序推理.



石晓军(1982—), 男, 正高级工程师, 主要研究领域为人工智能, 自然语言处理, 物联网, 信息系统架构.



侯睿(1995—), 男, 博士生, 主要研究领域为信息抽取, 知识图谱.