

基于句法特征的事件要素抽取方法

余子健 朱桐 陈文亮

苏州大学计算机科学与技术学院, 江苏苏州2150062

{zjyu, tzhu7}@stu.suda.edu.cn

wlchen@suda.edu.cn

摘要

事件要素抽取 (Event Argument Extraction, EAE) 旨在从非结构化文本中提取事件参与要素。编码器—解码器 (Encoder-Decoder) 框架是处理该任务的一种常见策略, 此前的研究大多只向编码器端输入文本的字词信息, 导致模型泛化和远程依赖处理能力较弱。为此, 本文提出一种融入句法信息的事件要素抽取模型。首先对文本分析得到成分句法解析树, 将词性标签和各节点的句法成分标签编码, 增强模型的文本表征能力。然后, 本文提出了一种基于树结构的注意力机制 (Tree-Attention) 辅助模型更好地感知结构化语义信息, 提高模型处理远距离依赖的能力。实验结果表明, 本文所提方法相较于基线系统F1值提升2.02%, 证明该方法的有效性。

关键词: 事件要素抽取 ; 成分句法分析 ; 事件抽取

Syntax-aware Event Argument Extraction

Zijian Yu Tong Zhu Wenliang Chen

School of Computer Science and Technology,
Soochow University, Suzhou, Jiangsu, 215006, China

{zjyu, tzhu7}@stu.suda.edu.cn

wlchen@suda.edu.cn

Abstract

Event Argument Extraction (EAE) aims to extract event arguments from unstructured text. The Encoder-Decoder framework is a common solution for this task. Most of the encoders in previous studies only receive plain text information, resulting in weak generalization and remote dependency processing capabilities of the models. To this end, we propose a new event argument extraction model incorporating syntactic information. Firstly, after the text is analyzed to obtain a constituent parsing tree, the part-of-speech tags of words and the syntactic component labels of nodes are encoded to enhance the context representation ability. Then, we propose a tree-structure-based attention mechanism (Tree-Attention) to assist the model to better perceiving structured semantic information and improve the model's ability to handle long-distance dependencies. The experimental results show that the method yields better results with the F1 value by 2.02% compared to the baseline system, which proves the effectiveness of our method.

Keywords: Event Argument Extraction , Constituency Parsing , Event Extraction

* 基金项目: 2020-2024自然科学基金重点联合项目: 自然语言对话交互的基础理论和方法(61936010)

1 引言

事件抽取是一项基础自然语言处理任务，旨在从非结构化文本中抽取结构化事件信息。该任务主要包括事件识别和事件要素抽取两部分。事件识别主要是确定事件属于哪种类型，而事件要素抽取则是根据事件类型预先定义好的事件框架，抽取相应要素类型的论元。

对于事件要素抽取任务的研究，通常是围绕词粒度⁰的语言模型展开。例如Hsi et al. (2016)使用了Word2Vec作为词的向量表示，该方法虽然能够一定程度上表达词的含义，但是由于训练窗口大小的限制，其所学习到的语义信息较为有限。Kenton and Toutanova (2019)提出的BERT语言模型很好地解决了Word2Vec的窗口限制问题，相较之下能够更好地学习到全局信息，这也是目前应用最为广泛的语言模型之一。但同类词¹之间被看作是相互独立的，缺少信息共享，导致泛化能力较弱。并且一些相关联的词随着距离的增大，模型对它们的处理能力也逐渐减弱。例如对中文句子中一个词语的表示，上述模型中并不能很好地体现其之间的依赖关系。传统的事件要素抽取方法通常是基于卷积神经网络(Li et al., 2019; Chen et al., 2015)和递归神经网络(Patchigolla et al., 2017; Zhao et al., 2018)来实现的，这些方法虽然能够学习一些底层的特征，但是同样存在泛化能力较弱和远距离依赖处理能力不强的问题。

为解决上述问题，增强模型的表达能力，本文提出融入成分句法树的模型框架。成分句法解析(Cross and Huang, 2016) (constituency parsing) 能根据给定的句子，分析出句子的短语结构句法树以及词性信息，能够为下游任务提供更丰富的句法语义信息。本文通过Tree-LSTM (Tree Structured Long Short-Term Memory) 将成分句法树进行编码融入Encoder端，并且提出了树注意力机制 (Tree-Attention) 来更好地学习树的结构特征。

总体来说本文包含以下三点贡献：

- 将成分句法解析的结果作为额外特征，借助词性标签和树节点句法标签不同粒度的表示单元对文本进行编码，从而得到更好的文本表示，增强模型泛化能力；
- 利用Tree-LSTM，并提出了树的注意力机制，辅助模型感知句法解析树的结构和特征，并且通过加权的方式将注意力网络进行融合，缓解远距离依赖的问题。进而提高了事件要素抽取的效果；
- 该方案在CCKS2022 Task10数据上的实验结果表明，融入句法信息的事件要素抽取的F1值相较于基线系统提高2.02%，证明了本文所提方案的有效性。

2 相关工作

近些年基于深度学习的文本表示研究主要有以下几点。Zaremba et al. (2014)提出的循环神经网络(Recurrent Neural Network, RNN)，按照时序对文本进行计算，相较于CNN(LeCun et al., 2015)拥有更好的短期记忆Hochreiter and Schmidhuber (1997)提出的长短期记忆网络 (long short-term memory, LSTM) 是RNN的一种变体，通过“输入门”、“输出门”和“遗忘门”的结构，让信息能够按序列传递下去。双向长短期记忆网络 (Bi-LSTM) 是LSTM的优化版本，通过正反两个方向进行单向的信息传递，最终通过拼接或加权的方式进行融合，该方法广泛应用于命名识别任务(成于思and 施云涛, 2020; 镇宇et al., 2019)。Kenton and Toutanova (2019)提出的基于预训练模型BERT (Bidirectional Encoder Rep-resentations from Transformers) 的框架，不仅能够很好地保留局部的关键信息，还能更好地挖掘全局的信息，因此常作为抽取任务的Encoder(藺志 et al., 2022; Zhang et al., 2020)。

现有的事件要素抽取方法，很大程度上忽略了要素之间的关系和相互作用，而这一层信息已被证实能辅助模型更好地进行要素抽取(Li et al., 2021)。不仅要素之间的关系影响模型性能，事件要素类型和事件要素之间也具有强相关性。Xu et al. (2023)显式地利用了要素类型和要素的关系，通过对比学习和循环训练策略，提高了模型的性能。

句法信息能够将句子中不同词联系起来，对要素抽取任务有一定的辅助效果。Ding et al. (2022)针对各要素类型之间的联系，提出了显式角色交互网络，它允许动态捕获事件中不同参数角色之间的相关性，以提高事件要素抽取的效果,但是仅使用了词性标签，并未充分利用文本的结构化信息;Sun et al. (2022)将句法结构信息与命名实体识别任务 (Named Entity Recognition, NER) 结合起来，提升了模型的性能，证明了句法信息有助于模型理解能力的提升;Veyseh et al. (2020)基于图转换网络提出了SemSynGTN模型，充分利用句子的句法和语义信

⁰在中文文本处理时通常以字为粒度

¹例如“可乐”中的“可”与“乐”

息，更好地学习文本的表示和结构，提升模型的要素抽取能力。本文将该工作作为对比模型进行比较。

Tai et al. (2015)等人提出了Tree-LSTM，与标准的LSTM相同，Tree-LSTM的每个单元都包含了三个门，但不同的是Tree-LSTM单元中门向量和细胞状态的更新依赖于所有与之相关的子单元的状态。另外，Tree LSTM拥有多个遗忘门，分别对应当前单元的每个子单元，这使得Tree-LSTM可以选择性地从子节点中获取信息。这种模型结构能够很好地将成分句法树融入文本语义表达中。

3 任务定义与数据

在本文中，抽取任务的输入为一段中文文本 $S = \{w_1, w_2, w_3, \dots, w_n\}$ 以及一段对相应事件的描述 $M = \{m_1, m_2, m_3, \dots, m_t\}$ ，描述 M 是对抽取事件主体的概括。输出为根据预先定义好的槽位抽取的事件论元。其中一段文本 S 所包含的事件描述 M 可以有一个或多个，每个槽位抽取的要素可以为空，也可以有多个。文本 S 与描述 M 均不为空，并且抽取的要素字段均出现在文本 S 中。

事件要素槽位包括以下三个：1)事件主体，表示在该事件描述下的施事方，通常为一个人、组织或机构；2)事件动作，表示事件主体的一些行为，大多为动词或包含动词的短语；3)事件客体，表示该事件的受事方，根据事件的类型不同，可能是一个人、机构、政策或动物等受施事方影响的客体。

如表1给出的样例所示，包含一段文本以及一个事件描述，事件要素包括：1) 事件主体为“美联储”；2) 事件动作为“采取”；3) 事件客体为“缩紧政策”。

文本	美元指数盘尾涨0.13%，报99.97，上周五曾触及100.19，2020年5月以来最高；由于预期美联储将采取更激进的紧缩政策，美债收益率急升，提振美元，指标10年期美债收益率周一达到2.793%的2019年1月以来最高。
事件描述	预期美联储将采取更激进的紧缩政策
事件主体	美联储
事件动作	采取
事件客体	缩紧政策

Table 1: 数据样例

我们使用HanLP(He and Choi, 2021)工具对文本解析，从而获得文本的成分句法结构。如图1所示，系统输入为一段文本，输出为该文本的成分句法树以及分词之后的词性结果。该工具的句法标签遵循PTB3.0(Xue et al., 2000)标注规范，其中，“IP”表示简单子句或句子，通常不带补语（如“的”、“吗”等）；“NP”表示名词短语，中心词通常为名词；“VP”表示动词短语，中心词通常为动词。

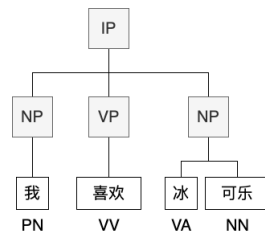


Figure 1: 成分句法解析样例

对于词性标签，假设原始文本为 $S = \{w_1, w_2, w_3, \dots, w_n\}$ ，分词之后得到 k 个词，结果为 $D = \{d_1, d_2, d_3, \dots, d_k\}$ ，其中 d_i 表示第 i 个词。并且 D 中每个元素 d_i 都对应一个词性标签。图1的实例中，“我”、“喜欢”、“冰”、“可乐”四个词的词性标签分别对应“PN（代词）”、“VV（动词）”、“VA（表语形容词）”、“NN（普通名词）”。词性标签的标注准则遵循PTB3.0(Xia, 2000)标注规范。

在该任务中，输入的句子可能包含多个简单句。解析器会自动识别并切分成多个简单句，每个简单句（IP）连接至根节点（Root），构成该句子的完整成分句法树。为了利用句子的句法成分信息，我们将“IP”标签的孩子节点标签作为其叶子节点的句法标签。具体的，在图1的例子中，“IP”标签下包含了“NP”、“VP”和“NP”三个孩子节点标签，则这三个句法标签分别对应了“我”、“喜欢”和“冰可乐”。若出现简单句的嵌套，即“IP”标签的孩子节点中存在另外一个“IP”标签（下称为子“IP”节点）。除子“IP”节点外的孩子节点与上述方法一致，作为其叶子节点的句法标签，而子“IP”节点则继续向下确定句法标签，以此类推。若数据中出现一个词语作为一个子句的情况，则失去了句法树的结构信息。

本文采用CCKS2022 Task10²面向金融领域的因果事件要素抽取学术评测提供的数据作为实验数据。该数据集主要来自公开的新闻和报告，共有4,000条标注规范的样本，其中包含9232个事件，平均一条文本包含2个事件，平均句子长度为191个字。在实验中，对该数据使用随机采样方法，将数据集打乱，按8:1:1的比例划分为训练集、开发集及测试集。

我们对句法分析结果进行统计分析，发现事件要素与词性标签以及句法标签有着密不可分的关联。其中，事件主体81.42%的句法标签为名词短语，词性标签包含63.13%的名词以及22.44%的形容词与名词的组合；事件动作88.74%的句法标签为动词短语，79.02%的词性标签为动词；事件客体94.12%的句法标签为名词短语，85.75%的词性标签为名词。

4 模型框架

本文考虑如何提高模型泛化能力和远距离表示能力，提出一种融合句法特征的事件要素抽取模型。我们的模型利用词性标签和句法标签建立不同词之间的联系增强模型泛化能力，基于句法树结构建立远距离依赖。

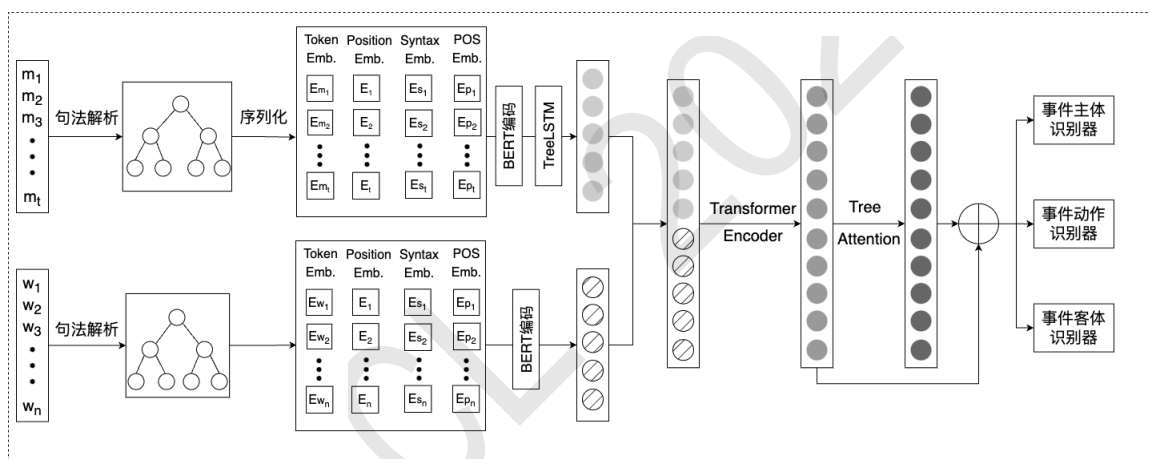


Figure 2: 总体模型框架图

图2是总体模型结构图，主要包括四个步骤：1) 利用成分句法解析器对输入文本进行分析，得到成分句法树、词性和各叶子节点的句法标签。分别将句法标签和词性标签通过BMES的标注模式打上标签并编码。2) 将事件描述M的成分句法树通过Tree-LSTM进行编码，再与第一步的编码进行拼接，作为Transformer Encoder的输入，并利用Transformer Encoder进行信息的交互。3) 将第二步的结果作为Tree-Attention的输入，利用注意力机制，增强成分句法树在隐藏向量中的表示，并将增强后的向量与第二步的结果通过加权的方式进行融合，得到最终的语义表示。4) 使用多层二元标志解码器对句子表示进行解码，识别出各事件要素的首尾位置，最终得到各槽位的论元。本节详细介绍我们的要素抽取模型，主要分为三个模块：编码、训练以及解码。

4.1 编码

4.1.1 词性及句法标签的嵌入

我们采用BMES标注模式对词性和句法标签进行表示，其中B表示某一标签内容的开始部分；M表示该字符为某标签内容的中间部分；E表示某标签的最后一个字符；S表示该字符单独

²<https://www.biendata.net/competition/ccks-nec-2022/>

作为一个标签的内容。具体而言，图1给出的例子中，“我”的词性标签为“S-PN”，“冰可乐”三个字符的句法标签分别对应“B-NP”，“M-NP”，“E-NP”。

将字符的词性及句法标签分别进行编号，并且进行随机初始化，之后连同文本一起作为BERT的输入，经过计算得到的句子表示为 $\mathbf{H} = \{\mathbf{h}_1, \dots, \mathbf{h}_n\}$, $\mathbf{H} \in \mathbb{R}^{L \times i}$ ，计算过程如式1所示。

$$\{\mathbf{h}_1, \dots, \mathbf{h}_n\} = \text{BERT}(x_t, x_c, x_s) \quad (1)$$

其中， x_t 为句子的字符， x_c 与 x_s 分别为词性标签与句法标签，与字符一一对应。每个字符的向量表示 $\mathbf{h}_i \in \mathbb{R}^i$ ， i 表示向量的维度， L 为最大句长。

4.1.2 基于Tree-LSTM的句法树编码

本节介绍基于Tree-LSTM的事件描述M的成分句法树的编码方案。基于LSTM的树编码方式主要有N-ary Tree-LSTM和Child-Sum Tree-LSTM。其区别在于，前者所编码的树中除了叶子节点外的每一个节点的孩子数相同，而后者对于树的结构没有特别的要求。由于文本的多样性，文本句法解析树的结构往往无法满足N-ary LSTM的要求，因此本文采用Child-Sum Tree-LSTM的方法对成分句法树进行编码。

与传统的LSTM相同，Tree-LSTM包含输入门 (i_j)、遗忘门 (f_{jk})、输出门 (o_j)、记忆单元 (c_j) 以及隐藏状态 h_j ，但不同的是Tree-LSTM循环单元需要处理一个或多个子节点的隐藏状态和单元状态的信息。为了让关键信息在传递的过程中尽可能保留下来，Tree-LSTM的循环单元对每个子节点单独设立遗忘门。因此Tree-LSTM循环单元包含一个输入门、一个输出门、一个单元状态和多个遗忘门。

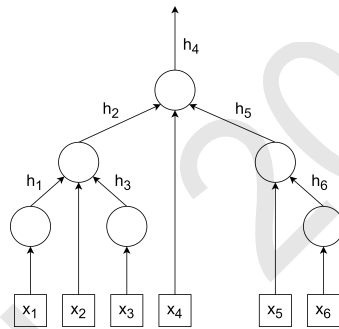


Figure 3: Tree-LSTM模型结构图

Tree-LSTM的结构如图3所示，不同于传统LSTM的线性循环单元，Tree-LSTM在树形结构上将循环单元自底向上递归展开，下面将具体计算过程及原理。在计算每个节点时，令该节点为 j ，其所有子节点的集合为 $C(j)$ ，细胞状态为 c_j ，隐含状态输出为 h_j 。节点 j 的输入为当前节点 j 的输入 x_j 以及所有子节点的隐含状态 h_i ($i \in C(j)$)。对于每个子节点 $i \in C(j)$ 都设置了一个遗忘门，以确定并保留子节点的哪一部分信息。每个子节点 $k \in C(j)$ 的遗忘门输出取决于当前节点输入 x_j 和子节点的隐含状态输出 h_k ，并且经过sigmoid函数激活得到权重值 f_{jk} ，计算过程如下式所示：

$$f_{jk} = \sigma(W^{(f)}x_j + U^{(f)}h_k + b^{(f)}) \quad (2)$$

取得所有子节点隐含状态之后，输入门首先对其求和得到 \tilde{h}_j ，根据 \tilde{h}_j 和当前节点输入 x_j 决定输入门权重值 i_j 及备选细胞状态 u_j ，之后根据遗忘权重和输入权重更新当前节点的状态，具体计算公式如下：

$$\begin{cases} \tilde{h}_j = \sum_{k \in C(j)} h_k \\ i_j = \sigma(W^{(i)}x_j + U^{(i)}\tilde{h}_j + b^{(i)}) \\ u_j = \tanh(W^{(u)}x_j + U^{(u)}\tilde{h}_j + b^{(u)}) \\ c_j = i_j \odot u_j + \sum_{k \in C(j)} f_{jk} \odot c_k \end{cases} \quad (3)$$

输出门根据当前节点的输入 \mathbf{x}_j 和子节点隐含状态之和 $\tilde{\mathbf{h}}_j$ ，计算得到输出门的权重 \mathbf{o}_j ，再通过当前细胞状态得到当前节点的输出 \mathbf{h}_j ，计算公式如下：

$$\begin{cases} \mathbf{o}_j = \sigma \left(W^{(o)} \mathbf{x}_j + U^{(o)} \tilde{\mathbf{h}}_j + b^{(i)} \right) \\ \mathbf{h}_j = \mathbf{o}_j \odot \tanh(\mathbf{c}_j) \end{cases} \quad (4)$$

其中， σ 表示sigmoid函数，通过sigmoid函数将向量的数值控制在0-1之间， \odot 为点积表示矩阵对应元素相乘。 W 和 U 为可学习的权重矩阵， b 为各个门的偏置， \tanh 为激活函数。

该部分对经过Tree-LSTM编码后的成分句法树和经过词性及句法增强的文本表示进行信息交互，以增强模型对成分句法树的感知，并且能够区分同一文本不同事件的隐藏表示。具体计算如下式所示：

$$\mathbf{h} = \text{BERT}(\text{concat}(\mathbf{h}_m, \mathbf{h}_t)) \quad (5)$$

其中， concat 表示拼接操作， \mathbf{h}_m 为成分句法树的语义表示， \mathbf{h}_t 为词性和句法增强的文本语义表示。

4.1.3 融入树结构的注意力机制

为了进一步增强树的语义表达，我们提出了树的注意力机制（Tree-LSTM），通过依次计算父子两个词之间的相似度，将句法树中除叶子节点外的两个词联系起来。其计算过程如下。

首先在注意力机制中，输入向量与三个不同的变换矩阵相乘的到 \mathbf{q} 、 \mathbf{k} 、 \mathbf{v} ，分别为查询向量，关键词向量和值向量。每个关键词向量依次与查询向量进行点积运算计算相似度，再利用softmax进行归一化，将相似度转化为和为1的得分矩阵，最后与值矩阵相乘的到Attention的计算结果，具体计算如下所示：

$$\text{Attention}(\mathbf{q}, \mathbf{k}, \mathbf{v}) = \text{softmax} \left(\frac{\mathbf{q} \cdot \mathbf{k}^T}{\sqrt{d_w}} \right) \cdot \mathbf{v} \quad (6)$$

其中 d_w 为输入向量的维度。

	IP	NP	我	VP	喜	欢	NP	冰	可	乐
IP	1	1	1	1	1	1	1	1	1	1
NP	1	1	1	0	0	0	0	0	0	0
我	1	1	1	0	0	0	0	0	0	0
VP	1	0	0	1	1	1	0	0	0	0
喜	1	0	0	1	1	1	0	0	0	0
欢	1	0	0	1	1	1	0	0	0	0
NP	1	0	0	0	0	0	1	1	1	1
冰	1	0	0	0	0	0	1	1	1	1
可	1	0	0	0	0	0	1	1	1	1
乐	1	0	0	0	0	0	1	1	1	1

Figure 4: Mask矩阵示意图

Tree-Attention是在注意力机制中，添加一个二维的掩码（Mask）矩阵，矩阵元素1表示参与注意力的计算，0则表示不参与。Mask矩阵根据成分句法树的父子关系来确定，除叶子节点外，其余节点均会参与其所有子孙节点的注意力计算。具体地，如图4所示为“我喜欢冰可乐”的Mask矩阵。Tree-Attention部分的输入为经过语义交互层Transformer Encoder的向量 \mathbf{X} ，输出为经过Tree-Attention计算后的向量 \mathbf{Y} ，最终的文本表示 \mathbf{h} 通过公式7对 \mathbf{X} 和 \mathbf{Y} 加权融合所得，其中 λ 为0到1(含)之间的数。

$$\mathbf{h} = \lambda \cdot \mathbf{X} + (1 - \lambda) \cdot \mathbf{Y} \quad (7)$$

4.2 解码及训练

本文采用多层二元解码器对语义表示进行解码，以得到各槽位的事件论元。对于每个槽位，我们利用两个全连接层，将语义的隐层表示映射到二维的向量中，分别来预测其论元的起始位置和结束位置。计算过程如下所示：

$$p_i^{(s,c)} = \sigma(W_s^c \cdot \mathbf{h}_i + b_s^c) \quad (8)$$

$$p_i^{(e,c)} = \sigma(W_e^c \cdot \mathbf{h}_i + b_e^c) \quad (9)$$

其中, h_i 为第*i*个隐藏表示, s 、 e 表示开始和结束位置。 $p_i^{(s,c)}$ 和 $p_i^{(e,c)}$ 分别表示第*i*个位置为要素类型*c*的开始和结束位置的概率。 W 为可学习参数, b 为偏置, σ 为激活函数。该部分模型的训练目标函数如下式所示:

$$loss = - \sum_{i=1}^n r_i^{(s,c)} \log(p_i^{(s,c)}) - \sum_{i=1}^n r_i^{(e,c)} \log(p_i^{(e,c)}) \quad (10)$$

其中*n*为文本序列的长度, $r_i^{(s,c)}$ 和 $r_i^{(e,c)}$ 分别为标注数据的正确分类标签。之后通过以下规则确定最终抽取的事件要素: 1)若在预测时没有预测出起始位置或者结束位置, 则判定为该文本中不存在相应槽位的事件要素; 2)若起始位置之后没有结束位置或者结束位置之前没有起始位置, 则判定该结果为非法结果, 不算作最终答案。

5 实验与分析

5.1 评价指标

在实验中, 本文采用微平均的精确率(Precision,P)、召回率(Recall,R)、F1值(F1-measure,F1)来评价事件要素抽取的效果, 即所有的事件要素类型一起计算P、R及F1值, 具体计算方法如下式所示:

$$P_{micro} = \frac{\sum_{i=1}^n TP_i}{N_{predict}} \quad (11)$$

$$R_{micro} = \frac{\sum_{i=1}^n TP_i}{N_{gold}} \quad (12)$$

$$F1_{micro} = 2 \cdot \frac{P_{micro} \cdot R_{micro}}{P_{micro} + R_{micro}} \quad (13)$$

其中 TP_i 表示第*i*类事件要素预测正确的数量, $N_{predict}$ 为所有预测出的要素数量, N_{gold} 表示所有金标答案的数量。除此之外, 在预测时仅当预测的要素与标准答案的要素完全一致才算正确。

5.2 实验设置

本实验使用开源的预训练语言模型roberta-base-wwm³作为我们编码层的一部分。我们根据模型在开发集的性能设置超参数。最大句长为512, 向量维度采用默认值768, mini-batch设置为16, 迭代次数为20, 采用Adam优化器以2e-5的学习率优化模型。为防止过拟合的情况, 最终的隐藏表示通过dropout层随机丢失一些节点, 丢失的比例为0.5。 λ 用于调节Tree-Attention过程中的向量加权融合的占比, 为0.5。

5.3 对比模型

为了对比前人的事件要素抽取方法与本文方法的差异, 我们选择了以下五种模型进行对比:

- BERT+Bi-LSTM+CRF(Jiang et al., 2019): 该模型将事件要素抽取作为序列化标注的问题, 依靠其独特的门控机制, 能够很好地学习并保留文本的局部和全局信息。该方法使用Roberta-base作为第一层Encoder, 提取句子特征, 紧接着利用Bi-LSTM的双向信息传播的特点对文本的信息进行交互并编码, 最后通过CRF (Conditional Random Fields) 进行解码, 得到最终的抽取结果。
- BERT+GlobalPointer(Tao et al., 2022): 该方法利用全局归一化的思想, 并且引入旋转式位置编码 (RoPE), 融入了要素在文中的开始和结束位置信息, 最后通过要素首尾对的打分, 得到抽取的结果, 相较于CRF的解码方式更加合理, 且泛化性更好。
- BERT+Biaffine(Yu et al., 2020): 该模型基于预训练语言模型对文本进行特征提取, 利用Biaffine模型在多层神经网络输出句子中所有可能的要素的分数, 然后对这些进行排序, 返回各个槽位分数最高的要素作为答案。

³<https://huggingface.co/hfl/chinese-roberta-wwm-ext>

- MRC+SPAN(Li et al., 2020): 该模型将事件的描述作为针对文本提出的问题, 并且通过特殊符号拼接在文本前面作为BERT的输入。最后通过三个二元分类器得到最终抽取的要素结果。
- SemSynGTN(Veyseh et al., 2020): 该模型利用了依存句法信息, 将句法和语义信息转换为一个加权有向图, 其中节点表示句子中的词, 边表示依存句法和语义角色关系, 权重表示边的重要程度, 并且引入了图转换网络, 每个图转换层通过聚合邻居节点和边的信息来更新当前节点的表示。最终通过全连接层抽取相应的要素。

5.4 实验结果

5.4.1 性能对比分析

方法	P/%	R/%	F1/%
BERT+Bi-LSTM+CRF	54.815	55.224	55.019
BERT+GlobalPointer	57.423	57.124	57.312
BERT+Biaffine	58.102	57.761	57.931
MRC+SPAN	58.682	58.146	58.413
SemSynGTN	60.024	59.113	59.565
本文方法	61.457	59.613	60.521

Table 2: 不同抽取方法的实验结果

表2是本文系统与对比模型的实验结果。从表中结果, 我们发现如下现象:

BERT+Bi-LSTM+CRF无法针对不同的事件提取不同的语义特征, 因此在本文的数据集上效果不佳。BERT+GlobalPointer和BERT+Biaffine可以利用特殊符号, 将不同事件的信息与文本进行交互, 提高了语义的表征能力, F1值相较于Bi-LSTM+CRF有明显提升。

MRC+SPAN利用了阅读理解的机制, 将事件的Mention作为问题, 并利用[CLS]和[SEP]特殊符号进行拼接作为输入并通过transformer进行信息交互, 学习并保存了不同事件描述但相同文本的不同语义表示, 再通过对要素起始位置和结束位置的预测, 得到最终抽取的结果。我们将该模型作为基础模型框架(基线模型)搭建本文系统。

本文方法和SemSynGTN的F1值均高于其他系统, 说明句法信息的确对事件要素抽取有帮助。而本文方法相较于SemSynGTN有0.956%的优势, 说明本文方法能更好的利用成分句法信息来帮助本文的抽取任务, 原因可能在于树结构相较于图结构更能体现句法树的依赖关系。

从表2的结果来看, 本文的方法相较于BERT+Bi-LSTM+CRF提升了5.50%。对比BERT+GlobalPointer和BERT+Biaffine也有3.209%和2.59%的优势。从而证实了该方案的有效性。和他们方法最大的差异在于, 本文方法将句法特征融入模型中, 提升模型对语义强关联词的信息捕捉能力, 一定程度上缓解了远距离依赖的问题。

总体来看, 本文的方法相较于前人方法在性能上有着一定优势。相对于基线模型MRC+SPAN也有2.02%的提升。说明本文方法能够较好地融合成分句法树的信息, 辅助模型挖掘更深层的信息, 提高了事件要素抽取的精确率和召回率。

5.4.2 λ 对结果的影响分析

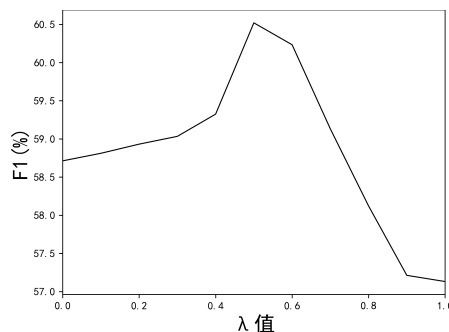


Figure 5: 模型的F1值随 λ 趋势变化图

通过Tree-LSTM和Transformer Encoder的文本嵌入与经过Tree-Attention后向量表示的融合是本文方法的核心。为探究Tree-Attention是否对模型有帮助以及其所占权重为多少的效果最佳，我们通过观察F1值随 λ 值的变化趋势，选择F1值最高的 λ 值作为融合的权重。

在开发集上，F1值随 λ 变化的趋势如图5所示，整体趋势随 λ 先上升后下降，在 λ 取0.5时达到峰值。从曲线在 $\lambda=0.5$ 前后变化可以发现，句法信息和文字信息之间的重要程度接近。当 $\lambda=1.0$ 时，模型仅对句法结构进行编码，此时出现的较快下降意味着简化的句法结构文本中含有的信息量较少。因此需要将文字信息和句法结构信息进行融合，同时增加模型的文本表征和长句依赖捕捉能力。

5.4.3 消融实验

本文模型在基础模型的基础上额外增加了4个重要组件，分别是词性嵌入、句法嵌入、句法树嵌入以及Tree-Attention。其中，上一节的实验表明，以0.5的权重融入Tree-Attention相比未添加Tree-Attention组件F1值提升了1.81%，充分证明其有效性，因此本节不再对该部分进行论证。在本节中，我们对其余3个组件进行消融实验，分别在基础模型上单独添加其中1个组件。

	P/%	R/%	F1/%
基础模型	58.682	58.146	58.413
+词性嵌入	62.026	56.320	59.035
+句法标签嵌入	61.153	57.374	59.203
+句法树嵌入	61.042	57.721	59.335
+以上三种组件	61.230	58.202	59.678

Table 3: 消融实验结果

消融实验结果如表3所示，词性嵌入、句法嵌入以及句法树嵌入三者相较于基础模型F1值分别有0.622%、0.79%以及0.922%的提升，说明三个组件均有助于模型性能提高。

词性标签来自于对分词后每个词的词性预测，词性嵌入将词性标签作为一种特征融入至模型之中。由于融入词性标签的分词以及词性信息，使得模型相较于基础模型能够更好地对文本进行表示，进而能够更好地识别要素在原文中的起始与结束位置，缓解了要素抽取的边界错误问题。

句法标签能够清晰表示文本中各个子句的组成成分，并且各字句的句法标签与抽取的要素有着密不可分的关系。例如事件动作的要素类型，其在子句中的句法标签通常为“VP”（动词短语）。依靠句法标签的句法嵌入能够辅助模型对要素的定位，缓解要素抽取错误以及非空槽位抽不出要素的问题。

句法树嵌入充分利用了文本的句法解析结果，不仅包含了句法标签的信息，还将成分句法树的结构信息融入模型，辅助模型对上下文的语义理解。句法树嵌入也是该三个组件中提升效果最明显的。

5.4.4 远距离依赖的效果分析

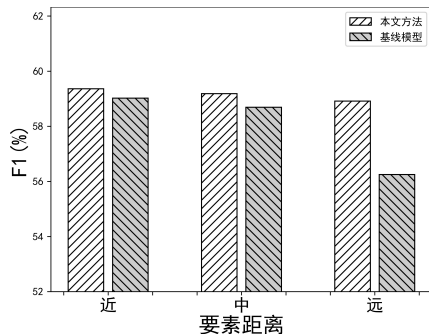


Figure 6: 模型F1值随主客体距离变化图

为了探究本文方法是否缓解远距离依赖的问题。根据事件主体与事件客体之间的距离，我

们将开发集划分成三等份，分别为远、中、近，并统计基线模型和本文模型在三个集合中事件主体与事件客体的F1值。

结果如图6所示，在事件主体与事件客体距离较近时，基线模型和本文方法效果接近。但随着其距离的增大，基线模型的F1值下降较为明显，本文方法在主客体之间距离较远时仍然有较好的表现。充分证明了本文方法对缓解远距离依赖问题的有效性。

5.4.5 实例分析

为更加直观地体现本文模型相较于基线模型的提升，本小节将展示具有代表性的3个文本以及抽取结果的实例。如表4所示，其中文本部分由于长度限制，仅展示含有要素的局部文本，省略上下文信息。

NO.	实例			
1	文本	...上海和吉林省是中国顶尖的汽车制造中心。最近，两地都实施了严格的疫情防控措施...		
	要素类型	事件主体	事件动作	事件客体
	金标答案	上海和吉林省	实施	严格的疫情防控措施
	基线模型	上海和吉林省	实施	严格的疫情
本文模型	上海和吉林省	实施	严格的疫情防控措施	
2	文本	...每一台进入冬奥会的服务机器人都经过严格的筛选评测，正是有了这些机器人的加入，才使得参会嘉宾有了别样新体验...		
	要素类型	事件主体	事件动作	事件客体
	金标答案	这些机器人	加入	冬奥会
	基线模型	这些机器人	加入	None
本文模型	这些机器人	加入	冬奥会	
3	文本	...还有部分区域受疫情影响，春耕用肥运输受阻，化肥难以送到田间地头。...		
	要素类型	事件主体	事件动作	事件客体
	金标答案	疫情	影响	部分区域
	基线模型	部分区域	受疫情影响	None
本文模型	疫情	影响	部分区域	

Table 4: 金标及预测的实例

实例1与实例2中，基线模型与本文模型均能正确识别事件主体和事件动作，但在事件客体的抽取中，实例1的基线模型由于边界问题未能完全正确地抽取正确答案，而本文模型融入句法特征，能更精准地识别各要素，减少边界问题带来的错误；实例2的基线模型未能识别出事件客体，原因可能是识别的要素为诸如“冬奥会”的专有名词时，基线模型并不能很好地理解，进而导致识别错误或者识别不出答案，相较之下，本文模型经过句法语义增强，即使要素为专有名词，也能在句法特征的加持下正确识别。

实例3是该任务的一个难题，原因主要是被动语态导致事件主体与事件客体的混淆，并且被动语态在数据集中的占比较少无法充分学习该特征。而本文模型将成分句法树进行编码，即使是被动语态也能够很好地识别各要素的成分。

6 总结与展望

为了提高模型泛化能力和远距离依赖处理能力，本文提出了一种基于句法特征的事件要素抽取模型。在所提模型中，我们利用词性和句法标签来提升模型泛化能力，通过对成分句法树建模来改进远距离依赖的处理能力。实验结果表明，本文提出的抽取模型能够充分利用成分句法信息，提高了系统性能。

同时，我们的工作也存在一些需要改进的地方。例如，本文使用的预训练语言模型是在大量语料上进行无监督训练，只包含文本的浅层语义信息。若在预训练阶段就融入句法信息等的深层特征，抽取性能可能会有进一步的提升。这也将是我们未来研究的一个重点。

参考文献

- Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 167–176.
- James Cross and Liang Huang. 2016. Span-based constituency parsing with a structure-label system and provably optimal dynamic oracles. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1–11.
- Nan Ding, Chunming Hu, Kai Sun, Samuel Mensah, and Richong Zhang. 2022. Explicit role interaction network for event argument extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3475–3485.
- Han He and Jinho D Choi. 2021. The stem cell hypothesis: Dilemma behind multi-task learning with transformer encoders. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5555–5577.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Andrew Hsi, Yiming Yang, Jaime G Carbonell, and Ruo Chen Xu. 2016. Leveraging multilingual training for limited resource event extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1201–1210.
- Shaohua Jiang, Shan Zhao, Kai Hou, Yang Liu, Li Zhang, et al. 2019. A bert-bilstm-crf model for chinese electronic medical records named entity recognition. In *2019 12th international conference on intelligent computation technology and automation (ICICTA)*, pages 166–169. IEEE.
- Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.
- Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature*, 521(7553):436–444.
- Junyi Li, Xiaobing Zhou, Yuhang Wu, and Bin Wang. 2019. Ynu-junyi in bionlp-ost 2019: Using cnn-lstm model with embeddings for seedev binary event extraction. In *Proceedings of The 5th Workshop on BioNLP Open Shared Tasks*, pages 110–114.
- Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A unified mrc framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5849–5859.
- Qian Li, Hao Peng, Jianxin Li, Jia Wu, Yuanxing Ning, Lihong Wang, S Yu Philip, and Zheng Wang. 2021. Reinforcement learning-based dialogue guided event extraction to exploit argument relations. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:520–533.
- Rahul VSS Patchigolla, Sunil Sahu, and Ashish Anand. 2017. Biomedical event trigger identification using bidirectional recurrent neural network based models. In *BioNLP 2017*, pages 316–321.
- Mengtao Sun, Qiang Yang, Hao Wang, Mark Pasquine, and Ibrahim A Hameed. 2022. Learning the morphological and syntactic grammars for named entity recognition. *Information*, 13(2):49.
- Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566.
- Shuheng Tao, Yi Chen, and Jiping Wang. 2022. A global pointer based entity relation extraction method for chinese pulmonary nodule medical records. In *2022 IEEE 16th International Conference on Application of Information and Communication Technologies (AICT)*, pages 1–8. IEEE.
- Amir Pouran Ben Veyseh, Tuan Ngo Nguyen, and Thien Huu Nguyen. 2020. Graph transformer networks with syntactic and semantic structures for event argument extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3651–3661.

- Fei Xia. 2000. The part-of-speech tagging guidelines for the penn chinese treebank (3.0). *IRCS Technical Reports Series*, page 38.
- Jing Xu, Dandan Song, Siu Cheung Hui, Fei Li, and Hao Wang. 2023. Multi-view entity type overdependency reduction for event argument extraction. *Knowledge-Based Systems*, 265:110375.
- Nianwen Xue, Fei Xia, Shizhe Huang, and Anthony Kroch. 2000. The bracketing guidelines for the penn chinese treebank (3.0). *IRCS Technical Reports Series*, page 39.
- Juntao Yu, Bernd Bohnet, and Massimo Poesio. 2020. Named entity recognition as dependency parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6470–6476.
- Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent neural network regularization. *CoRR*, abs/1409.2329.
- Hongkuan Zhang, Hui Song, Shuyi Wang, and Bo Xu. 2020. 基于bert 的端到端中文篇章事件抽取(a bert-based end-to-end model for chinese document-level event extraction). In *Proceedings of the 19th Chinese National Conference on Computational Linguistics*, pages 390–401.
- Yue Zhao, Xiaolong Jin, Yuanzhuo Wang, and Xueqi Cheng. 2018. Document embedding enhanced event detection with hierarchical and supervised attention. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 414–419.
- 成于思 and 施云涛. 2020. 融合词典特征的bi-lstm-wcrf 中文人名识别. *中文信息学报*, 34(4):69–76.
- 镇宇, 蒋盛益, 张礼明, and 包睿. 2019. 基于多特征bi-lstm-crf 的影评人名识别研究. *中文信息学报*, 33(3):94–101.
- 蔺志, 李原, and 王庆林. 2022. 基于bert 改进的文化活动事件论元抽取研究. *中文信息学报*, 36(12):115–122.