

## 第 4 章 基于多角度对比学习的实体类型过度依赖消解

事件要素抽取任务是事件抽取的关键组成部分，成为了限制事件抽取整体性能的瓶颈。正如章节2.2所述，现有事件要素抽取的建模存在两种不同的定义，其不同定义的一个显著区别为是否将实体提及识别与事件抽取进行分隔，本章节研究在已经给定实体提及的情况下进行的事件要素抽取。

基于给定的实体提及，大多数事件要素模型在建模过程中利用了其正面信息增益，而忽视了实体类型和要素类型间的强关联对性能产生的负面影响。为了更好地研究该负面影响，本章节首先定义实体类型过度依赖并通过实验验证其降低了多种事件要素抽取模型的整体性能。进一步，本章提出了一种基于多角度对比学习的实体类型过度依赖消解方法解决该问题。实验评估表明，提出的方法可以通用有效地消解不同事件要素抽取模型对于实体类型信息的过度依赖。

### 4.1 引言

作为基于实体的抽取任务，事件要素抽取<sup>[41,99,104]</sup>主要研究在给定上下文语义信息的情况下如何更好地建模实体提及和事件触发词间的复杂交互。进一步，结合现有的事件要素抽取方法和以 ACE2005 为代表的事件要素抽取数据集，可以观察到以下两个数据特征：（1）实体类型为事件要素抽取的实体提及提供了主要信息来源。具体地，按照 Peng 等人<sup>[144]</sup>在关系抽取任务中的类似设置，将 ACE2005 数据集中的实体提及替换成对应的实体类型，并以此训练一个事件要素抽取任务的代表性模型 DMBERT<sup>[70]</sup>。实验结果表明 DMBERT 取得了和替换前几乎相同的性能结果。（2）大多数实体类型在事件中仅以少数特定要素类型参与<sup>[36]</sup>。例如，ACE2005 数据集中出现了 43 种实体类型，但有超过 74% 的实体类型以少于 5 种的要素类型参与到事件中。

基于上述两个数据特征，可以得知实体类型信息对于事件要素抽取任务是至关重要的，且与要素类型密切相关。现有事件要素抽取方法基于实体类型信息的重要性，更多地关注其正面信息增益，而忽略了实体类型和要素类型间的相关性带来的潜在负面影响。如表4.1所示，给定目标抽取文本“A rocket holding the first of two Mars rovers blasted off Tuesday on a seven-month voyage to the planet.”和 *Transport*（运输）事件的事件触发词“voyage”，一个性能优良的事件要素抽取模型需要识别出实体类型为

*Landing* 的实体提及 “two Mars rovers” 没有以任何要素类型参与到该事件中，即对应的事件要素类型为空。然而，由于其他实体类型为 *Landing* 的实体提及在训练数据中较高频率地以要素类型 *Vehicle* 出现在事件中，正如表4.1中所示的句1和句2，使得现有事件要素抽取模型容易受到实体类型 *Landing* 的误导，将实体提及 “two Mars rovers” 的要素类型错误地预测为 *Vehicle*。

表 4.1 实体类型过度依赖示例

实例	实体类型	要素类型标签	要素类型预测
例: A rocket holding the first of <b>two Mars rovers</b> blasted off Tuesday on a seven-month <b>voyage</b> to the planet.	Landing	None	Vehicle ×
句 1: <b>The rovers' landing</b> sites, on opposite sides of the planet, were chosen for their likelihood of holding evidence of water.	Landing	Vehicle	Vehicle ✓
句 2: <b>The bus</b> was ripped to shreds while <b>traveling</b> between a residential area and Haifa university.	Landing	Vehicle	Vehicle ✓

为此，本章首先定义了实体类型过度依赖并通过预实验评估证明了不同事件要素抽取基线模型均受到实体类型过度依赖问题的影响，使得整体性能出现下降。由于实体类型过度依赖是由建模时对实体类型信息的严重依赖导致的，消解该问题可以考虑提升模型学习除实体类型外的更多语义信息的能力。

广泛应用于计算机视觉<sup>[145,146]</sup>和自然语言处理<sup>[147,148]</sup>的对比学习技术是解决这一问题的有效方法，尤其是监督对比学习技术<sup>[149–151]</sup>，其可将属于同一类型实例（称之为正样本）的特征表示距离拉近，属于不同类型实例（称之为负样本）的特征表示距离推远。然而，在事件要素抽取任务中，直接应用传统的监督对比学习技术只能根据要素类型信息进行特征学习，而无法同时考虑到实体类型信息。

为了解决上述提到的挑战，本章提出基于多角度对比学习的实体类型过度依赖消解方法。在不同的事件要素抽取模型的基础上，提出了通用的多角度实体类型过度依赖消解（Multi-view Entity Type Overdependency Reduction, METOR）模型，其中多角度指正样本角度和负样本角度。基于正样本角度，本章提出了一种加权-选择的对比学习方法，以提升具有相同要素类型但具有不同实体类型的实例特征表示间的相似性。基于负样本角度，本章提出了一种伪正对比学习方法来降低具有不同要素类型但具有相同实体类型的实例特征表示间的相似性。此外，为了提升上述两种对比学习方

法间的协作能力，在模型训练过程中提出了一种高效的循环训练策略。本章的主要贡献概括如下：

- 定义了实体类型过度依赖问题并通过预实验验证了不同事件要素抽取基线模型均受到该问题的影响而导致整体性能下降。
- 针对实体类型过度依赖问题，本章提出了基于多角度对比学习的实体类型过度依赖消解方法。该方法可以应用于不同基线模型中，由两种对比学习方法和一种循环训练策略组成，其中循环训练策略能够提升不同对比学习方法间的训练协作。
- 在广泛使用的 ACE2005 数据集上进行了大量实验。实验结果表明，本章提出的方法能够通用有效地消解不同基线模型的实体类型过度依赖问题，并基于预训练架构取得当前最优性能。

## 4.2 实体类型过度依赖定义与验证

如上一章节所述，实体类型为事件要素抽取任务提供了关键信息且和要素类型信息强关联。为了更好地研究实体类型和要素类型之间的关联性，本节首先形式化地定义实体类型过度依赖，并通过预实验评估其对于事件要素抽取性能的影响。

### 4.2.1 定义

一般来说，事件要素抽取模型由两个主要部分组成：编码器和分类器。其中编码器将实例编码成特征表示，而分类器识别实例的特征表示和不同要素类型的表示间的相似性。由于实体类型构成了实体提及的主要信息，则以实体提及为重要组成部分的特征编码表示依赖于实体类型。假设  $G(r_k)$  是由要素类型为  $r_k$  的实例构成的群组， $C(e_i, r_k)$  是由实体类型为  $e_i$  和要素类型为  $r_k$  的实例构成的簇，而  $\mathbf{h}_{r_k}^M$  和  $\mathbf{h}_{e_i, r_k}^M$  分别是  $G(r_k)$  和  $C(e_i, r_k)$  由给定事件要素抽取模型  $M$  的编码器编码得到的表示，则簇级别的实体类型依赖定义如下：

**定义 1** (实体类型依赖). 给定  $C(e_i, r_k)$  和由  $M$  编码得到的对应编码表示  $\mathbf{h}_{e_i, r_k}^M$ ，如果存在  $C(e_i, r_l)$  和  $C(e_j, r_k)$  ( $i \neq j, l \neq k$ ) 满足如下条件，则称  $C(e_i, r_k)$  依赖于实体类型  $e_i$ ：

$$\mathbf{h}_{e_i, r_k}^M \cdot \mathbf{h}_{e_i, r_l}^M > \mathbf{h}_{e_i, r_k}^M \cdot \mathbf{h}_{e_j, r_k}^M \quad (4.1)$$

在定义1中, 如果  $C(e_i, r_k)$  和  $C(e_i, r_l)$  之间的相似性比  $C(e_i, r_k)$  和  $C(e_j, r_k)$  之间的相似性高, 则认定  $C(e_i, r_k)$  满足实体类型依赖定义。该定义阐述了由于实体类型  $e_i$  和要素类型  $r_k$  的关联性使得  $C(e_i, r_k)$  与群组  $G_{r_l}$  中的  $C(e_i, r_l)$  的特征表示间的相似性反而比其与同在一个群组中的  $C(e_j, r_k)$  的特征表示间的相似性更高的现象。

为了进一步评估这种实体类型依赖是否影响事件要素抽取模型的性能, 需要定义簇之间的语义不一致性。Ohashi 等人<sup>[152]</sup> 研究发现如果实例特征表示的相似性与其类别标签不一致, 则相应的分类器更加容易出现预测错误。因此, 簇级别的语义不一致性可以定义如下:

**定义 2 (语义不一致).** 给定  $C(e_i, r_k)$  和由  $M$  编码得到的对应编码表示  $\mathbf{h}_{e_i, r_k}^M$ , 如果存在  $G(r_l)$  ( $l \neq k$ ) 满足如下条件, 则称  $C(e_i, r_k)$  存在语义不一致:

$$\mathbf{h}_{e_i, r_k}^M \cdot \mathbf{h}_{r_l}^M > \mathbf{h}_{e_i, r_k}^M \cdot \mathbf{h}_{r_k}^M \quad (4.2)$$

在定义2中, 虽然  $C(e_i, r_k)$  中实例的要素类型是  $r_k$  而不是  $r_l$ , 但  $C(e_i, r_k)$  与群组  $G(r_k)$  的特征表示间的相似性比其与群组  $G(r_l)$  的特征表示间的相似性低。然而, 理想情况下  $C(e_i, r_k)$  应该与自身所属的群组相似性更高, 否则将出现语义不一致现象, 从而使得编码器  $M$  对应的分类器更容易预测出错。因此, 语义不一致性可用于确定存在实体类型依赖的簇是否影响事件要素抽取模型的性能。基于此, 定义簇级别的实体类型过度依赖如下:

**定义 3 (实体类型过度依赖).** 给定  $C(e_i, r_k)$  和由  $M$  编码得到的对应编码表示  $\mathbf{h}_{e_i, r_k}^M$ , 如果依次满足如下条件, 则称  $C(e_i, r_k)$  存在实体类型过度依赖:

- (a) 存在  $C(e_i, r_l)$  和  $C(e_j, r_k)$  ( $i \neq j, l \neq k$ ) 使得  $C(e_i, r_k)$  满足定义1。
- (b) 存在  $G(r_l)$  ( $l \neq k$ ) 使得  $C(e_i, r_k)$  满足定义2。

在定义3中, 如果簇  $C(e_i, r_k)$  满足实体类型依赖的定义且该依赖导致语义不一致, 进而导致分类器容易预测错误<sup>[152]</sup>, 则认为该簇过度依赖实体类型。因此, 实体类型过度依赖本质上为实体类型依赖导致的语义不一致。以图4.1为例, 若存在  $C(e_1, r_2)$  和  $C(e_5, r_1)$  使得  $C(e_1, r_1)$  满足定义1, 且存在  $G(r_2)$  使得  $C(e_1, r_1)$  满足定义2, 则可以判定  $C(e_1, r_1)$  对实体类型  $e_1$  过度依赖。

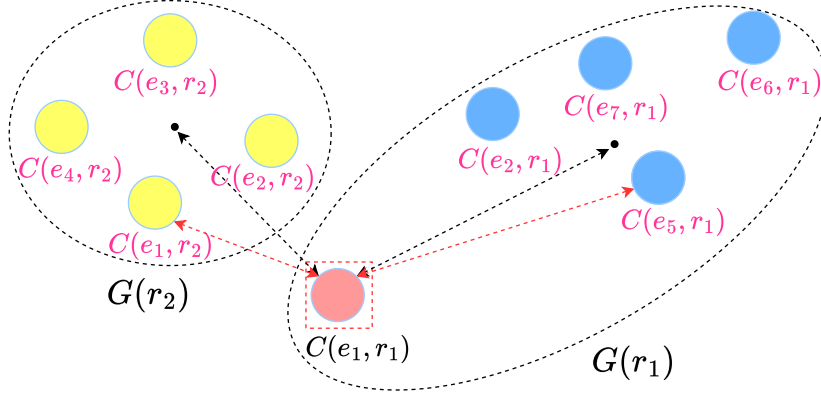


图 4.1 实体类型过度依赖示例图

#### 4.2.2 预实验验证

基于实体类型过度依赖的定义，本章节通过预实验验证其对于事件要素抽取性能的影响。

**数据集和模型** 在数据集方面，预实验评估选择了广泛使用的事件检测语料库 ACE2005 数据集，其具体细节将在主实验章节进行阐述。在模型方面，由于 RNN、CNN 和 BERT 是事件要素抽取最常用的三类编码器，预实验基于所选数据集训练了三种不同的事件要素抽取模型，包括 AttRNN、DMCNN<sup>[41]</sup> 和 DMBERT<sup>[70]</sup>。其中，基于 RNN 的事件要素抽取模型如 JRNN<sup>[100]</sup> 和 dbRNN<sup>[101]</sup> 未公布源代码，因此本文受 Lu 等人<sup>[54]</sup> 研究工作的启发，利用增加了注意力机制的双向 LSTM 作为使用 RNN 架构的基线模型，记作 AttRNN。此外，由于训练集和测试集之间的数据分布相似，实验中将根据实体类型依赖、语义不一致和实体类型过度依赖的定义，分别评估测试集中相应簇的性能。由于事件要素抽取的测试集依赖于事件检测的预测结果，本文遵循 Wang 等人和 Xi 等人<sup>[99,104]</sup> 的实验设置，分别使用训练好的 AttRNN、DMCNN 和 DMBERT 来检测事件类型和相应的触发词。考虑到测试的实例数量取决于不同事件检测模型正确预测的事件数量，本实验中三种基线模型对应的测试集实例数分别为 4013、3630 和 4081。

**实验步骤** 首先，对于给定模型，本章使用其编码器获取测试集中每个实例的特征编码表示。然后，基于要素类型，测试集中的所有实例被划分为不同的群组，使得每个群组中实例的要素类型是相同的。进一步地，根据实体类型，将每个群组中的实例划分为不同的簇。因此，同一簇中实例的实体类型和要素类型是相同的。对于 AttRNN、DMCNN 和 DMBERT 各自对应的编码器，根据上述划分步骤可分别得到簇



的数目为 156、148 和 161。基于此，本章通过简单地计算各个群组/簇中所有实例特征表示的平均值，以用作其整体表示，其避免了引入额外计算参数和降低了计算复杂度。然后，本实验根据实体类型依赖、语义不一致和实体类型过度依赖的定义对每个簇进行判定，并分别计算符合不同定义的测试子集的性能（F1 值）。

**评估验证** 预实验结果如表4.2所示，其展示了根据不同定义类型划分的测试子集的簇数目及相应性能结果。从表中可以观察到：（1）对于不同的事件要素抽取模型，符合实体类型过度依赖的实例簇的性能都较低。其与其他类型的实例簇之间的巨大性能差异表明，现有事件要素抽取基线模型不擅长处理实体类型过度依赖问题。（2）符合实体类型过度依赖的实例簇和语义不一致的实例簇之间的性能差异表明，与语义不一致相比，由实体类型和要素类型之间的相关性导致的实体类型过度依赖进一步降低了模型的性能。（3）未对实体类型信息过度依赖的簇和整个测试集的性能差异表明，如果事件要素抽取模型缓解了实体类型过度依赖问题，则整体性能可以得到提升。

表 4.2 测试集中不同类型实例簇的数目及性能结果

模型	实体类型依赖 (数目/F1 (%))	语义不一致 (数目/F1 (%))	实体类型过度依赖 (数目/F1 (%))	非实体类型过度依赖 (数目/F1 (%))	所有簇 (数目/F1 (%))
AttRNN	100 / 40.6	56 / 38.9	48 / <b>18.1</b>	108 / 63.6	156 / 50.9
DMCNN	112 / 46.0	76 / 32.6	52 / <b>15.6</b>	96 / 71.4	148 / 53.5
DMBERT	95 / 38.6	48 / 40.7	31 / <b>15.3</b>	130 / 68.9	161 / 57.2

由于不同簇中的实例数量差异较大，本章节进一步统计了符合实体类型过度依赖的簇中实例的数目占据所有测试实例的比例，记作“实体类型过度依赖比例”。表4.3展示了不同事件要素抽取模型的比例结果。本文将在章节4.4中评估下一节所提的实体类型过度依赖消解方法是否可以降低“实体类型过度依赖比例”，进而提高整体性能。

表 4.3 符合实体类型过度依赖的簇中实例统计信息

模型	实体类型过度依赖的簇中实例数	总实例数	实体类型过度依赖比例 (%)
AttRNN	2845	4013	70.9
DMCNN	2799	3630	77.1
DMBERT	1281	4081	31.4

从预实验中可以观察到，实体类型过度依赖问题降低了事件要素抽取模型的整体性能。为了缓解该问题，本文考虑降低“实体类型过度依赖比例”。进一步，其是由实体类型依赖导致，则可以利用以下两种方式处理：（1）增加具有相同要素类型但实体类型不同的实例特征表示之间的相似性；（2）降低具有不同要素类型但实体类型相

同的实例特征表示之间的相似性。基于此，下一节将具体介绍将上述两种方式整合的所提方法，从而提高事件要素抽取的性能。

### 4.3 方法设计

本章将详细介绍基于多角度对比学习的实体类型过度依赖消解方法，其可以应用于不同的事件要素抽取基线模型。为了描述方便，本章选取 DMBERT<sup>[70]</sup>，以其为基础介绍相应的多角度实体类型过度依赖消解的事件要素抽取模型 METOR。图4.2展示了所提模型的总体架构，其主要由四个模块组成：编码器、正样本处理、负样本处理和循环训练策略。

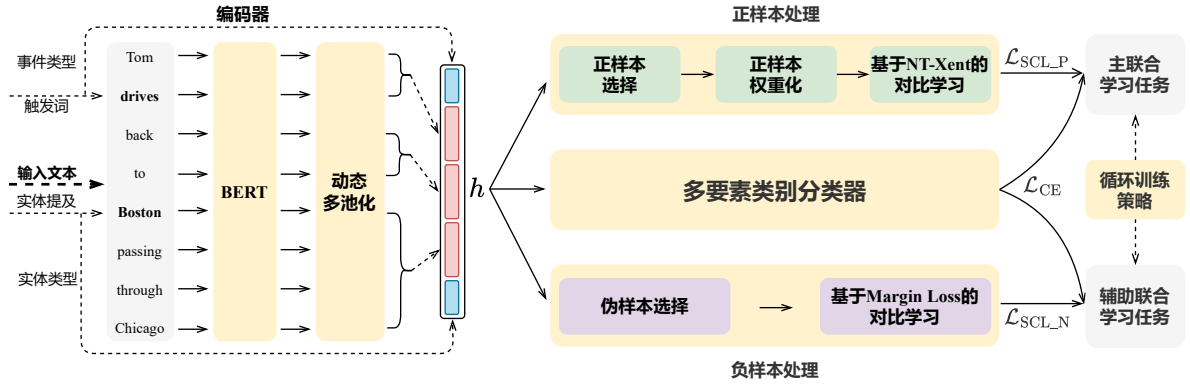


图 4.2 METOR 模型架构图

#### 4.3.1 编码器

编码器模块旨在将每个实例编码成对应的特征表示。由于事件要素抽取为事件检测的后续任务，本章节遵循 Wang 等人<sup>[99]</sup>和 Xi 等人<sup>[104]</sup>的研究工作，先利用 Wang 等人<sup>[99]</sup>提出的 DMBERT 事件检测模型进行事件触发词和相应事件类型的预测，并将所有实体提及设置为待抽取的事件要素候选。由于句子中可能存在多个事件要素候选，本章将这些候选分成多个实例处理。假设给定的事件要素候选为  $arg$ ，其实体类型为  $e$ ，所参与预测的事件触发词为  $tri$ ，对应的事件类型为  $t$ ，首先利用 BERT<sup>[103]</sup> 对所在句子文本  $x = \{w_1, \dots, tri, \dots, arg, \dots, w_n\}$  进行编码得到特征表示：

$$\{\hat{h}_1, \dots, \hat{h}_{p_{tri}}, \dots, \hat{h}_{p_{arg}}, \dots, \hat{h}_n\} = \text{BERT}(w_1, \dots, tri, \dots, arg, \dots, w_n) \quad (4.3)$$

其中  $p_{tri}$  和  $p_{arg}$  分别表示  $tri$  和  $arg$  在输入句子中的位置。

在此之后，使用一个动态多池化操作分段聚合编码后的特征表示：

$$[h_{1,p_{tri}}]_k = \max \left\{ [\hat{h}_1]_k \cdots, [\hat{h}_{p_{tri}}]_k \right\} \quad (4.4)$$

$$[h_{p_{tri}+1,p_{arg}}]_k = \max \left\{ [\hat{h}_{p_{tri}+1}]_k \cdots, [\hat{h}_{p_{arg}}]_k \right\} \quad (4.5)$$

$$[h_{p_{arg}+1,n}]_k = \max \left\{ [\hat{h}_{p_{arg}+1}]_k \cdots, [\hat{h}_n]_k \right\} \quad (4.6)$$

其中  $[\cdot]_k$  表示对应向量的第  $k$  个值。然后，本章随机初始化所有的事件类型标签和实体类型标签，分别表示为  $\mathbf{W}_T \in \mathbb{R}^{n_t \times d_s}$  和  $\mathbf{W}_E \in \mathbb{R}^{n_e \times d_s}$ 。其中， $d_s$ 、 $n_t$  和  $n_e$  分别表示事件类型（实体类型）向量的维度、预定义的事件类型数目和预定义的实体类型数目。此外， $\mathbf{W}_T(t)$  表示使用事件类型  $t$  从向量矩阵  $\mathbf{W}_T$  中查找出的事件类型向量。类似地， $\mathbf{W}_E(e)$  表示根据实体类型  $e$  查找出的实体类型向量。最后，串接上述表示，得到实例的特征表示  $\mathbf{h} \in \mathbb{R}^{d_{out}}$  如下：

$$\mathbf{h} = [h_{1,p_{tri}}; h_{p_{tri}+1,p_{arg}}; h_{p_{arg}+1,n}; \mathbf{W}_T(t); \mathbf{W}_E(e)] \quad (4.7)$$

### 4.3.2 正样本处理

本模块提出了一种选择-加权的对比学习方法，该方法包括三个步骤：正样本选择、正样本加权和基于归一化温度标度交叉熵（NT-Xent）的对比学习。首先，该方法基于给定实例的实体类型选择正样本，使得模型能够聚焦于处理与给定实例的实体类型无关的正样本。然后，计算全局共现信息和语义相关性信息，以对所选正样本的相关性进一步权重化处理。基于此，模型可以提升特定正样本的对比学习贡献，其包含了更多与给定实例要素类型相关的关键语义信息，从而学习到实体类型信息之外的更多有效语义知识。最后，计算基于 NT-Xent 的选择-加权对比学习方法的训练损失。

**正样本选择** 给定一批包含  $K$  个实例的批数据  $\mathcal{B}$ ，对于其中的每个实例  $x_i$ ，首先选择与其要素类型相同的实例作为正样本集合  $\mathcal{P}(x_i)$ ，选择与其要素类型不同的实例作为负样本集合  $\mathcal{N}(x_i)$ 。基于  $\mathcal{P}(x_i)$ ，本节采用一个简单的选择策略去获取新的正样本集合：

$$\mathcal{P}_s(x_i) = \{x_p : x_p \in \mathcal{B}, (e_p \neq e_i) \wedge (r_p = r_i) \wedge (p \neq i)\} \quad (4.8)$$

其中  $r_i$  和  $e_i$  分别表示实例  $x_i$  的要素类型和实体类型。本选择策略只用于包含了不止



一个样本的  $\mathcal{P}(x_i)$ ，以避免造成对比数据的稀疏。

**正样本加权** 在完成正样本选择后，本章根据要素类型  $r_i$ ，对  $\mathcal{P}_s(x_i)$  中的每个正样本分配一个权重。传统的对比学习方法在对比学习损失中同等地对待所有正样本，无法区分具有不同实体类型的正样本的重要程度。为此，本章基于要素类型  $r_i$ ，从全局共现信息和语义相关性信息两个方面，计算不同正样本对于  $x_i$  的重要性。

全局共现信息可表示为记录了整个训练集中不同实体类型和要素类型共现数量的共现矩阵  $I_{e,r}$ 。假设  $\mathcal{E}$  和  $\mathcal{R}$  分别为训练集中的实体类型和要素类型集合，批数据  $\mathcal{B}$  中第  $p$  个实例（即  $x_p$ ）存在于集合  $\mathcal{P}_s(x_i)$  中，其实体类型为  $e_p$ 。受 TF-IDF<sup>[153]</sup> 启发，首先定义实体类型频率（ETF），用于表示  $e_p$  与  $r_i$  同时出现的频率：

$$\text{ETF}(e_p, r_i) = \frac{I_{e_p, r_i}}{\sum_{m \in \mathcal{E}} I_{m, r_i}} \quad (4.9)$$

此外，本章使用与  $e_p$  同时出现的要素类型的数量来衡量  $e_p$  的重要性。要素类型数量越少， $e_p$  则越重要。因此，反向要素类型频率（IRTF）可根据如下定义得到：

$$\text{IRTF}(e_p) = \log \frac{|\mathcal{R}|}{|\{r \in \mathcal{R} : I_{e_p, r} > 0\}| + 1} \quad (4.10)$$

基于 ETF 和 IRTF，可得实例  $x_p$  对于实例  $x_i$  在全局共现信息方面的重要性得分：

$$w_1(x_p, x_i) = \text{ETF}(e_p, r_i) \times \text{IRTF}(e_p) \quad (4.11)$$

然而，在事件类型不同的实例中，具有相同实体类型的实体提及可能以不同的要素类型参与在事件中。因此，对于不同的事件类型，即使实例的实体类型相同，对给定实例的重要性也应有所区分。因此，考虑  $x_p$  对  $x_i$  的重要性时仅使用全局共现信息不足以建模不同事件类型的影响。基于此，本节进一步利用语义相关性信息。首先，将所有要素类型标签随机初始化成向量矩阵  $\mathbf{W}_R \in \mathbb{R}^{n_r \times d_s}$ ，其中  $n_r$  为预定义的要素类型的数目（包括特殊要素类型“空”）， $d_s$  是要素类型向量的维度。然后，对于给定的实例  $x_i$ ，得到融合事件类型信息的实体类型和要素类型表示：

$$\mathbf{e}_i^t = \mathbf{W}_T(t_i) \odot \mathbf{W}_E(e_i) \quad (4.12)$$

$$\mathbf{r}_i^t = \mathbf{W}_T(t_i) \odot \mathbf{W}_R(r_i) \quad (4.13)$$

其中  $\odot$  表示元素依次相乘,  $r_i$ 、 $e_i$  和  $t_i$  分别表示实例  $x_i$  的要素类型、实体类型和事件类型。 $\mathbf{W}_T(t_i)$  表示使用事件类型  $t_i$  从向量矩阵  $\mathbf{W}_T$  中查找出的事件类型向量。类似地, 可以分别得到实体类型向量  $\mathbf{W}_E(e_i)$  和要素类型向量  $\mathbf{W}_R(r_i)$  的含义。由于  $\mathbf{W}_T$  和  $\mathbf{W}_E$  同时也是公式 4.7 中串接向量的一部分, 事件类型和实体类型的语义信息实现了在不同模块间的共享。同样, 可以得到任意正样本  $x_p$  的融合事件类型信息的实体类型表示  $\mathbf{e}_p^t$ 。然后, 本章基于语义相关性信息, 利用注意力机制计算  $x_p$  对于  $x_i$  的重要性得分, 并根据 softmax 函数得到归一化结果  $w_2(x_p, x_i)$ :

$$s(x_p, x_i) = \mathbf{v}^\top \tanh(\mathbf{U} [\mathbf{e}_i^t; \mathbf{e}_p^t; \mathbf{r}_i^t]) \quad (4.14)$$

$$w_2(x_p, x_i) = \frac{\exp(s(x_p, x_i))}{\sum_{e_j \in \mathcal{E}} \exp(s(x_j, x_i))} \quad (4.15)$$

其中  $\mathbf{v} \in \mathbb{R}^{3d_s}$  和  $\mathbf{U} \in \mathbb{R}^{3d_s \times 3d_s}$  分别表示可训练学习的向量和矩阵参数。然后, 可得  $x_p$  对于  $x_i$  的最终重要性得分:

$$\mathbf{w}(x_p, x_i) = \alpha w_1(x_p, x_i) + (1 - \alpha) w_2(x_p, x_i) \quad (4.16)$$

其中  $\alpha$  ( $0 < \alpha < 1$ ) 为权重参数。

基于 NT-Xent 的对比学习 遵循 Chen 等人<sup>[145]</sup> 和 Khosla 等人<sup>[149]</sup> 的研究, 本章首先将实例  $x_i$  的特征表示  $\mathbf{h}_i$  映射到一个新空间上, 以提高对比学习的质量:

$$\mathbf{z}_i = \mathbf{W}_1 f(\mathbf{W}_2 \mathbf{h}_i) \quad (4.17)$$

$\mathbf{W}_1 \in \mathbb{R}^{d_1 \times d_2}$  和  $\mathbf{W}_2 \in \mathbb{R}^{d_2 \times d_{out}}$  为可训练学习的矩阵参数,  $f(\cdot)$  为 ReLU 激活函数, 其中  $d_1$  和  $d_2$  为维度参数。在选择出正样本并对其权重化后, 本章基于 NT-Xent<sup>[145]</sup> 计算相应的对比学习损失  $\mathcal{L}_{SCL\_P}$  如下:

$$\mathcal{L}_{SCL\_P} = \sum_{i=1}^K \frac{-1}{|\mathcal{P}_s(x_i)|} \sum_{p \in \mathcal{P}_s(x_i)} \mathbf{w}(x_p, x_i) \cdot \log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_p) / \tau)}{\sum_{n \in K \setminus i} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_n) / \tau)} \quad (4.18)$$

其中  $\text{sim}(\mathbf{z}_i, \mathbf{z}_p) = \mathbf{z}_i^\top \mathbf{z}_p / \|\mathbf{z}_i\| \|\mathbf{z}_p\|$ ,  $\tau$  表示温度参数。

### 4.3.3 负样本处理

本模块提出了一种伪正对比学习方法，该方法包括两个步骤：伪样本选择和基于边际损失（Margin Loss）的对比学习。首先，基于实体类型从负样本中选择出伪正样本。然后，利用这些选择出的样本作为对比参考，降低给定实例和与其实体类型相同的负样本之间的特征表示相似性。

**伪样本选择** 给定一批包含  $K$  个实例的批数据  $\mathcal{B}$ ，对于其中的每个实例  $x_i$ ，首先选择对应的伪正样本和伪负样本，选择过程如下：

$$\mathcal{P}_{pseudo}(x_i) = \{x_p : x_p \in \mathcal{B}, (e_p \neq e_i) \wedge (r_p \neq r_i) \wedge (p \neq i)\} \quad (4.19)$$

$$\mathcal{N}_{pseudo}(x_i) = \{x_n : x_n \in \mathcal{B}, (e_n = e_i) \wedge (r_n \neq r_i) \wedge (n \neq i)\} \quad (4.20)$$

其中伪正样本集合和伪负样本集合分别表示为  $\mathcal{P}_{pseudo}(x_i)$  和  $\mathcal{N}_{pseudo}(x_i)$ 。

**基于边际损失的对比学习** 首先，本章计算  $\mathcal{P}_{pseudo}(x_i)$  和  $\mathcal{N}_{pseudo}(x_i)$  中实例特征表示的平均值，分别得到  $\mathbf{z}_i^P$  和  $\mathbf{z}_i^N$ ，以表示这两个集合的整体信息。然后，利用  $\mathcal{P}_{pseudo}(x_i)$  作为对比参考进行对比学习，以降低实例  $x_i$  与属于  $\mathcal{N}_{pseudo}(x_i)$  的样本之间的相似性。由于  $\mathcal{P}_{pseudo}(x_i)$  中的样本本质上仍是  $x_i$  的负样本，使用实例级的对比学习损失 NT-Xent 以增加伪正样本与给定实例之间的相似性是不合适的。因此，本章使用边际损失来对伪正样本集合和伪负样本集合的整体表示进行对比学习：

$$\mathcal{L}_{SCL\_N} = \sum_{i=1}^K \max \{ \mathbf{z}_i^N \cdot \mathbf{z}_i - \mathbf{z}_i^P \cdot \mathbf{z}_i + \gamma, 0 \} \quad (4.21)$$

其中  $\gamma$  为边际超参数。

### 4.3.4 循环训练策略

本模块首先计算事件要素抽取任务的损失。给定实例  $x$ ，将其从编码器模块得到的特征表示  $\mathbf{h}$  输入到多要素类别分类器，以计算概率分布  $p(x)$  如下：

$$p(x) = \text{softmax}(\mathbf{W}^c \mathbf{h} + \mathbf{b}^c) \quad (4.22)$$

其中  $\mathbf{W}^c \in \mathbb{R}^{n_r \times d_{out}}$  和  $\mathbf{b}^c \in \mathbb{R}^{n_r}$  分别表示分类器中要训练的参数。给定包含  $K$  个实例的批数据  $\mathcal{B}$ ，交叉熵损失  $\mathcal{L}_{CE}$  计算如下：

$$\mathcal{L}_{CE} = - \sum^K \log p(r|x) \quad (4.23)$$

其中  $p(r|x)$  表示将实例  $x$  预测为正确的要素类型  $r$  的概率值。在测试阶段， $p(x)$  用于获取事件要素抽取的预测要素类型。

进一步，本章注意到优化  $\mathcal{L}_{SCL\_N}$  将不可避免地提升实例  $x_i$  和  $\mathcal{P}_{pseudo}(x_i)$  中的样本（本质上为负样本）间的特征表示相似度。然而，根据公式4.18，优化  $\mathcal{L}_{SCL\_P}$  会降低实例  $x_i$  和所有负样本间的特征表示相似度。因此， $\mathcal{L}_{SCL\_N}$  和  $\mathcal{L}_{SCL\_P}$  的训练目标是不一致的。为此，本章提出了一个循环训练策略，该策略依次使用两个不同的联合训练目标来协调  $\mathcal{L}_{SCL\_P}$  和  $\mathcal{L}_{SCL\_N}$  的不一致。具体来说，将联合优化  $\mathcal{L}_{SCL\_P}$  和  $\mathcal{L}_{CE}$  视为主联合学习任务，先训练  $f$  轮，其中  $f$  是超参数。然后，将联合优化  $\mathcal{L}_{SCL\_N}$  和  $\mathcal{L}_{CE}$  视作辅助联合学习任务，再训练一轮。重复上述过程直到训练完成。通过这种方式，可以逐步调整处理  $\mathcal{L}_{SCL\_P}$  和  $\mathcal{L}_{SCL\_N}$  训练目标的不一致。本章提出的训练策略表示如下：

$$\mathcal{L}_{JOINT} = \begin{cases} \mathcal{L}_{SCL\_P} + \mathcal{L}_{CE} & epoch \bmod f \neq 0 \\ \mathcal{L}_{SCL\_N} + \mathcal{L}_{CE} & epoch \bmod f = 0 \end{cases} \quad (4.24)$$

其中  $epoch$  表示完成训练的轮次。

#### 4.4 实验评估

在实验评估中，本章所提的方法不仅应用于基线模型 DMBERT，也应用于 AttRNN 和 DMCNN 基线模型，分别记作 METOR (RNN) 和 METOR (CNN)。在 METOR (RNN) 和 METOR (CNN) 中，将编码器模块中的 DMBERT 分别替换为 AttRNN 和 DMCNN，其他模块保持不变。由于事件要素抽取任务依赖于事件检测的结果，本章遵循 Wang 等人<sup>[99]</sup> 和 Xi 等人<sup>[104]</sup> 的研究设定，使用经过训练的 AttRNN、DMCNN 和 DMBERT 分别检测事件触发词和相应的事件类型，作为 METOR (RNN)、METOR (CNN) 和 METOR 模型的输入。接下来，本章将介绍实验设置、性能结果、消融研究、实体类型信息深入分析、超参数敏感性分析、计算复杂性分析和案例研究。

#### 4.4.1 实验设置

**1. 数据集** 遵循最近主流的给定实体提及的事件要素抽取研究工作<sup>[57,100,104]</sup>，本章实验评估在 ACE2005 数据集上进行。与章节3.3.1相同，其为 ACE2005 语料库<sup>[138]</sup> 中的英文版本数据集。根据 Wang 等人<sup>[99]</sup> 的预处理设置<sup>1</sup>，该数据集包含 599 个文档、33 种事件类型、46 种实体类型和 35 种事件要素类型。此外，引入 *None*（空）作为特殊要素类型，以表示相应的候选要素在给定实例中为非事件要素。进一步，该数据集被划分为 529、30 和 40 个文档，分别作为训练集、验证集和测试集。

**2. 评价指标** 本章实验遵循事件要素抽取任务的标准评估。如果事件类型、偏移量和要素类型与数据集的标注完全相同，则认定候选要素被正确抽取，其中偏移量指候选要素文段在句子中的开始和结束位置索引。基于此，本文使用准确率（P）、召回率（R）和 F1 值（F1）作为评估指标，其计算方式参照章节3.3.1。

**3. 实验配置** 在编码器模块中，对于 METOR（RNN）本章使用预训练 Glove 词向量、随机初始化事件类型向量、随机初始化实体类型向量和随机初始化位置向量，维度分别为 100、5、5 和 5，并在模型中使用双向 LSTM 和自注意力机制。对于 METOR（CNN），使用与 DMCNN 相同的输入向量和超参数。类似地，本章将 5 维随机初始化的实体类型向量添加到 METOR（CNN）的输入中。对于 METOR，编码使用的 BERT 预训练模型为 HuggingFace 发布的“google-bert/bert-base-uncased”版本<sup>2</sup>，实体类型向量维度  $d_s$  设置为 50。在正样本处理模块中，本章将  $d_1$  设置为 512， $d_2$  设置为 512，温度参数  $\tau$  设置为 0.1，权重  $\alpha$  设置为 0.7。在负样本处理模块中，将  $\gamma$  设置为 0.1。在循环训练策略中，METOR（RNN）、METOR（CNN）和 METOR 的循环频率  $f$  分别设置为 4、3 和 3。此外，其批数据大小均设置为 80，训练轮次设置为 10，优化器使用 AdamW，对应的学习率分别设置为  $1 \times 10^{-3}$ 、 $1 \times 10^{-3}$  和  $5 \times 10^{-5}$ 。对于不同的模型，训练均在相同实验服务器上进行，具体配置为：CPU 型号 Intel(R) Xeon(R) Gold 5320，主频 2.20GHz；内存 256G；GPU 计算卡型号 NVIDIA GeForce RTX 3090，显存 24G。上述模型均基于 Pytorch<sup>[139]</sup> 框架进行构建。

**4. 基线模型** 本节选择以下模型作为实验比较的基线：

- DMCNN<sup>[41]</sup> - 该模型使用了 CNN 和动态多池化操作提取实例的特征表示。

<sup>1</sup><https://github.com/thunlp/HMEAE>

<sup>2</sup><https://huggingface.co/google-bert/bert-base-uncased>

- JRNN<sup>[100]</sup> - 该模型提出了包含离散二元矩阵的 RNN，其利用了事件类型和要素类型、要素类型间的相互依赖性。
- dbRNN<sup>[101]</sup> - 该模型在 RNN 中引入了依存句法信息，并构建事件内候选要素间的交互向量以捕捉其关联性。
- DMBERT<sup>[70]</sup> - 该模型使用了 BERT 和动态多池化操作来提取实例的特征表示。
- PLMETOR<sup>[69]</sup> - 该模型基于 BERT 来预测事件要素类型并生成额外的训练实例，以进一步提高性能。
- HMEAE<sup>[99]</sup> - 该模型提出了一种利用事件要素类型语义相关性的层次模块化注意力网络。
- BERT (Inter)<sup>[104]</sup> - 该模型基于 BERT 并利用了跨事件的相互依赖关系<sup>[100]</sup>。
- BERT (Intra)<sup>[104]</sup> - 该模型基于 BERT 并利用了事件内部的要素依赖关联<sup>[101]</sup>。
- BERD<sup>[104]</sup> - 该模型提出了一种能够利用事件内其他要素的类型预测语义信息的编码器-解码器框架。

本章注意到 RCEE\_ER<sup>[71]</sup> 在 F1 指标上取了优秀的性能表现，但其受益于额外的数据增广和无监督数据。因此，本章遵循 Xi 等人<sup>[104]</sup> 在 BERD 模型中的基线模型设置，未将 RCEE\_ER 模型作为实验性能比较的基线。此外，本章将 ChatGPT 在 ACE2005 数据集上的性能评估保留到章节 5.3 进行展示与分析。

#### 4.4.2 性能结果

表 4.4 展示了本章所提方法与基线模型在 ACE2005 数据集上的性能结果，其中 † 表示将本章提出的方法应用在 DMBERT 基线模型时，其性能在双侧  $t$  检验下显著 ( $p < 0.01$ ) 优于之前性能最优的事件要素抽取模型 BERD。根据结果可进一步观察到，与 AttRNN 相比，METOR (RNN) 在 F1 指标上实现了 3.9% 的性能提升。即使与利用了事件类型和要素类型、要素类型间的相互依赖性的 JRNN 相比，METOR (RNN) 也取得了相差无几的 F1 指标结果。此外，dbRNN 的 F1 指标显著优于 METOR (RNN)，本文将其归因于两个原因：(1) dbRNN 利用了额外的句法信息。(2) dbRNN 同时优化了事件检测和事件要素抽取任务的性能，而 METOR (RNN) 没有受益于这种联合



优化。当本章提出的方法应用在 DMCNN 时, METOR (CNN) 相比该基线在 F1 指标上实现了 5% 的提升, 比同样使用了 DMCNN 作为编码器的 HMEAE (CNN) 提升了 2.8%。并且, METOR (CNN) 的 F1 指标优于大多数没有使用预训练模型的事件要素抽取模型, 并与 dbRNN 相比时展现了相当的性能结果。同时, 可以观察到 METOR (CNN) 的性能优于使用了 BERT 预训练模型作为编码器的 DMBERT、PLMETOR 和 BERT (Inter)。

当在编码器模块使用 DMBERT 时, METOR 取得了当前最优的 F1 性能, 且相比于之前性能最优的 BERD 模型提升了 2.1%。此外, METOR 的 F1 指标比相应的 DMBERT 基线高出了 5.2%。总的来说, 由于本章所提的 METOR 模型可以学习到除实体类型之外的更多语义信息, 从而消解了实体类型过度依赖问题, 实现了在事件要素抽取任务上对所有基线模型的性能超越。

表 4.4 ACE2005 数据集上的事件要素抽取性能结果

模型	P(%)	R(%)	F1(%)
AttRNN	50.6	51.1	50.9
DMCNN	62.2	46.9	53.5
JRNN	54.2	56.7	55.4
dbRNN	<b>66.2</b>	52.8	58.7
HMEAE (CNN)	57.3	54.2	55.7
<b>METOR (RNN)</b>	51.9	58.0	54.8
<b>METOR (CNN)</b>	56.2	61.1	58.5
+ BERT ( <i>base</i> )			
DMBERT	58.8	55.8	57.2
PLMETOR	62.3	54.2	58.0
BERT (Inter)	58.4	57.1	57.8
BERT (Intra)	56.4	61.2	58.7
HMEAE (BERT)	62.2	56.6	59.3
BERD	59.1	61.5	60.3
<b>METOR</b>	60.3	<b>64.7</b>	<b>62.4<sup>†</sup></b>

#### 4.4.3 消融研究

本章通过消融研究分析所提方法中每个模块的有效性, 包括正样本处理 (Positive Sample Processing)、负样本处理 (Negative Sample Processing) 和循环训练策略 (Cyclic Training Strategy), 其中  $\Delta F1$  表示与完整方法在 F1 指标上的性能差距。如表 4.5 所

示，将本章所提的方法应用到不同的基线模型上，三个模块都能提升相应基线模型的性能。本节选择 METOR 模型进行着重讨论。首先，如果移除正样本处理和负样本处理模块，METOR 模型在 F1 指标上的性能将分别下降 2.0% 和 1.3%，这表明从正样本和负样本角度消解实体类型过度依赖都可以有效地提升事件要素抽取性能。其次，如果将不同模块的损失直接求和并进行联合训练，以替代循环训练策略，则 METOR 模型在 F1 上的性能将下降 2.9%。这表明循环训练策略可以有效地调和正样本处理模块和负样本处理模块中的不同对比学习目标。此外，可以观察到即使移除掉正样本处理模块或负样本处理模块，METOR 模型仍优于之前性能最优的 BERD 模型。本文将其归因于这两个模块都推动了模型学习除实体类型之外的更多语义信息，从而消解了实体类型过度依赖问题，提升了整体性能。

表 4.5 ACE2005 数据集上的消融研究结果

模型	P(%)	R(%)	F1(%)	$\Delta$ F1
<b>METOR (RNN)</b>	51.9	58.0	54.8	-
w/o Positive Sample Processing	48.7	55.7	52.0	-2.8
w/o Negative Sample Processing	51.1	57.2	54.0	-0.8
w/o Cyclic Training Strategy	50.8	52.7	51.7	-3.1
<b>METOR (CNN)</b>	56.2	61.1	58.5	-
w/o Positive Sample Processing	52.9	59.2	55.9	-2.6
w/o Negative Sample Processing	57.4	56.3	56.8	-1.7
w/o Cyclic Training Strategy	54.0	57.4	55.7	-2.8
<b>METOR</b>	60.3	64.7	62.4	-
w/o Positive Sample Processing	59.7	61.2	60.4	-2.0
w/o Negative Sample Processing	59.3	62.9	61.1	-1.3
w/o Cyclic Training Strategy	57.1	62.1	59.5	-2.9

#### 4.4.4 实体类型信息分析

作为基于实体的任务，实体类型信息对于事件要素抽取任务的影响是多方面的。一方面，实体类型是事件要素抽取任务中实体提及的主要信息来源。另一方面，其导致事件要素抽取模型产生了实体类型过度依赖，进而降低了模型的整体性能。因此，本节将对实体类型信息进行深入全面的分析。

**不同变体模型的实体类型信息利用效率分析** 本节设计两种变体模型作为额外的比较基线，即“Baseline+type”和“Baseline+type+CL”，其中“Baseline”可以为 AttRNN、

DMCNN 或 DMBERT。具体地,“Baseline”模型不使用实体类型信息,而“Baseline+type”使用实体类型信息作为输入的一部分。此外,对于“Baseline+type+CL”,其除了使用实体类型信息作为输入外,还引入监督对比学习方法<sup>[149]</sup>损失,与公式4.23中的交叉熵损失直接求和后并进行联合训练。

表4.6展示了本章提出的方法与不同变体模型的性能比较。可以观察到,当“Baseline”为 AttRNN、DMCNN 和 DMBERT 时,“Baseline+type”在 F1 指标上的性能与其相比分别提升了 0.8%、1.2% 和 1.2%。这表明在事件要素抽取基线模型的输入中利用实体类型信息,其性能能够得到一定的提升。此外,当“Baseline”为 DMCNN 和 DMBERT 时,“Baseline+type+CL”可以进一步提高“Baseline+type”在 F1 指标上的性能,分别为 0.2% 和 0.6%,其表明传统的监督对比学习方法可以进一步提高基线模型利用实体类型信息的效率。进一步,由于 METOR 模型使用了实体类型信息并针对该信息设计了多角度对比学习方法,使得其能在利用实体类型信息的同时,从文本中学习除实体类型信息之外的更多语义信息。因此,与 AttRNN、DMCNN 和 DMBERT 相比, METOR 模型在 F1 上的性能分别提升了 3.9%、5.0% 和 5.2%。

表 4.6 ACE2005 数据集上不同变体模型的性能结果对比

模型	P(%)	R(%)	F1(%)	$\Delta$ F1
AttRNN	50.6	51.1	50.9	-
AttRNN+type	51.8	51.6	51.7	+0.8
AttRNN+type+CL	50.2	53.4	51.7	+0.8
<b>METOR (RNN)</b>	51.9	58.0	54.8	+3.9
DMCNN	62.2	46.9	53.5	-
DMCNN+type	51.5	58.3	54.7	+1.2
DMCNN+type+CL	53.8	56.1	54.9	+1.4
<b>METOR (CNN)</b>	56.2	61.1	58.5	+5.0
DMBERT	58.8	55.8	57.2	-
DMBERT+type	55.7	61.3	58.4	+1.2
DMBERT+type+CL	55.9	62.4	59.0	+1.8
<b>METOR</b>	60.3	64.7	62.4	+5.2

**实体类型过度依赖程度分析** 在章节4.2中,可以得知不同的事件要素抽取基线模型均存在实体类型过度依赖问题。如果反映实体类型过度依赖程度的“实体类型过度依赖比例”指标降低,则对应的整体性能也会提升。为了直观验证本章提出的方法在解决实体类型过度依赖问题上的有效性,本节将基于“实体类型过度依赖比例”指标进行具体分析。如图4.3所示,与相应的“Baseline”模型相比,“Baseline+type+CL”的

“实体类型过度依赖比例”均略有下降，表明普通监督对比学习可以在一定程度上消解实体类型过度依赖问题。进一步，可以观察到将本章提出的方法应用到不同的基线模型时，与相应的“Baseline”模型相比，其“实体类型过度依赖比例”均显著下降。因此，本章所提方法可以有效降低对实体类型的过度依赖程度。此外，可以观察到“Baseline+type+CL”在处理实体类型过度依赖问题的有效性不如本章所提方法，其原因为普通的监督对比学习主要解决定义2中的语义不一致问题，而没有聚焦于消解实体类型过度依赖问题。

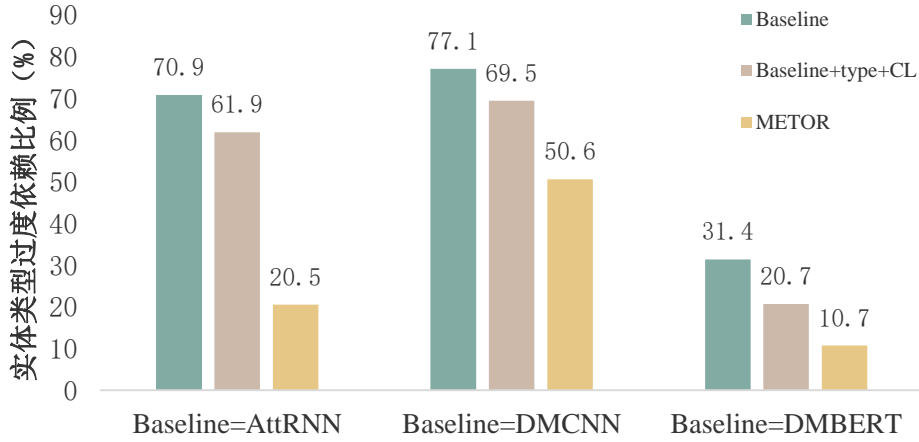


图 4.3 不同模型的实体类型过度依赖程度比较

#### 4.4.5 超参数敏感性分析

本章分析不同超参数对 METOR 模型性能的影响，包括公式4.18中的温度参数  $\tau$ 、公式4.21中的边际参数  $\gamma$  和公式4.24中的循环训练频率  $f$ 。

如图4.4(a)所示，METOR 模型在温度参数  $\tau$  为 0.1 时性能最佳。当其在 0.02 到 0.5 的范围内变化时，METOR 模型的 F1 性能变化幅度不超过 1.1%，这证明了 METOR 模型在不同  $\tau$  的设置下的鲁棒性。

从图4.4(b)中可以观察到，当边际参数  $\gamma$  设置为 0.1 时，METOR 模型的 F1 指标性能达到最佳。当边际参数  $\gamma$  设置为 0.05 时，从负样本角度消解实体类型过度依赖问题的程度存在不足，METOR 模型的 F1 的性能略有下降。而当边际参数  $\gamma$  大于 0.1 时，其性能也会下降。这是由于给定实例和伪正样本集合中的负样本之间的特征表示过度相似，导致正样本处理模块和负样本处理模块间的训练目标差距较大。然而，由于 METOR 模型中的循环训练策略能够极大缓解由较大边际参数  $\gamma$  引起的训练目标不一致性问题，其性能的下降较为有限。

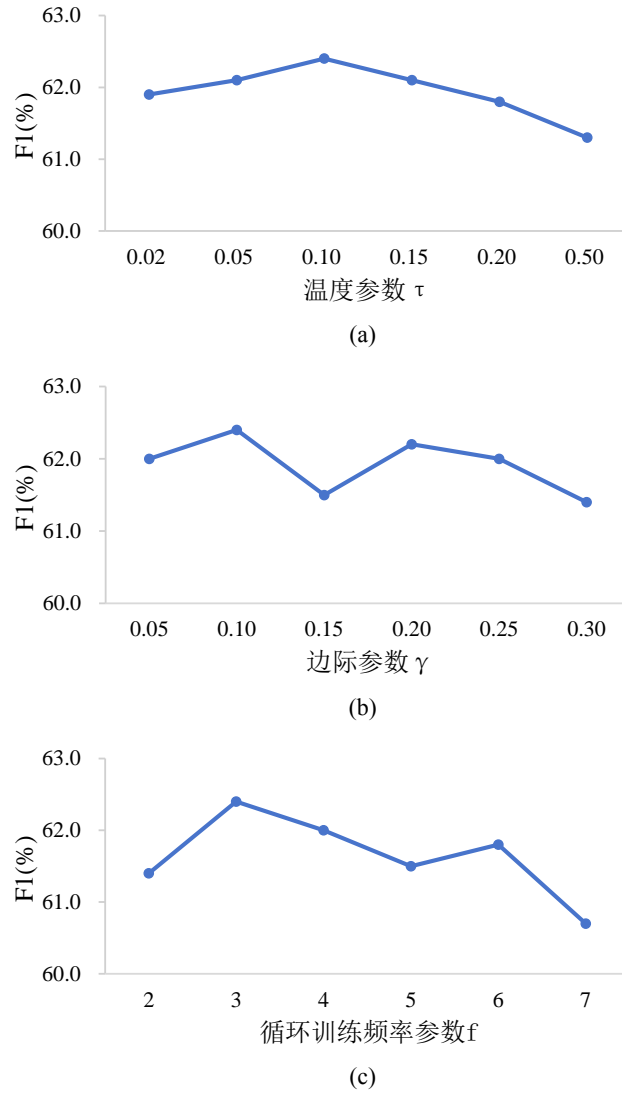


图 4.4 METOR 模型参数敏感性分析

从图4.4(c)中可以观察到,当循环训练频率  $f$  设置为 3 时, METOR 模型实现了最佳性能。当循环频率  $f$  设置为 2 时,循环训练策略中主任务和辅助任务的训练频率没有差异,导致 METOR 模型在两个不一致的训练目标间频繁切换,从而性能出现下降。此外,当循环频率  $f$  大于 3 时,导致循环训练策略中负样本处理模块的训练频率较低,其将接近于退化为不包含负样本处理模块的 METOR 模型。

#### 4.4.6 计算复杂度分析

虽然本章所提方法在实验中应用到了不同的事件要素抽取模型,但计算复杂度的分析是相似的。因此,本节着重讨论基于 DMBERT 的 METOR 模型的情况,具体计算

如下：在编码器模块中，BERT 的计算复杂度主要来源于多头注意力<sup>[67]</sup>，为  $O(N^2d + Nd^2)$ ，其中  $N$  是输入序列的最大长度， $d$  是 BERT 隐藏层输出向量的维度。在正样本处理模块中，语义相关性信息的计算复杂度为  $O((d_s^2 + d_s)n_e)$ ，其中  $d_s$  是事件类型/实体类型/要素类型向量的维度， $n_e$  是给定事件要素抽取数据集中预定义实体类型的数目。对于基于 NT-Xent 的对比学习来说，其计算复杂度为  $O((d + d_s + d_2)d_1 + Kd_2 + K^2)$ ，其中  $d_1$  和  $d_2$  是维度参数， $K$  是批数据大小。在负样本处理模块中，计算复杂度为  $O(d_2)$ 。在循环训练策略模块中，多类分类器的计算复杂度为  $O(n_r(d + d_s))$ ，其中  $n_r$  是给定事件要素抽取数据集中预定义要素类型的数目。

因此，METOR 模型的总计算复杂度为  $O(Nd^2 + (N^2 + d_1 + n_r)d + (d_1 + K)d_2 + n_e d_s^2 + (d_1 + n_r + n_e)d_s + K^2)$ 。如果忽略较小的参数  $n_e$ ， $n_r$  和  $d_s$ ，则其计算复杂度可以简化为  $O(Nd^2 + (N^2 + d_1)d + (d_1 + K)d_2 + K^2)$ 。进一步，与计算复杂度为  $O(Nd^2 + N^2d)$  的基线模型 DMBERT 相比，所提模型的计算复杂度增长了  $O(d_1d + (d_1 + K)d_2 + K^2)$ 。由于在本章的超参数设置中，存在  $d_1 + K < d$ ， $d_2 < d$  和  $K < N$ ，使得增长的计算复杂度  $O(d_1d + (d_1 + K)d_2 + K^2)$  约等于  $O(d^2 + N^2)$ 。因此，可以得知 METOR 模型的计算复杂度略高于 DMBERT，但其 F1 性能显著提高了 5.2%。

#### 4.4.7 案例研究

本章节选取 ACE2005 数据集测试集中的六个实例进行案例研究。表 4.7 展示了使用 DMBERT、DMBERT (CL) 和本章所提的 METOR 模型对这些实例的预测结果，其中 DMBERT (CL) 为表 4.6 中的变体模型 DMBERT+type+CL。

在实例 (1) 中，“rally”触发了 *Demonstrate* 事件，且事件候选要素为“Manhattan”，其实体类型和要素类型分别为 *County – or – District* 和 *Place*。由于实体类型 *County – or – District* 通常以要素类型 *Place* 参与到事件中，为该要素类型预测提供了重要信息。因此，DMBERT、DMBERT (CL) 和本章所提的 METOR 模型通过实体类型 *County – or – District* 提供的语义信息给出了正确的要素类型预测。

在实例 (2) 中，候选事件要素“the court”的实体类型和要素类型分别为 *Government* 和 *Adjudicator*。虽然实体类型 *Government* 无法提供明显的语义信息来预测要素类型，但当“verdict”触发的 *Convict* 事件给出时，其可以为预测提供信息提示。换言之，*Government* 实体经常在 *Convict* 事件中以要素类型 *Adjudicator* 出现，所以三个模型均能正确预测要素类型。因此，前两个实例证明了实体类型可以为预测其频



表 4.7 ACE2005 数据集上的案例分析

实例	实体类型	DMBERT	DMBERT (CL)	METOR
(1) Liana Owen drove 10 hours from Pennsylvania to attend the rally in Manhattan with her parents.	County-or-District	Place ✓	Place ✓	Place ✓
(2) He asked the court to invalidate the verdict and throw out the criminal case against Pasko.	Government	Adjudicator ✓	Adjudicator ✓	Adjudicator ✓
(3) Separately, former WorldCom CEO Bernard Ebbers failed on April 29 to make a first repayment of 25 million dollars on a 400-million-dollar loan from MCI, the Journal said, citing SEC documents.	Individual	Giver ×	Recipient ✓	Recipient ✓
(4) U.S. Defense Secretary Donald H. Rumsfeld discussed the resolution with Prime Minister Tony Blair and Defense Secretary Geoff Hoon on Friday as Rumsfeld returned from a tour of Iraq, Afghanistan and the Persian Gulf region.	Nation	Destination ×	Origin ✓	Origin ✓
(5) It was not clear how many people were in the cafe at the time of the blast.	Indeterminate	None ×	None ×	Target ✓
(6) The space agency has been under intense scrutiny since February when the space shuttle Columbia disintegrated over Texas, killing all seven crew members.	Air	Instrument ×	Instrument ×	None ✓

繁参与的要素类型提供有用的预测线索。

在实例（3）中，在由“loan”触发的 *Transfer – Money* 事件中，候选事件要素“former WorldCom CEO Bernard Ebbers”的实体类型和要素类型分别为 *Individual* 和 *Recipient*。由于要素类型 *Giver* 和 *Recipient* 均与实体类型 *Individual* 在 *Transfer – Money* 事件中频繁关联，DMBERT 无法在该实例中正确区分 *Giver* 和 *Recipient*。然而，DMBERT (CL) 和本章所提的 METOR 模型均受益于对比学习，避免被同一事件中具有相同实体类型的要素类型所混淆，给出了正确的识别结果。实例（4）展示了另一个类似的实例。

在实例（5）中，候选事件要素“how many people”的实体类型和要素类型分别为 *Indeterminate* 和 *Target*。从语义上来看，如果候选要素为一个 *Indeterminate* 实体，则该实体很有可能在事件中不作为要素出现。这种强语义关联导致了 DMBERT 对要素类型的预测错误。进一步，由于多数实体类型为 *Indeterminate* 的候选要素在训练集中不为事件要素出现，DMBERT (CL) 使用传统的对比学习来避免对于实体类型 *Indeterminate* 过度依赖的效果不佳。然而，本章所提的 METOR 模型消解了这种实

体类型过度依赖问题，避免了该错误。

在实例 (6) 中，在由 “killing” 触发的 *Die* 事件中，候选事件要素 “the space shuttle Columbia” 的实体类型和要素类型分别是 *Air* 和 *None*。和实例 (5) 相似，如果候选事件要素为一个 *Air* 实体，则该实体很有可能在事件中以要素类型 *Instrument* 出现，这阻碍了 DMBERT 和 DMBERT (CL) 从文本中理解语义信息，导致它们做出错误的预测。与之相比，METOR 模型正确预测了要素类型。因此，METOR 可以同时减少假负 (false negative) 预测（如实例 (5) 所示）和假正 (false positive) 预测（如实例 (6) 所示），从而取得优秀的事件要素抽取性能表现。

## 4.5 本章小结

本章首先研究了事件要素抽取的实体类型依赖性，并利用预实验评估其对不同事件要素抽取基线模型整体性能的影响。预实验表明了不同基线模型均面临不同程度的实体类型过度依赖问题，从而降低了整体性能。为此，本章提出了基于多角度对比学习的实体类型过度依赖消解方法。该方法分别从正样本和负样本角度消解模型在训练过程中对于实体类型的过度依赖，并设计了一种循环训练策略使这两种对比学习方法能够高效协作。本章基于 ACE2005 数据集进行了充分的实验，结果表明所提方法能够有效地对不同事件要素抽取基线模型进行实体类型过度依赖问题的消解，且当应用在 DMBERT 基线时，性能显著优于当前最佳的事件要素抽取模型。