

基于关系提示的单模块单步骤实体关系抽取方法研究

刘辉¹, 张智², 王启源¹

(1. 国防科技大学电子对抗学院, 230037, 合肥; 2. 哈尔滨工业大学计算机科学与技术学院, 150001, 哈尔滨)

摘要: 针对现有关系三元组抽取方法由于忽略关系本身的关系语义信息以及三元组中元素的相互依赖和不可分性所导致的抽取效果不佳问题,提出了一种基于关系提示的实体关系抽取方法。在构建单模块单步关系三重抽取模型(RPSS)的基础上,考虑不同层次的关系语义信息和符号级和特征级的关系提示信息,对实体和关系提示符进行联合编码,得到统一的全局表示;同时通过注意力机制挖掘不同嵌入之间的深层关联,构建三重交互矩阵,可在一个步骤中直接从单个模块中提取所有三元组。结果表明:所提方法在 NYT、WebNLG 两个基准数据集上实现了最佳的表现, F_1 分别达到了 93.3% 和 94.9%。

关键词: 实体关系抽取;注意力机制;联合编码

中图分类号: TP391.1 **文献标志码:** A

DOI: 10.7652/xjtuxb202503020 **文章编号:** 0253-987X(2025)03-0222-13

Research on Entity Relation Extraction Method Based on Relational Prompts with Single-Module Single-Step Approach

LIU Hui¹, ZHANG Zhi², WANG Qiyuan¹

(1. College of Electronic Engineering, National University of Defense Technology, Hefei 230037, China;

2. School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China)

Abstract: To address the issue of suboptimal performance in existing relation triplet extraction methods, caused by the oversight of relationship semantic information and the interdependence and indivisibility of elements in triplets, a novel entity relation extraction method based on relational prompts is proposed. Building upon a single-module single-step relation triple extraction model (RPSS), this method incorporates relationship semantic information at varying levels and symbol-level as well as feature-level relational prompt information. By jointly encoding entities and relational prompt symbols to derive a unified global representation, and employing attention mechanisms to delve into deep associations among different embeddings, a triple interaction matrix is constructed. This approach enables the extraction of all triplets directly from a single module in a single step. The proposed method demonstrates superior performance on benchmark datasets like NYT and WebNLG, achieving F_1 scores of 93.3% and 94.9%, respectively.

Keywords: entity relation extraction; attention mechanism; joint encoding

自然语言处理中最重要的一项信息提取任务就是从非结构化文本中提取实体和实体之间的关系^[1-2],从而生成结构化的关系三元组。通常,结构化三元组的形式是主体、关系、客体,其中主体和客体是通过关系联系在一起的实体。从非结构化文本中抽取结构化的三元组能够服务众多下游任务,如知识图谱^[3]、问答系统^[4]、医学任务^[5]、网络空间测绘^[6]等。

早期关于实体关系三元组抽取的研究^[7-8]主要包括两部分:识别实体和预测它们之间的关系。通过识别文本中的所有实体,然后枚举实体进行关系判断,这种方法被称为流水线方法。这种方法的优点是可以利用现有技术识别命名实体并对关系进行分类。然而,这种方法不但忽略了实体识别和关系预测之间的联系,而且一个子任务中的错误很容易传播到其他任务中,从而影响其应用。

随着深度学习在各种任务上的广泛应用,文献^[9-10]提出了以端到端方式识别实体及其关系的联合提取模型。例如,Wei 等^[11]提出的 CasRel 方法首先检测所有的主体,然后以映射函数的形式推断出相应的关系和对象。文献^[12]尝试对实体和关系进行统一表示。Kong 等^[13]提出的共同注意网络(CARE)方法采用并行编码策略学习不同子任务的解码方式。总的来说,这些模型通过分解和重组关系三重提取任务来实现端到端的关系三重提取。虽然这些方法取得了相当大的成功,但仍然存在一些问题:一方面,这些方法将关系三元抽取简单地视为实体对上的多标签分类任务,这种观点将关系视为无意义的标签,忽略了关系中丰富的语义信息,即存在关系标签语义缺失的问题;另一方面,实体和关系之间有着丰富而深刻的联系,三要素是不可分割的、相互依存的,忽略三元组完整性使模型容易丢失正确的关系组并引入无关噪声。这些缺点阻碍了当前关系提取模型对于关系语义的有效利用。

针对当前句子级实体关系抽取中关系标签语义缺失的问题,本文提出了一种基于关系提示的单模块单步骤实体关系抽取模型(RPSS)。关系提示方法能够提供丰富的标签语义信息,通过对句子和关系提示的统一编码构建实体关系交互矩阵;在解码阶段将关系三元组的三种元素作为一个整体进行抽取,实现了一次性抽取整个三元组。通过公开数据集实验验证了关系提示方法的有效性,表明了关系语义和关系三元组的整体性对于实体关系抽取的重要性。

1 基本思路

图 1 所示为传统实体关系抽取方法与本文所提方法的比较。其中,左侧为传统实体关系三元组抽取方法,首先通过不同的标注方法识别实体对,然后将其进行关系分类。这种方法虽然可以捕捉实体之间的相互联系,但却忽略了关系的语义信息以及实体与关系之间的相互联系。例如,关系“live in”的语义信息可以推断出关系三元组与人和地方有关,主体的实体类型应该是人物,客体的实体类型应该是地点。因此,关系的语义信息对于关系三元组的提取至关重要,越早构建实体和关系之间的链接结构,就能越深入地捕捉实体和关系之间的关联特征。为解决这一问题,本文引入了提示学习,并提出了基于关系提示的三元组抽取方法,如图 1 的右侧。与提示学习中的离散提示和连续提示相对应,本文提出了关系硬提示和关系软提示。关系硬提示提供 token 级的关系语义特征信息,关系软提示提供特征级的关系语义特征信息。引入关系提示具有以下优势:首先,构建的 token 级关系硬提示与句子中的实体标记是同构的,硬提示使模型能够同时对实体和关系进行编码,从而获得统一的表征,有利于同时挖掘实体和关系之间的深层交互关系;其次,构建的特征级关系软提示可以充分利用标签数据捕捉关系特征语义信息,从而在不同语境下提供丰富的关系特征和先验信息。

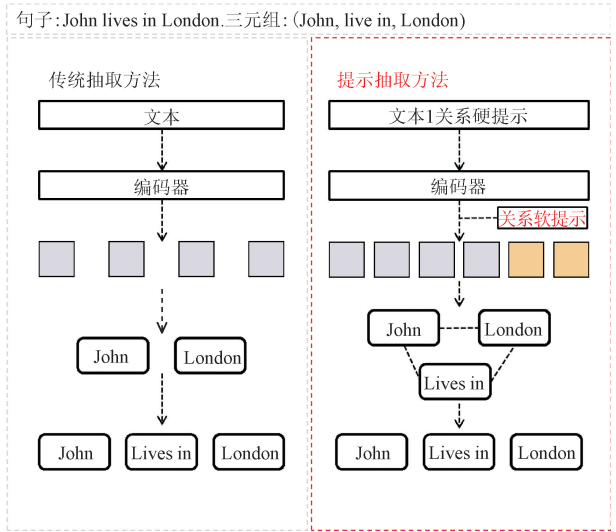


图 1 实体关系三元组抽方法的比较
Fig. 1 Comparison of RPSS models and methods

以往的方法在分解和重组关系三元组抽取任务时,多采取多模块结构或多步解码方法。如图 2 所

示,第一类多模块多步骤方法使用级联模块进行序列化的实体抽取和关系抽取;第二类多模块单步骤方法试图单独确定关系三元组的不同元素,然后根据潜在的相关性将它们结合起来。然而,这两种方法都受到分类错误和大量冗余信息存的困扰,这些问题的根源在于分解和重组任务忽略了关系三元组的三种元素,而主体、客体和关系是相互关联和不可分割的,抽取其中任一元素在没有对其他两个元素的信息完全感知的情况下都是不可信的。

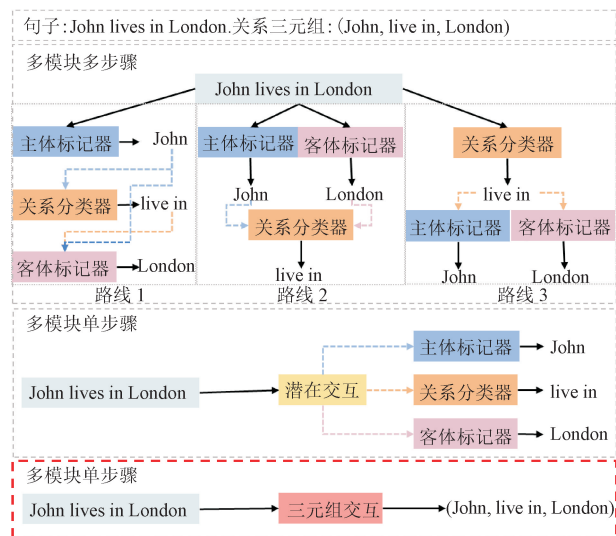


图 2 实体关系抽取模型结构的比较

Fig. 2 Comparison of model structure

为了解决这个问题,本文共同编码句子和关系提示,并使用 Transformer 结构网络来构建一个实体关系交互矩阵,用于挖掘不同元素之间的交互。通过这种方式,将实体抽取和关系判断集成到一个模块中,使模型能够同时关注到 3 种元素,而不仅仅在某一时刻只关注到实体或关系。这种结构具有以下优点:首先,实体和关系是共同编码和相互交互的,能够完整捕捉实体与关系之间的相互依赖;其次,交互矩阵能够在单个步骤中发现句子中所有的关系三元组,级联误差问题被有效地避免;第三,模型结构简单,易于训练。

基于以上分析,本文提出了一种基于关系提示的单模块单步骤实体关系抽取模型 (relational prompt-based single-module single-step model, RPSS),以充分挖掘和利用三元组三种元素之间的相互关联。

2 模型算法设计

首先,定义实体关系抽取任务。对于给定的句子 $S = \{x_1, x_2, \dots, x_N\}$ 含有 N 个词,实体关系抽取

任务的目标是抽取出句子 S 中潜在的所有关系三元组 $T = \{(s_l, r_l, o_l) \mid s, o \in R, r \in E\}_{l=1}^L$, 其中 L 表示关系三元组的数量, s_l, r_l, o_l 分别表示主体、客体和主客体之间的关系, $E = \{e_1, e_2, \dots, e_M\}$ 是一个包含所有的主体和客体的实体集合,其中 M 表示实体的数量, $R = \{r_1, r_2, \dots, r_K\}$ 是一个预定义的关系集合,其中 K 表示关系类型的数量。需要注意的是,关系集合 R 是已知的,实体和关系能够同时出现在不同的三元组中。

图 3 描述了本文提出的基于关系提示的单模块单步骤实体关系抽取方法框架。输入是句子和关系硬提示的连接,将预训练语言模型编码器和关系式软提示获得的特征嵌入进行特征融合。实体关系交互矩阵是从预训练语言模型编码器的自注意网络中学习得到的。矩阵的红色部分是实体与实体之间的交互,矩阵的绿色部分是客体与关系之间的交互,矩阵的蓝色部分是主体与关系之间的交互。之后通过交互矩阵就可以直接一步对齐识别出关系三元组。

2.1 关系提示

关系提示有两种主要类型:关系硬提示和关系软提示。

2.1.1 关系硬提示

关系硬提示有两个主要特点:人工选择和 token 级提示。关系硬提示使用自然语言中的词来表示关系集合 R 中的关系类型。例如,关系词 “contain” 代表关系类型 “/location/location/contains”; 关系词 “capital” 代表关系类型 “/location/country/capital”。有时存在多个语义相近的关系词可以表示一个关系类型。例如,用来表示关系类型 “/business/company/founders” 的关系词有 “creator”、“builder”、“founder”、“author” 等。从理论上讲,使用多个关系词共同表示一个关系类型可以提供更准确、更丰富的关系语义信息。为了简单起见,本文手动选择了一个语义信息最丰富的关系词,因为这些词在语义上差距并不大,选择不同的语义相关的提示词并不会导致结果的波动。

最后,得到一个关系硬提示序列 $R_{\text{HP}} = \{r_1, r_2, \dots, r_K\}$, 关系硬提示包含丰富语义信息的关系词,而不是无意义的关系标签,同时将实体和关系的结构统一为 token。

2.1.2 关系软提示

关系软提示有两个主要特点:集成生成和特征级提示。标注数据中的关系语义曾被严重忽视,标注数据中的三元组包含丰富的先验关系语义信息,

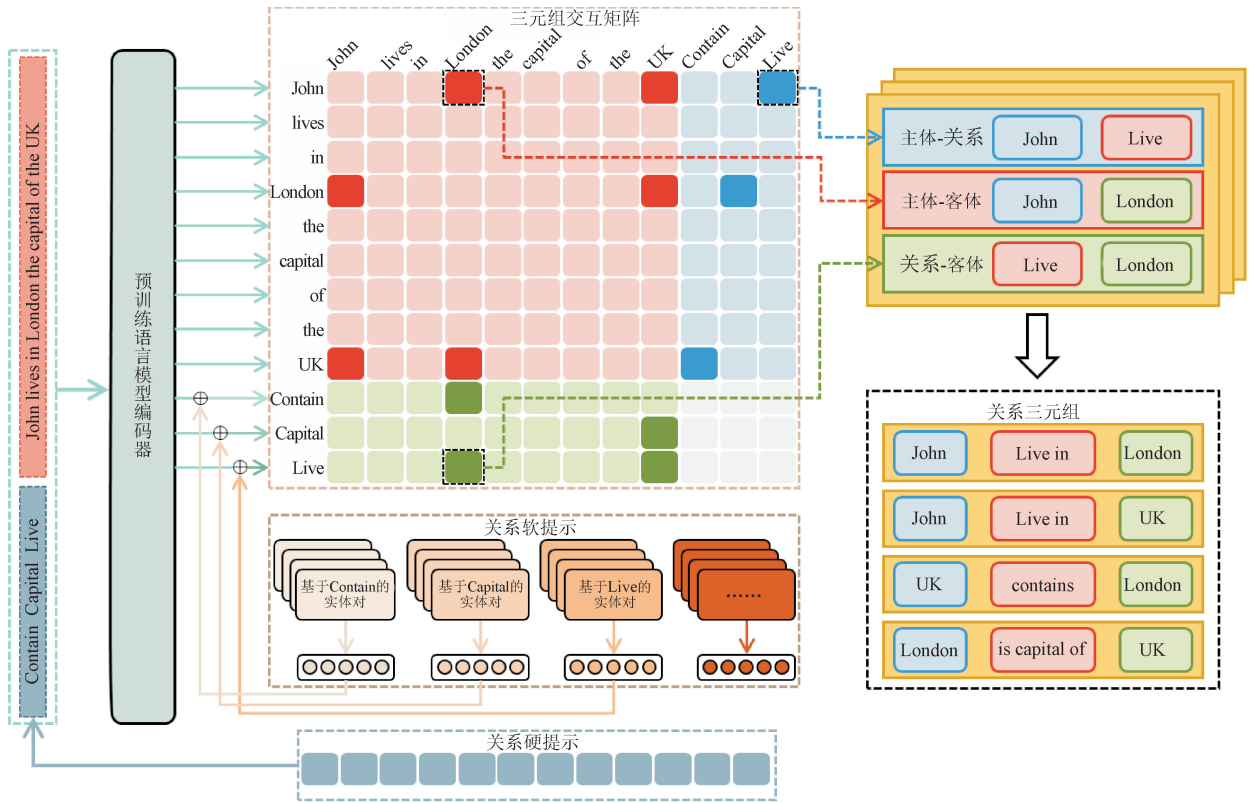


图 3 RPSS 的模型框架

Fig. 3 Model framework of RPSS

充足的标注数据可以构建关系特征语义信息,具有很强的泛化能力和广泛的适用性。通过对训练数据中所有特定关系实体对进行特征融合,可以得到包含丰富语义信息的先验关系嵌入式表示,这种嵌入式表示就是关系软提示。具体来说,对于一个关系 r_i ,实体对集合 $P_i = \{(s_l, o_l) | s, o \in E\}_{l=1}^{L_i}$ 是所有 r_i 特定实体对的集合^[1];句子编码完成后,通过平均池化抽取实体对所在的 token 的特征进行特征整合,得到每个实体对 (s_l, o_l) 的特征表示为 i_l ,然后通过整合所有实体对的 i_l 得到关系 a_i 的嵌入表示的语义特征。计算关系软提示的公式为

$$a_i = \frac{1}{L_i} \sum_{l=1}^{L_i} \text{AvgPool}(\text{Encoder}(s_l, o_l)) \quad (1)$$

式中:AvgPool 是平均池化方法;Encoder 表示文本编码器; a_i 表示关系软提示的嵌入; L_i 表示关系包含的实体对的数量。

最后,得到一组关系软提示 $R_{SP} = \{a_1, a_2, \dots, a_K\}$,其中每个向量 $a_i, i \in [1, K]$ 表示关系类型集合 R 中对应的关系类型的关系软提示。关系软提示是通过整合特定关系实体对的特征信息生成的。关系软提示利用现有的标注数据,从不同语境中充分挖掘特征级的关系语义信息。

2.2 RPSS 编码

在编码前,将关系硬提示和句子连接在一起,统一进行编码;编码后,关系软提示和统一表示进行特征融合。为了进行公平的比较,选择预训练语言 BERT-Base(Cased) 模型^[14] 为编码器,但理论上可以选择其他预训练语言模型作为编码器,目前基于 Transformer 架构的预训练语言模型发展迅速,包括但不限于 RoBERTa^[15]、BART^[16]、T5^[17]、GPT-4^[18] 等。更大的模型自然会带来更丰富的语义信息,但为了与其他基线模型进行公平比较,选择 BERT 作为与其他基线模型一致的编码器。与以往只对句子中进行编码的工作不同,本文提出的关系硬提示与句子具有相同的结构,这使得编码时可以对实体和关系进行联合编码,从而统一表示。联合编码的计算公式如下

$$\begin{cases} T = \text{Contat}(S, R_{HP}) \\ \mathbf{H} = \text{Encoder}(T) \end{cases} \quad (2)$$

式中:Contat 表示连接操作; T 表示关系和句子的连接之后的文本; $\mathbf{H} \in R^{(N+K) \times d}$ 是句子和关系的统一表示, d 是嵌入的维度。

接下来,嵌入特征级关系软提示,以获得包含关系语义信息的全局统一表示。具体来说,通过平均

池化关系软提示和统一表示的关系嵌入 $\mathbf{H}$ 来进行特征融合。关系软提示嵌入的计算公式如下

$$\mathbf{H} = \left[\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_N, \frac{\mathbf{h}_{N+1} + \mathbf{r}_1}{2}, \frac{\mathbf{h}_{N+K} + \mathbf{r}_K}{2} \right] \quad (3)$$

式中: $\mathbf{h}_{N+i}, \mathbf{r}_i \in \mathbf{R}^{1 \times d}, i \in [1, K]$ 。最后得到了实体和关系的全局统一表示 $\mathbf{H}$, $\mathbf{H}$ 包含丰富的关系语义信息。

2.3 三元组交互矩阵

在前一阶段获得统一表示之后,编码器通过使用自注意力技术^[19]挖掘出每个输入词之间的相关性,其中,基于 Transformer 的预训练模型采用 Transformer 网络层的整体架构。Transformer 网络层由两个子层组成:多头自注意网络和简单的前馈神经网络。语言模型强大的词嵌入表征能力主要得益于多头自注意机制,其使得模型能够同时关注来自不同位置 and 不同表征子空间的信息。本文选择多头自注意力机制的原因如下。首先,为了解决模型在对当前位置的信息进行编码时会过度关注自身位置的问题,多头注意力能够学习不同的特征信息,并最终将其整合在一起;其次,在一定程度上,头越多,整个模型的表现力就越强,也越能提高模型对注意力权重的合理分配;最后,多头的本质是多个独立的注意力网络并行计算,起到融合的作用,防止过拟合。多头自注意力机制主要用于捕捉 $\mathbf{H}$ 中每个输入嵌入之间的相互联系。每个注意力头由 3 个不同的线性变换网络组成,其将输入单词嵌入 $\mathbf{H}$ 转换为查询向量 $\mathbf{Q}$ 、键向量 $\mathbf{K}$ 和值向量 $\mathbf{V}$,然后计算 $\mathbf{Q}$ 与 $\mathbf{K}$ 的缩放点积,并使用非线性激活函数获得 $\mathbf{V}$ 的权重。相关计算公式如下

$$\mathbf{Q} = \mathbf{W}_{\text{query}} \mathbf{H} + \mathbf{b}_{\text{query}} \quad (4)$$

$$\mathbf{K} = \mathbf{W}_{\text{key}} \mathbf{H} + \mathbf{b}_{\text{key}} \quad (5)$$

$$\mathbf{V} = \mathbf{W}_{\text{value}} \mathbf{H} + \mathbf{b}_{\text{value}} \quad (6)$$

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right)\mathbf{V} \quad (7)$$

$$\mathbf{A} = \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \quad (8)$$

式中: $\mathbf{W}_{\text{query}}, \mathbf{W}_{\text{key}}, \mathbf{W}_{\text{value}} \in \mathbf{R}^{(N+K) \times (N+K)}$ 是可训练的权重; $\mathbf{b}_{\text{query}}, \mathbf{b}_{\text{key}}, \mathbf{b}_{\text{value}} \in \mathbf{R}^d$ 是可训练的偏置; $\text{Attention}()$ 和 $\text{softmax}()$ 分别为注意力函数和归一化指数函数。

参数更新规则如下:第一步,在模型初始化过程中,模型参数根据正态分布随机分布;第二步,在模型训练过程中,为了最小化损失函数,通过反向传播不断修改参数,最终实现收敛。在多层网络中,每一个 Transformer 层通过注意机制从上一层的输出中生成新的嵌入,这使得 Transformer 网络能够充分捕捉不同嵌入之间丰富的深度相关性。用 $\mathbf{H}_i$ 表示

第 i 个 Transformer 层的输出。在此基础上,将实体和关系提示统一到输入嵌入 $\mathbf{H}$ 中, $\mathbf{H}_i$ 包含了实体和关系之间丰富的内在关联。Transformer 结构的 BERT 具有强大的捕获深层特征的能力。使用 BERT 的最后一层来构建三元组交互矩阵。BERT 的 Transformer 网络共有 12 层,将最后一层定义为 $\mathbf{H}_{11}$ 。从 $\mathbf{H}_{11}$ 中得到嵌入向量矩阵的 $\mathbf{Q}$ 和 $\mathbf{K}$,然后对 $\mathbf{Q}$ 和 $\mathbf{K}$ 的多个注意头的点乘运算结果进行平均,并直接使用 sigmoid 非线性激活函数得到交互矩阵的结果。获取三元组交互矩阵的公式如下

$$\mathbf{I} = \text{sigmoid}\left(\frac{1}{L} \sum_i^L \frac{\mathbf{Q}_i \mathbf{K}_i^T}{\sqrt{d}}\right) \quad (9)$$

式中: L 表示多头自注意网络中的头数,在 BERT 中通常为 12; $\mathbf{I} \in \mathbf{R}^{(N+K) \times (N+K)}$ 是三元组交互矩阵。如果 $\mathbf{I}(\cdot)$ 的值超过了预定义的阈值 σ ,实体-实体对 (e_a, e_b) 或实体-关系对 (e, r) 将被视为有效,否则,将被视为无效。

最后,得到用于三元组对齐抽取的交互矩阵 $\mathbf{I}$ 。

2.4 RPSS 解码

如图 3 所示,在交互矩阵中,有两类交互:实体-实体交互(矩阵的红色部分)和实体-关系交互(矩阵的蓝色和绿色部分),通过对齐这两类交互直接抽取关系三元组。实体-实体交互用于提取有效的实体对,实体-实体对用于构建有效的关系三元组。给定句子 S 中的两个实体 e_i, e_j ,当交互矩阵中位置 (i, j) 的值为 True 时,实体对 (e_i, e_j) 是有效的。只有有效的实体对 (e_i, e_j) 才有可能有效的关系三元组,在这里既可以是 (e_i, r, e_j) ,也可以是 (e_j, r, e_i) 。例如,在图 3 中,交互矩阵中的位置 $(0, 3)$ 是 True,这意味着实体对 (John, London) 和 (London, John) 是有效的,其对构成关系三重 (John, lives in, London) 提供支持。通过三元组交互矩阵中的实体-实体交互,得到实体-实体对集。实体-关系交互识别每个关系的所有相关实体,并形成有效的实体-关系对用于构建有效的关系三重。给定一个关系 r ,当存在由实体 e 和关系 r 组成的三元组时,认为实体 e 和关系 r 是相互作用的。此外,由于关系的方向性,不应该对称地提取主体-关系对和客体-关系对。为了区分主体和客体,将实体-关系交互分为两部分:主体-关系交互(图 3 矩阵右上方的蓝色部分)和关系-客体交互(图 3 矩阵左下方的绿色部分)。例如,在图 3 中,交互矩阵中的位置 $(0, 12)$ 是 True,这意味着主体-关系对 (John, lives in) 是有效的,交互矩阵中的位置 $(12, 3)$ 是 True,这意味着关系-客体对 (lives in, London) 是有效的。他们对构建关系三元组 (John,

lives in, London)提供支持。通过交互矩阵中的实体-关系交互,得到所有主体-关系对和客体-关系对的集合。

这种简单高效的解码方法考虑了所有可以形成有效关系三元组的实体和关系。因此,该方法不仅能有效对齐三元组,还能在复杂场景中有效抽取嵌套三元组。如图 3 中的例子所示,RPSS 模型识别出了句子“John lives in London, the capital of the UK.”中的 4 个三元组,包括两个 SEO 类型的嵌套三元组:(John, lives in, London)和(John, lives in, UK)以及两个 EPO 类型的嵌套三元组:(London, is capital of, UK)和(UK, contains, London)。

2.5 训练策略

在模型训练过程中,计算二元交叉熵损失作为 RPSS 的目标函数,公式如下

$$L = - \frac{1}{(N + K)^2} \sum_i^{N+K} \sum_j^{N+K} (y_{i,j} \lg I_{i,j} + (1 - y_{i,j}) \lg (1 - I_{i,j}))$$

(10)

式中: $y_{i,j}$ 是对应于位置 (i, j) 的正确标签。

3 实验设计与结果分析

在本节中进行了多个实验来验证所提出的 RPSS 的效果,然后详细讨论实验结果。

3.1 数据集与评测指标

为了验证本文 RPSS 模型的有效性,在两个公

开的实体关系抽取基准数据集 NYT^[20] 和 WebNLG^[21] 上进行实验。采用精确率、召回率、 F_1 作为模型的评价指标。

3.2 基线方法

为了进行比较,选择 14 个模型作为基线,包括 SOTA 模型 CasRel^[11]、TPLinker^[22]、R-BPtrNet^[23]、TDEER^[24]、PFN^[25]、PRGC^[26]、CARE^[13]、GRTE^[27] 和 OneRel^[28]。上述基线模型的结果均来自其对应的文献。

3.3 实现细节

实验使用含有 108 M 参数量的 Cased BERT base 作为编码器,包含 12 个 Transformer 层,每层都有 12 个注意力头。文本的最大长度设置为 100, NYT 和 WebNLG 数据集在实验中的批大小分别设置为 24 和 6,实验总共进行 100 轮,使用 Pytorch 框架实现了提出的方法,所有的参数通过 Adam 优化器^[29]进行优化,NYT 和 WebNLG 数据集的学习率分别设置为 3×10^{-5} 和 5×10^{-5} ,权重衰减比例设置为 0.01,阈值 σ 设置为 0.5。所有的训练和测试都在一台配有 InterXeon(R) Gold 6230R CPU@2.10 GHz、128 G 内存、NVIDIA GeForce RTX3090 GPU 和 Windows10 专业版系统的计算机上进行。

3.4 实验结果与分析

3.4.1 总体结果分析

表 1 展示了 RPSS 与其他 14 个基线的对比结果。

表 1 RPSS 与其他基线模型的结果对比
Table 1 Comparison of results between RPSS and other baseline models

模型	NYT 数据集			WebNLG 数据集		
	精确率/%	召回率/%	F_1 /%	精确率/%	召回率/%	F_1 /%
NovelTagging ^[30]	62.4	31.7	42.0	52.5	19.3	28.3
CopyRE ^[31]	61.0	56.6	58.7	37.7	36.4	37.1
GraphRel ^[32]	63.9	60.0	61.9	44.7	41.1	42.9
OrderCopyRE ^[33]	77.9	67.2	72.1	63.3	59.9	61.6
ETLSpan ^[34]	84.9	72.3	78.1	84.0	91.5	87.6
CasRel ^[11]	89.7	89.5	89.6	93.4	90.1	91.8
TPLinker ^[22]	91.3	92.5	91.9	91.8	92.0	91.9
SPN ^[35]	93.3	91.7	92.5	93.1	93.6	93.4
TDEER ^[24]	93.0	92.1	92.5	93.8	92.4	93.1
R-BPtrNet ^[23]	92.7	92.5	92.6	93.7	92.8	93.3
PFN ^[25]			92.4			93.6
PRGC ^[26]	93.3	91.9	92.6	94.0	92.1	93.0
GRTE ^[27]	92.9	93.1	93.0	93.7	94.2	93.9
CARE ^[13]			92.6			93.9
OneRel ^[28]	92.8	92.9	92.8	94.1	94.4	94.3
RPSS	93.5	93.2	93.3	94.7	95.1	94.9

注:表中加粗数据表示最优值。

从表 1 可知,许多基线的 F_1 都超过了 90%,尤其是在 WebNLG 数据集上,表现出了优于人类水平的性能。与基线相比,RPSS 在两个数据集的所有指标上都表现得更好, F_1 得分的绝对值分别提高了 0.5%和 0.6%。结果表明了提出的 RPSS 的优越性和有效性。RPSS 在所有指标上都超过了最具竞争力的 GRTE 和 OneRel 模型,原因有两个。首先,关系语义信息对于关系三元组抽取至关重要。关系提示方法为模型引入了丰富的关系语义信息,然而,GRTE 和 OneRel 在抽取过程中只维护关系矩阵,并为矩阵分配可训练的权重,无法充分利用和挖掘关系语义信息。其次,RPSS 结合实体和关系的依赖关系构建实体关系交互矩阵,与传统的填表法相比,复杂度从 $O(N \times N \times K)$ 降至 $O((N+K)^2)$ 并减少了冗余噪声信息。最后,RPSS 是单模块、单步骤的。RPSS 同时与关系三元组的所有 3 个元素交互,检测主体、客体和关系,并在一个步骤中提取三元组,而其他方法需要多步交互,一步步提取实体和关系。本文发现,同时挖掘三元间的交互能取得更好的结果,并有效避免级联误差传播问题。

此外,WebNLG 比 NYT 数据集更具挑战性,因为其包含了更多的关系类型,而训练数据却很少。然而,本文提出的方法不仅取得了更显著的进步,而且首次在 3 个指标上同时达到了 95%。一般来讲,关系类型越多,抽取正确的关系类型难度越大,但 RPSS 将实体和关系整合到一个模块中,增强了交互和约束,使得模型能够同时关注三元组的所有 3 个元素,而不是仅在某时刻关注到关系或实体,这也是 RPSS 在 WebNLG 上比其他模型取得更大改进的主要原因。所有这些都表明,本文提出的 RPSS 具有良好的鲁棒性,能够适应不同的数据集,在处理多种关系类型的复杂情况时优势明显。

3.4.2 复杂场景结果分析

为了验证 RPSS 抽取嵌套三元组和多重三元组的能力,在本节进行了额外的实验。选择了 8 个功能强大的模型进行比较,结果如表 2 和表 3 所示。表中,Normal、EPO、SEO、SOO 分别表示普通三元组、某实体与多个实体存在关系、两个实体之间存在多个关系、两个实体之间存在字符重叠, N 为一个句子中三元组的数量。

表 2 RPSS 与其他基线模型在 NYT 数据集上不同复杂场景下的 F_1 结果对比

Table 2 Comparison of F_1 results between RPSS and other baseline models in different complex scenarios on NYT dataset

模型	$F_1/\%$								
	Normal	EPO	SEO	SOO	$N=1$	$N=2$	$N=3$	$N=4$	$N \geq 5$
CasRel	87.3	91.4	92.0	77.0	88.2	90.3	91.9	94.2	83.7
TPLinker	90.1	93.4	94.0	90.1	90.0	92.8	93.1	96.1	90.0
SPN	90.8	94.0	94.1		90.9	93.4	94.2	95.5	90.6
R-BPtrNet	90.4	94.4	95.2		89.5	93.1	93.5	96.7	91.3
TDEER	90.8	94.1	94.5		90.8	92.8	94.1	95.9	92.8
PRGC	91.0	94.0	94.5	81.8	91.1	93.0	93.5	95.5	93.0
GRTE	91.1	94.4	95.0		90.8	93.7	94.4	96.2	93.4
OneRel	90.6	94.8	95.1	90.8	90.5	93.4	93.9	96.5	94.2
RPSS	91.2	95.2	95.2	90.5	91.1	94.4	94.4	96.4	94.4

注:表中加粗数据表示最优值。

表 3 RPSS 与其他基线模型在 WebNLG 数据集上不同复杂场景下的 F_1 结果对比

Table 3 Comparison of F_1 results of RPSS and other baseline models in different complex scenarios on WebNLG dataset

模型	$F_1/\%$								
	Normal	EPO	SEO	SOO	$N=1$	$N=2$	$N=3$	$N=4$	$N \geq 5$
CasRel	89.4	92.2	94.7	90.4	89.3	90.8	94.2	92.4	90.9
TPLinker	87.9	92.5	95.3	86.0	88.0	90.1	94.6	93.3	91.6
SPN	89.5	94.1	90.8		89.5	91.3	96.4	94.7	93.8
R-BPtrNet	89.5	93.9	96.1		88.5	91.4	96.2	94.9	94.2
TDEER	90.7	93.5	95.4		90.5	93.2	94.6	93.8	92.3
PRGC	90.4	93.6	95.9	94.6	89.9	91.6	95.0	94.8	92.8
GRTE	90.6	94.5	96.0		90.6	92.5	96.5	95.5	94.4
OneRel	91.9	94.7	95.4	94.9	91.4	93.0	95.9	95.7	94.5
RPSS	93.7	95.4	97.1	96.5	93.1	94.7	96.9	95.6	96.0

注:表中加粗数据表示最优值。

可以看出,RPSS 在所有 18 个复杂场景的子集下,在 15 个子集中的表现优于之前的基线方法,取得了一定的性能提升。在其余 3 个子集中,与最佳结果的差距很小,可以忽略不计。此外,RPSS 在子集 Normal、SEO、EPO 和 $N=1,2,3$ 上也取得了可喜的性能提升,这些子集占数据集的比例较大。

实验结果表明,RPSS 在常规场景和复杂场中的表现同样出色,表明了该模型的鲁棒性。总之,两个进一步的实验表明了 RPSS 在复杂场景中的优势。这是因为模型构建了一个实体关系交互矩阵,可以捕捉多个关系三元组和不同关系类型嵌套模式下的复杂交互,这对于处理复杂场景中的关系三元组提取至关重要。此外,所构建的实体关系交互矩阵中的每一项都表示是否可以建立实体-实体对或实体-关系对,考虑了所有可能的实体-实体和实体-关系对,同时,在解码时不遗漏任何一个三元组,使得模型能够高效抽取嵌套三元组和一个句子中多个关系三元组。

3.4.3 子任务结果分析

RPSS 通过交互矩阵实现单模块单步提取实体

和关系,能更好地捕捉实体-实体和实体-关系之间的相互联系。此外,单模块、单步提取模型避免了级联错误和暴露偏差。为了验证这些特性,进一步研究了 RPSS 在两个子任务(实体对提取和关系提取)上的效果。选择 CasRel、SPN 和 PRGC 作为基线,首先,都是近期的 SOTA 模型;其次,分别是经典的多模块多步骤和多模块单步骤模型;此外,在实体对对齐和关系分类方面非常有效。

实验对比结果如表 4 所示。在多数情况下,RPSS 在所有子任务上的表现都优于之前的基线方法。尽管在某些情况下无法达到最佳性能,但在最终的三元组抽取任务上,RPSS 的单模块单步骤提取方法的整体性能优越,并比之前的优秀模型有显著的性能提升,这表明本文提出的单模块、单步提取方法避免了多步骤和多模块之间的级联错误。这再次证实了将不同元素的提取合并到单模块中的动机是正确的。此外,单模块单步抽取解码方式允许两个子任务相互联系并相互促进。

表 4 RPSS 与其他基线模型在不同子任务上的对比结果							
Table 4 Comparison results of RPSS and other baseline models on different subtasks							
模型	子任务	NYT 数据集			WebNLG 数据集		
		精确率/%	召回率/%	F_1 /%	精确率/%	召回率/%	F_1 /%
Casrel	(s, o)	89.2	90.1	89.7	95.3	91.7	93.5
	r	96.0	93.8	94.9	96.6	91.5	94.0
	(s, r, o)	89.7	89.5	89.6	93.4	90.1	91.8
SPN	(s, o)	93.2	92.7	92.9	95.0	95.4	95.2
	r	96.3	95.7	96.0	95.2	95.7	95.4
	(s, r, o)	93.3	91.7	92.5	93.1	93.6	93.4
PRGC	(s, o)	94.0	92.3	93.1	96.0	93.4	94.7
	r	95.3	96.3	95.8	92.8	96.2	94.5
	(s, r, o)	93.3	91.9	92.6	94.0	92.1	93.0
RPSS	(s, o)	93.6	93.6	93.6	96.3	96.6	96.5
	r	97.0	95.5	96.3	96.3	96.1	96.2
	(s, r, o)	93.5	93.2	93.3	94.7	95.1	94.9

注:表中加粗数据表示最优值。

3.4.4 消融实验分析

本节进行了消融实验,以验证关系提示方法在 RPSS 模型中的效果。

(1)关系提示的影响关系。提示方法的提出是为了充分利用关系的语义信息,而不仅仅是无意义的关系标签。在关系提示方法的消融实验中,删除

了关系软提示和关系硬提示,并使用无意义占位符来表示每个关系的标签,即[unused1]、[unused2]、[unused3],这些无意义的占位符在微调阶段随机初始化。模型会给它们分配一个随机的词嵌入,而不会考虑从预训练中学到的语义信息。RPSS 消融实验结果见表 5。

表 5 RPSS 的消融实验结果
Table 5 Ablation results of RPSS

数据集	模型	精确率/%	召回率/%	F_1 /%
NYT	RPSS	93.5	93.2	93.3
	不含 Relational Prompt	92.6	92.3	92.4
	不含 Relational Hard-Prompt	92.9	92.7	92.8
	不含 Relational Soft-Prompt	93.0	92.9	92.9
WebNLG	RPSS	94.7	95.1	94.9
	不含 Relational Prompt	78.2	60.6	67.9
	不含 Relational Hard-Prompt	92.9	85.1	88.9
	不含 Relational Soft-Prompt	94.6	94.1	94.3

注:表中加粗数据表示最优值。

从表 5 可以看出,在没有关系提示的情况下,各项指标有不同程度的下降,尤其是在 WebNLG 数据集上,下降幅度接近 30%。由于 WebNLG 的关系类型较多,多达 171 种,因此仅通过无意义的标签与实体交互是无法获得实体与关系之间的深层关联的。模型无法关注与实体密切相关的关系类型,导致精确度和召回率都大幅下降。相比之下,NYT 数据集上的关系类型较少,只有 24 种,因此模型性能的下降并不明显。

实验验证了关系提示方法的有效性,同时也表明该方法具有很强的普适性和实用价值,因为真实场景中的关系数量很大。

(2)关系硬提示和关系软提示的影响。在关系硬提示的消融实验中,使用无意义占位符代替关系式硬提示。在关系软提示的消融实验中,使用随机初始化的关系软提示,而不是来自特征融合的关系软提示。

从表 5 可以看到,缺少其中任何一种提示都会导致模型性能下降,这说明两种关系提示对于构建

实体-关系交互矩阵的重要性。在没有 token 级关系硬提示的情况下,实体和关系类型之间在模型的初始阶段没有注意力交互。在模型的初始阶段,实体和关系标签之间缺乏注意力交互和特征流,也就无法在当前上下文中构建实体和关系之间的深度关联,从而导致性能下降。在缺乏特征级关系软提示的情况下,实体与关系的交互只能停留在浅层次,这适用于简单三元组的情况,但无法满足具有多个三元组和三元组重叠的复杂场景的需求。关系软提示是不同语境下关系特征的融合,对于多关系三元组和嵌套三元组的特征对齐和深度交互非常重要。

以上实验结果表明,关系硬提示和关系软提示对于建立实体和关系之间的深度交互与依赖至关重要。

3.4.5 效率分析

在训练阶段所有实验均采用相同的硬件配置,将 NYT 和 WebNLG 数据集的批量大小分别统一设置为 12、6,在推理阶段将批量大小设置为 1。PRGC、GRTE、OneRel 和 RPSS 模型的复杂度、浮点运算次数及参数量见表 6。

表 6 模型在两个数据集上计算效率的结果对比
Table 6 Comparison of computational efficiency of the model on two datasets

数据集	模型	复杂度	浮点运算次数/ 10^9 s^{-1}	参数量/ 10^6	训练时间/ms	推理时间/ms	F_1 /%
NYT	PRGC	$O(N^2)$	77.5	112.5	1286	34	92.6
	GRTE	$O(KN^2)$	136.6	119.4	1419	40	93.0
	OneRel	$O(KN^2)$	80.7	113.4	1162	31	92.8
	RPSS	$O((N+K)^2)$	67.3	108.3	648	20	93.3
WebNLG	PRGC	$O(N^2)$	54.5	112.6	326	36	93.0
	GRTE	$O(KN^2)$	139.1	122.1	251	48	93.9
	OneRel	$O(KN^2)$	45.2	113.4	324	44	94.3
	RPSS	$O((N+K)^2)$	36.1	108.3	151	32	94.9

注:表中加粗数据表示最优值。

在复杂度方面,RPSS 明显优于相同的基于表格填充的 GRTE 和 OneRel 方法。由于 RPSS 是通过 BERT 的最后一层进行解码的,没有为解码设计额外的网络,因此在浮点运算次数和参数数量方面与其他模型相比具有明显优势。从参数量来看,4 个模型都使用 BERT 作为编码器,BERT 的参数量为 108 M。去除 BERT 的参数量后,与其他模型相比,RPSS 的参数量非常小。在时间方面,由于 RPSS 的浮点运算次数和参数量相较其他模型都较小,因此在训练时间和推理时间上都小于其他基线模型。RPSS 的训练速度是其他模型的 2 倍,推理速度是其他模型的 1~2 倍,同时获得了更好的 F_1 。尽管 PRGC 的复杂度较低,但 RPSS 的效率和 F_1 仍有显著优势,即训练时间提高了 2 倍, F_1 在 WebNLG 数据集上提高了 1.9%。此外,PRGC 的复杂度较低是因为其是一种基于标记的方法,无法考虑到所有实体和关系,容易出现级联错误传播或遗漏嵌套三元组,表 2 和 3 中所有复杂场景下 RPSS 的性能都优于 PRGC 就验证了这一点。

从图 4 可以看出,RPSS 模型能更好、更快地收敛,这体现了 RPSS 在收敛速度方面的优势,即 RPSS 在效率方面的进步。总的来说,本文方法具有更低的复杂度和更高的效率。这主要有两个原因:首先,模型结构简单,解码前没有多余的其他网络层,直接利用预训练语言模型的最终编码层构建实体关系交互矩阵;第二,单步提取,解码简单。RPSS 在解码阶段只通过一个实体交互矩阵进行解码,不像其他方法按不同顺序分别抽取关系三元组的不同元素,解码方法复杂,本文提出的解码方法简单有效。

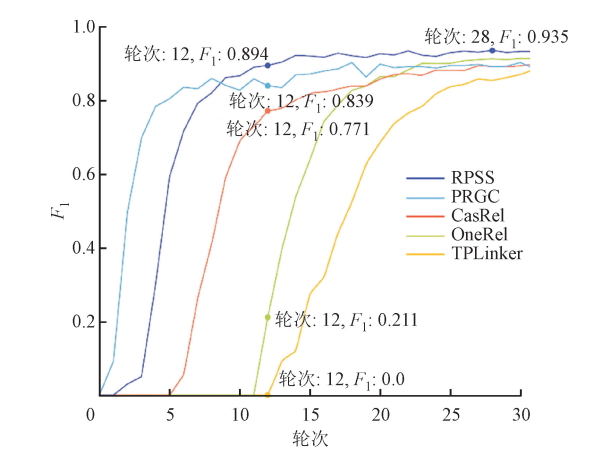


图 4 RPSS 模型的收敛速度对比

Fig. 4 Comparison of convergence speeds of RPSS model

3.4.6 案例分析

在本节中分别选取了数据集 NYT 和数据集 WebNLG 中包含嵌套三元组和多个关系三元组的案例。

选取的 NYT 数据集案例为:“In perhaps the most ambitious Mekong cruise attempt, Impulse Tourism, an operator based in Chiang Mai, Thailand, is organizing an expedition starting in November in Jinghong, a small city in the Yunnan province in China.”。选取的 WebNLG 数据集的案例为:“American test pilot Alan Shepard died in California and was born in New Hampshire.”。

利用 RPSS 将上述案例的实体关系三元组交互矩阵可视化,结果如图 5 和图 6 所示,其中实体-实体交互、主体-关系交互和客体-关系交互分别用红色部分、蓝色部分和绿色虚线框表示。可以看出,RPSS 能精确定位有效的实体对和有效的实体-关系对,并提供较高的置信度。由于该模型是单模块单步骤的,因此增强了实体和关系之间的交互和依赖。

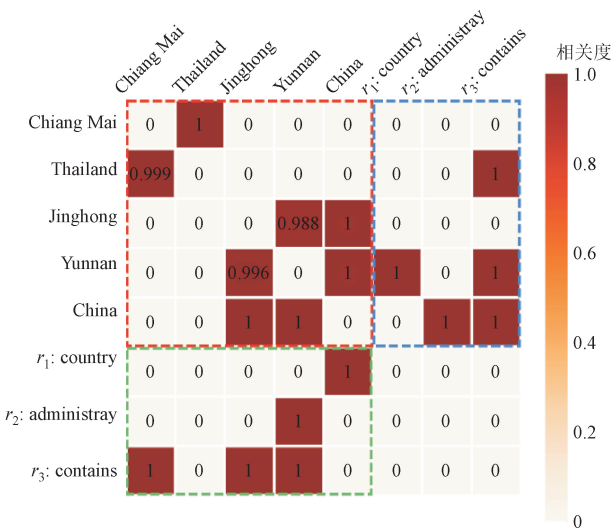


图 5 NYT 案例的交互矩阵可视化

Fig. 5 Interaction matrix visualization of NYT case

通过矩阵中 3 种元素的关联,可以一步提取出所有关系三元组。通过图 5 可以提取出 NYT 案例中包含的 6 个三元组:(China,/location/location/contains, Jinghong), (China,/location/location/contains, Yunnan), (China,/location/country/administrative_divisions, Yunnan), (Yunnan,/location/administrative_divisions/country, China), (Thailand,/location/location/contains, ChiangMai) and (Yunnan,/location/location/contains, Jinghong)。

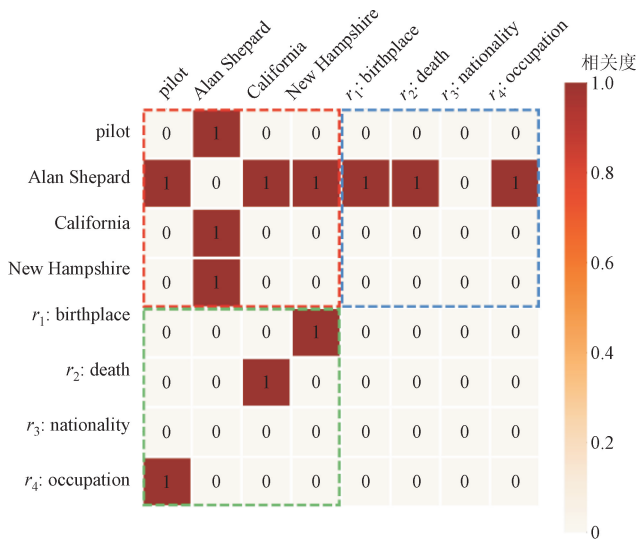


图 6 WebNLG 案例的交互矩阵可视化
Fig. 6 Interaction matrix visualization of the WebNLG case

通过图 6 可以提取出 WebNLG 案例中的 3 个关系三元组: (Alan Shepard, death, California), (Alan Shepard, occupation, pilot) 和 (Alan Shepard, birthplace, New Hampshire)。

4 结 论

针对当前实体关系抽取中关系标签语义缺失的问题,本文提出了一种基于关系提示的单模块单步骤实体关系抽取方法。关系提示方法能够提供丰富的标签语义信息,通过对句子和关系提示的统一编码构建实体关系交互矩阵。在解码阶段将关系三元组的 3 种元素作为一个整体进行抽取,实现了一次性抽取整个三元组。实验验证了关系提示方法的有效性,表明了关系语义和关系三元组的整体性对于实体关系抽取的重要性。实验结果表明该方法实现了最佳性能。

未来,我们将在现有研究的基础上,继续开展实体关系抽取方法研究。本文的方法需要手动选择关系提示词,这对于具有很多关系的数据来说是复杂的。未来,我们将尝试设计一个关系的自动语言表达式,提升算法选择关系提示词的自适应性。此外,模型在少样本情况下的效能如何还需进一步探索。

参考文献:

[1] 耿汝山, 陈艳平, 唐瑞雪, 等. 跨度语义增强的命名实体识别方法 [J]. 西安交通大学学报, 2022, 56(7): 118-126.
GENG Rushan, CHEN Yanping, TANG Ruixue, et

al. Named entity recognition based on span semantic enhancement [J]. Journal of Xi'an Jiaotong University, 2022, 56(7): 118-126.

[2] 秦涛, 杜尚恒, 常元元, 等. ChatGPT 的工作原理、关键技术及未来发展趋势 [J]. 西安交通大学学报, 2024, 58(1): 1-12.
QIN Tao, DU Shangheng, CHANG Yuanyuan, et al. Principles, key technologies and emerging trends of ChatGPT [J]. Journal of Xi'an Jiaotong University, 2024, 58(1): 1-12.

[3] NAYAK T, MAJUMDER N, GOYAL P, et al. Deep neural approaches to relation triplets extraction: a comprehensive survey [J]. Cognitive Computation, 2021, 13(5): 1215-1232.

[4] ZHAO Tianyang, YAN Zhao, CAO Yunbo, et al. Asking effective and diverse questions: a machine reading comprehension based framework for joint entity-relation extraction [C]//Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence Main track. California, USA: IJCAI, 2021: 3948-3954.

[5] ZENG Zheni, YAO Yuan, LIU Zhiyuan, et al. A deep-learning system bridging molecule structure and biomedical text with comprehension comparable to human professionals [J]. Nature Communications, 2022, 13(1): 862.

[6] 薛鹏飞, 沈毅, 胡森, 等. 基于规则的域名 WHOIS 信息抽取技术研究 [J]. 信息对抗技术, 2023, 2(1): 66-77.
XUE Pengfei, SHEN Yi, HU Miao, et al. Rule-based WHOIS information extraction technology [J]. Information Countermeasure Technology, 2023, 2(1): 66-77.

[7] GORMLEY M R, YU Mo, DREDZE M. Improved relation extraction with feature-rich compositional embedding models [C]//Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: ACL, 2015: 1774-1784.

[8] LIN Yankai, SHEN Shiqi, LIU Zhiyuan, et al. Neural relation extraction with selective attention over instances [C]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL, 2016: 2124-2133.

[9] ZHANG Yunqi, CHEN Yubo, HUANG Yongfeng. RelU-net: syntax-aware graph U-net for relational triple extraction [C]//Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: ACL, 2022: 4208-4217.

- [10] WADDEN D, WENBERG U, LUANYI, et al. Entity, Relation, and event extraction with contextualized span representations [C]//Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Stroudsburg, PA, USA: ACL, 2019: 5784-5789.
- [11] WEI Zhepei, SU Jianlin, WANG Yue, et al. A novel cascade binary tagging framework for relational triple extraction [C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL, 2020: 1476-1488.
- [12] TANG Wei, XU Benfeng, ZHAO Yuyue, et al. UniRel: unified representation and interaction for joint relational triple extraction [C]//Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: ACL, 2022: 7087-7099.
- [13] KONG Wenjun, XIA Yamei. CARE: co-attention network for joint entity and relation extraction [C]//Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024). Stroudsburg, PA, USA: ACL, 2024: 2864-2870.
- [14] DEVLIN J, CHANG Mingwei, LEE K, et al. BERT: pre-training of deep bidirectional transformers for language understanding [C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA, USA: ACL, 2019: 4171-4186.
- [15] LIU Yinhan, OTT M, GOYAL N, et al. RoBERTa: a robustly optimized BERT pretraining approach [EB/OL]. (2019-07-26)[2024-06-01]. <https://arxiv.org/abs/1907.11692>.
- [16] LEWIS M, LIU Yinhan, GOYAL N, et al. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension [C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL, 2020: 7871-7880.
- [17] RAFFEL C, SHAZEER N, ROBERTS A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer [J]. The Journal of Machine Learning Research, 2020, 21(1): 5485-5551.
- [18] OpenAI. GPT-4 technical report [EB/OL]. (2024-03-04)[2024-06-01]. <https://arxiv.org/abs/2303.08774>.
- [19] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need [C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook, NY, USA: Curran Associates Inc., 2017: 6000-6010.
- [20] RIEDEL S, YAO Limin, MCCALLUM A. Modeling relations and their mentions without labeled text [C]//Machine Learning and Knowledge Discovery in Databases. Berlin, Germany: Springer Berlin Heidelberg, 2010: 148-163.
- [21] GARDENT C, SHIMORINA A, NARAYAN S, et al. Creating training corpora for NLG micro-planners [C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL, 2017: 179-188.
- [22] WANG Yucheng, YU Bowen, ZHANG Yueyang, et al. TPLinker: single-stage joint extraction of entities and relations through token pair linking [C]//Proceedings of the 28th International Conference on Computational Linguistics. Stroudsburg, PA, USA: ACL, 2020: 1572-1582.
- [23] CHEN Yubo, ZHANG Yunqi, HU Changran, et al. Jointly extracting explicit and implicit relational triples with reasoning pattern enhanced binary pointer network [C]//Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA, USA: ACL, 2021: 5694-5703.
- [24] LI Xianming, LUO Xiaotian, DONG Chenghao, et al. TDEER: an efficient translating decoding schema for joint extraction of entities and relations [C]//Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: ACL, 2021: 8055-8064.
- [25] YAN Zhiheng, ZHANG Chong, FU Jinlan, et al. A partition filter network for joint entity and relation extraction [C]//Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: ACL, 2021: 185-197.
- [26] ZHENG Hengyi, WEN Rui, CHEN Xi, et al. PRGC: potential relation and global correspondence based joint relational triple extraction [C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. Stroudsburg, PA, USA: ACL, 2021: 6225-6235.
- [27] REN Feiliang, ZHANG Longhui, YIN Shujuan, et al. A novel global feature-oriented relational triple extraction model based on table filling [C]//Proceedings of

- the 2021 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: ACL, 2021: 2646-2656.
- [28] SHANG Yuming, HUANG Heyan, MAO Xianling. OneRel: joint entity and relation extraction with one module in one step [C]//Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI Press, 2022: 11285-11293.
- [29] KINGMA D P, BA J. Adam: a method for stochastic optimization [EB/OL]. (2014-12-22) [2024-07-01]. <https://www.semanticscholar.org/paper/Adam%3A-A-Method-for-Stochastic-Optimization-Kingma-Ba/a6cb366736791bcccc5c8639de5a8f9636bf87e8>.
- [30] ZHENG Suncong, WANG Feng, BAO Hongyun, et al. Joint extraction of entities and relations based on a novel tagging scheme [C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL, 2017: 1227-1236.
- [31] ZENG Xiangrong, ZENG Daojian, HE Shizhu, et al. Extracting relational facts by an end-to-end neural model with copy mechanism [C]//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL, 2018: 506-514.
- [32] FU T J, LI P H, MAWeiyun. GraphRel: modeling text as relational graphs for joint entity and relation extraction [C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: ACL, 2019: 1409-1418.
- [33] ZENG Xiangrong, HE Shizhu, ZENG Daojian, et al. Learning the extraction order of multiple relational facts in a sentence with reinforcement learning [C]//Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Stroudsburg, PA, USA: ACL, 2019: 367-377.
- [34] YU Bowen, ZHANG Zhenyu, SHU Xiaobo, et al. Joint extraction of entities and relations based on a novel decomposition strategy [C]//ECAI 2020. Amsterdam, Netherlands: IOS Press, 2020: 2282-2289.
- [35] SUI Dianbo, ZENG Xiangrong, CHEN Yubo, et al. Joint entity and relation extraction with set prediction networks [J]. IEEE Transactions on Neural Networks and Learning Systems, 2024, 35(9): 12784-12795.

(编辑 亢列梅)