

文章编号: 1003-0077(2020)08-0051-10

# 利用门控机制融合依存与语义信息的事件检测方法

陈佳丽,洪宇,王捷,张婧丽,姚建民

(苏州大学 计算机科学与技术学院,江苏 苏州 215006)

**摘要:** 句子级事件检测任务目的是识别和分类事件触发词。现阶段工作主要将句子作为神经分类网络的输入,学习句子的深层语义信息,从而优化句子表示来改进事件检测任务的性能。该文发现除句子语义信息外,依存树包含的句法结构信息也有助于获取准确的句子表示。为此,该文采用双向长短时记忆网络对句子进行编码,捕获其语义信息;同时,设计图神经网络对句子的依存结构进行表示,获取其依存信息;此外,在对句子进行语义编码与依存编码时,该文利用自注意力机制使模型选择性地关注句子中的不同词,从而捕获句子中有助于事件检测的关键信息,并尽可能避免无关词的干扰;最后,该文提出门控机制,通过加权实现上述两种信息的动态融合。该文在自动文本抽取(automatic content extraction, ACE)数据集上进行实验,结果显示,该文提出的动态融合语义信息与依存信息的方法能更加有效地对句子进行编码,并捕获句子中的事件信息,在触发词识别与事件类型分类这两个子任务中, $F_1$  值均有较大提升,分别达到 76.3%和 73.9%。

**关键词:** 语义信息;依存信息;门控机制;事件检测

**中图分类号:** TP391      **文献标识码:** A

## Combination of Dependency and Semantic Information via Gated Mechanism for Event Detection

CHEN Jiali, HONG Yu, WANG Jie, ZHANG Jingli, YAO Jianmin

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

**Abstract:** Sentence-level Event Detection (ED) is a task of identifying and classifying event triggers. Existing approaches mainly use sentences as the input of the neural classification network and learn the deep semantic information of sentences. Base on the fact that the dependency tree contains rich syntactic structure features for more accurate sentence representation, we use a Bidirectional Long Short-Term Memory (Bi-LSTM) to learn semantic information, and use a Graph Convolutional Network (GCN) to learn dependency information. To concentrate more on event-related information and reduce the interference of redundant words, we add self-attention on the Bi-LSTM and GCN respectively. Finally, we propose to use the gated mechanism to dynamically fuse semantic information and dependency information. The experimental results on ACE show that the performance of the proposed method reaches 76.3% and 73.9% in  $F_1$ -score for trigger identification and event type classification, respectively.

**Keywords:** semantic information; dependency information; gated mechanism; event detection

## 0 引言

事件抽取任务旨在从非结构化自由文本中抽取用户感兴趣的事件信息。其包含四个子任务:触发词识别、事件类型分类、论元识别和论元角色分类。通常,将前两个子任务合并,称为事件检测(event

detection, ED)。事件检测的目的是识别出给定句子中的触发词,并将其分类为正确的事件类型,以 ACE 2005<sup>①</sup> 数据集中的例 1 为例,事件检测系统需要识别出“arrested”和“killings”为触发词,并为触发词分别分配事件类型“Arrest-Jail(逮捕)”和“Die

① <https://catalog.ldc.upenn.edu/Ldc2006t06>

(死亡)”。

**例 1** Police have **arrested** four people in connection with the **killings**.

(译文: 警方逮捕了与杀人案有关的四人。)

目前,深度学习被广泛应用于事件检测任务中。根据句子的表示方法不同,其可以分为两种形式:基于语义表示<sup>[1]</sup>和基于句法依存表示<sup>[2]</sup>。基于句法依存表示的方法通常使用依存树,依存树以多叉树形式表示句子中的依存关系,通过连接触发词与关键词以准确定位与触发词最相关的信息;同时,若句子中存在多个事件,通过依存表示可以增强多个事件间的信息流动。如图 1(例 1 的依存树结构图)所示,触发词“**arrested**”与其两个论元(即事件的参与者)“**police**”和“**people**”直接相连,模型利用边可将相邻节点的信息融入当前节点中,能有效捕获直接与触发词相连的关键词信息,从而有助于提高事件类型分类的准确性。此外,一个句子中的多个事件之间大多是相互关联的,但仅仅利用语义信息对句子进行编码难以获取多个事件间的联系。如例 1 中的两个事件同时出现在一句话中,若用基于语义表示的模型——双向长短时记忆网络(bidirectional long short-term memory, Bi-LSTM)<sup>[3]</sup>进行编码, Bi-LSTM 中句子长距离依赖问题将导致“**arrested**”与“**killings**”间的相互作用弱;而在图 1 的依存结构中,触发词“**arrested**”与“**killings**”只需通过三条边相连,因此两者联系较 Bi-LSTM 编码更加密切,事件间信息流动性更强,这对含有多个事件的单个句子的事件识别与分类有较大帮助。

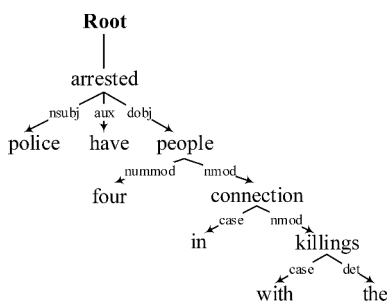


图 1 例 1 的依存树

基于依存树结构,图卷积网络(graph convolutional network, GCN)<sup>[4]</sup>可以有效地编码句子的依存信息,但由于其只能构建依存树中当前节点与相邻节点间的信息,故存在编码范围较短、编码信息不完整的问题;同时,如果相邻节点为无义词,这种编码方式会加大这些无义词的影响,并且那些非相邻却携带关键信息的节点将无法被关注。如例 1 中的

“**have**”虽然为触发词“**arrested**”的直接相邻节点,但其并未携带与事件相关的信息,可将它们视为无义词,因此,这些节点不应得到过多的关注。针对该问题,本文提出以 GCN 网络结构为基础,融合多头自注意力机制(multi-head self-attention)<sup>[5]</sup>的句子依存关系编码方式。当利用 GCN 网络获取相邻节点信息时,注意力机制为图中所有节点赋予不同的权重,使模型关注到其他关键节点;同时,采用多头自注意力机制让模型在不同的表示子空间中学习句子中的关键信息,增强模型的表征能力。

此外,为充分获取句子的语义表示,本文采用双向长短时记忆网络(bidirectional long short-term memory, Bi-LSTM)<sup>[3]</sup>对句子进行编码,基于顺序相关性学习句子的上下文语义特征;同时,融合自注意力机制(self-attention)<sup>[5]</sup>,重点关注句子中的关键词,以实现与事件相关的关键信息的筛选,并减弱无义词影响。在分别获得给定句中的依存信息表示与语义信息表示后,本文设计了一种门控机制对上述两种表示进行动态融合,并合理分配这两种表示以提升事件检测的性能。

综上所述,本文主要做出如下贡献:

(1) 在事件检测任务中,首次提出将句中的语义表示信息与依存关系信息进行动态融合,以提高句子的表征能力;

(2) 设计基于多头自注意力机制的图卷积网络捕获句子的依存信息;设计基于自注意力机制的双向长短时记忆网络捕获句子的语义信息;采用门控机制合理融合两种信息;

(3) 通过在 ACE2005 语料上的大量实验,结果表明本文提出的方法能够在事件检测任务上取得较好的性能,触发词识别和事件类型分类的  $F_1$  值分别达到 76.3% 和 73.9%。

下面本文将重点从以下几个方面介绍具体工作:①相关工作;②任务定义;③方法;④实验;⑤总结与展望。

## 1 相关工作

目前事件检测研究主要针对触发词识别和事件类型分类进行,并取得一定的成果。首先介绍基于特征的事件检测方法,Grishman<sup>[6]</sup>通过三个最大熵模型统计分类;Ahn<sup>[7]</sup>使用模块化方法进行事件抽取;Ji 等<sup>[8]</sup>提出跨文档推理;Gupta 等<sup>[9]</sup>提出跨事件推理以缓解论元缺失问题;Grishman<sup>[10]</sup>提出融合

文档信息实现句子级事件类型分类;Hong 等<sup>[11]</sup>提出跨实体推理提升事件抽取任务的性能;Liu 等<sup>[12]</sup>提出全局推理方法实现事件检测任务;Zeng 等<sup>[13]</sup>使用远程监督方法生成高质量的数据集。

另一种是基于深度学习的事件检测方法。Nguyen 等<sup>[14]</sup>首次将卷积神经网络(convolutional neural networks, CNN)方法用于事件检测任务;Chen 等<sup>[15]</sup>提出动态多池卷积神经网络;Nguyen 等<sup>[16]</sup>提出非连续卷积方法缓解非连续信息忽略问题;Nguyen 等<sup>[1]</sup>首次采用双向循环神经网络(bidirectional recurrent neural network, Bi-RNN)联合抽取事件触发词和事件元素;Duan 等<sup>[17]</sup>提出文档级递归神经网络方法;Feng 等<sup>[18]</sup>提出混合神经网络方法;Liu 等<sup>[2]</sup>提出联合多事件抽取;Chen 等<sup>[19]</sup>提出一种基于门控多层注意力机制的层级偏置网络;Zhao 等<sup>[20]</sup>提出利用文档级向量改进 Bi-RNN 的新颖模型;Hong 等<sup>[21]</sup>提出利用生成对抗网络生成虚假特征自我调节的方法;Liu 等<sup>[22]</sup>提出门控多语言注意力模型。

同时,已有相关事件检测研究利用了依存关系,例如,Li 等<sup>[23]</sup>提出基于结构化感知机模型融合全局和局部特征,将依存关系作为全局特征联合抽取事件触发词和论元;Sha 等<sup>[24]</sup>提出基于依存关系桥的循环神经网络,将依存关系作为特征输入到循环神经网络中,联合抽取事件触发词和论元。而本文通过基于多头自注意力机制的图卷积网络学习句子的依存关系表示,对依存树中的边加权,让模型有选择地关注依存树中的节点,并利用门控机制动态地融合句子的语义表示与依存表示,有助于模型学习句子的语义信息,同时关注到依存关系。

图卷积网络(graph convolutional network, GCN)<sup>[4]</sup>借助图谱理论实现拓扑图上的卷积操作,近期在自然语言处理上得到了广泛使用。例如,Cao 等<sup>[25]</sup>使用实体图中节点间的关系及查询和实体图间的注意力信息实现多条推理问答任务;Guo 等<sup>[26]</sup>使用注意力机制引导图卷积神经网络,充分利用依存树信息,从而实现实体关系抽取。

## 2 任务定义

本文专注于 ACE<sup>[27]</sup>评估系统中定义的事件检测任务。本节将简单阐述 ACE 中定义的相关术语及事件检测任务。

**实体(entity)** 某个语义类别的对象或对象集

合,例如,人、物、地点等实体词;

**实体描述(entity mention)** 某个特定类型的实体短语;

**事件触发词(event trigger)** 描述事件发生的核心词,常为名词或短语,由单个词或短语组成;

**事件元素(event argument)** 事件的参与者,是事件的核心部分,由实体、时间、值组成;

**事件描述(event mention)** 包含触发词、事件参与者的短语或句子。

通常,一个事件包含特定类型的触发词和与该事件相关的一个或多个参与者。事件检测任务包含两个子任务:触发词识别和事件类型分类。即先识别出句子中的事件触发词,再对事件触发词进行事件类型分类。下面对事件检测任务进行形式化的定义: $X$  为数据集来源上的句子,针对每个句子  $x = \langle t, c \rangle (t = \{x_t\}$  为候选触发词,  $c = \{1, \dots, x_{t-1}, x_t, \dots, x_n\}$  为触发词在句中的上下文,  $n$  为句子长度),都有触发词标签  $y_{\text{识别}} \in [0, 1]$  (1 表示候选词为触发词, 0 表示候选词为非触发词)和事件类型标签  $y_{\text{分类}} \in [0, 33]$  (ACE 2005 中定义 33 种事件子类型, 0 表示非事件类型)。

## 3 方法

与之前研究者们<sup>[14-18]</sup>的工作类似,本文仍将事件检测任务当作多分类任务,将句子中的每个词当作候选触发词,目的是对候选词进行 34 种分类(33 种事件类型及 1 种非事件类型)。本文先使用基于多头自注意力机制(multi-head self-attention)的 GCN 学习依存表示,再将 Bi-LSTM 与自注意力机制(self-attention)结合学习句子语义表示,最后利用门控机制将语义表示与依存表示动态融合,以进行触发词识别与事件类型分类,模型的基本架构如图 2 所示。

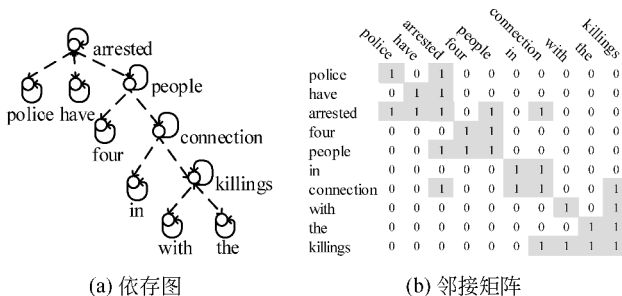


图 2 例 1 的依存图与邻接矩阵

本节从以下几个方面介绍具体方法:①向量层,即模型的输入层;②依存表示层;③语义表示

层; ④门控机制; ⑤输出层; ⑥训练过程。

### 3.1 向量层

本文按照 Nguyen 等<sup>[14]</sup>的输入处理方式, 仍采用目标句子表示作为模型的输入。通过查找向量表

将目标句中的每个词转换成实值向量, 以补全或截断的方式固定句子的长度, 可以将其表示为  $X = \{x_1, x_2, \dots, x_i, \dots, x_n\}$ ,  $n$  为句子长度。模型的输入向量由词向量、实体类型向量和词性向量拼接, 具体如图 3 所示。

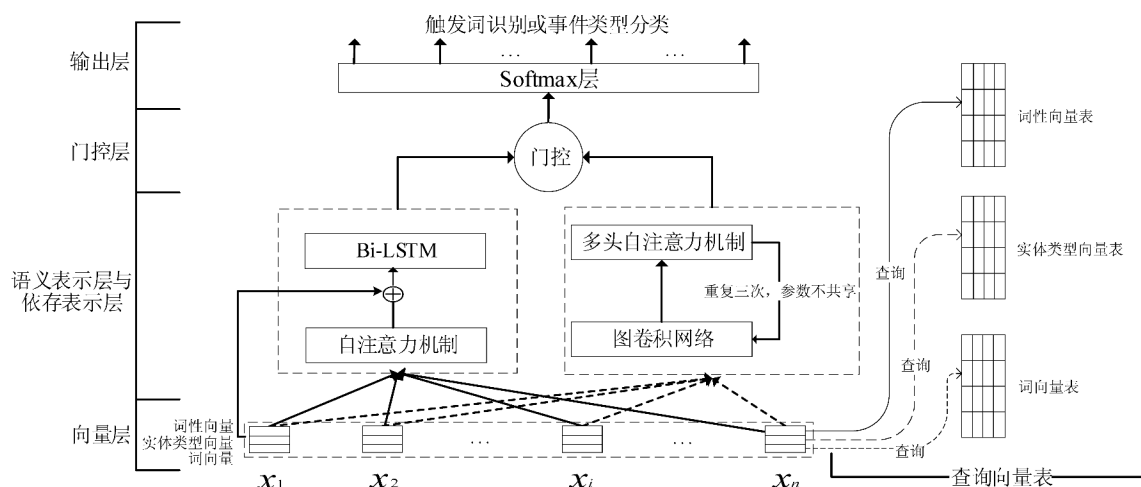


图3 系统结构图

**词向量** 本文使用 Skip-gram<sup>[28]</sup> 模型训练 NYT<sup>①</sup> 语料中的词向量。句子中每个单词的词向量表示  $w$  拼接后得到词级的句子向量表示  $W$ :  $W = \{w_1, w_2, \dots, w_i, \dots, w_n\}$ , 其中,  $w_i$  表示第  $i$  个词的隐藏语义, 词向量通过查找词向量表获得。

**实体类型向量** 与词向量一致, 实体类型向量被映射到与句子长度一致的实值矩阵中。此处加入的实体特征是语料中标注的实体, 若句子中的词为实体部分时, 则该词的实体标签为实体类型, 否则将其标记为“0”。例如, 句子中实体的分布式表示为:  $E = \{0, e_i, \dots, e_j, 0\}$ , 其中,  $e_i$  或  $e_j$  表示第  $i$  或  $j$  个词的实体类型, “0”表示该位置上的词不是实体部分。通过统计实体类型(包括非实体类型“0”)的个数, 随机初始化实体类型向量表, 并从该表中查找句子中每个词的实体类型向量  $W_e$ 。

**词性向量** 与实体类型向量一致, 词性向量被映射到与句子长度一致的实值矩阵中。本文使用 Stanford CoreNLP<sup>②</sup> 工具识别句中每个词的词性, 并且统计词性类型个数, 将词的词性标签标记为词性类型。通过随机初始化方式构建词性类型向量表, 并查找该表以获取句子中每个词的词性向量  $W_p$ 。

最后, 将词向量  $W$ 、实体类型向量  $W_e$  及词性向量  $W_p$  拼接, 构成模型输入  $X$ , 如式(1)所示。

$$X = [W, W_e, W_p] \quad (1)$$

### 3.2 依存表示层

图卷积网络 (graph convolutional network, GCN) 由 Kipf 等<sup>[4]</sup> 提出, 该方法将卷积神经网络应用于图数据。

给定一个无向图:  $G=(V, E)$ ,  $V$  和  $E$  分别表示图中的节点与边的集合。本文基于句子依存树构建图, 图中节点为句中的每个词表征, 边则是  $w_i$  到  $w_j$  的有向句法依存边, 其中,  $w_i$  或  $w_j$  表示第  $i$  或  $j$  个词表征。为增强 GCN 对依存关系编码, 本文沿用了 Guo 等<sup>[2]</sup> 的做法, 在树中添加了反向边与自循环边, 如图 2(a) 所示, 在例 1 的依存树中加入反向边与自循环边; 图 2(b) 根据该可视化图构建了  $n \times n$  的邻接矩阵  $A$  ( $n$  为句中词的个数), 若  $w_i$  到  $w_j$  间有边相连, 则  $A_{ij} = A_{ji} = 1$ , 否则,  $A_{ij} = A_{ji} = 0$ 。本文连续使用了 3 层的 GCN 网络, 若第  $l$  层的第  $i$  个节点表示为  $g_i^l$ , 则下一层的依存表示通过第  $l$  层表示与邻接矩阵间进行矩阵乘积获得, 表示为  $g_i^{l+1}$ , 具体如式(2)所示。

$$g_i^{l+1} = f\left(\sum_{j \in N_i} \frac{1}{c_{ij}} A_{ij} g_j^l W_j^l + g_i^l W_0^l\right) \quad (2)$$

其中,  $N_i$  为图中所有与第  $i$  个节点有连边的节

① <https://catalog.ldc.upenn.edu/LDC2008T19>

② <http://corenlp.run/>



点集合,  $j \in N_i$  表示节点集合中的第  $j$  个节点,  $c_{ij}$  是通过节点的度对特征表征进行归一化,  $\mathbf{W}_j^i$  与  $\mathbf{W}_0^i$  均为权重矩阵,  $f$  为激活函数  $\text{relu}$ 。此外, 使用词向量  $\mathbf{X}$  初始化第一层 GCN 的  $g_i^0$  值, 并且三层的 GCN 权重参数不共享。

使用依存树构建的图易导致模型只关注图中有边相连的节点, 而忽略那些无边相连却携带关键信息的节点, 因此, 本工作在 GCN 网络后融入了多头自注意力机制, 给图中所有节点分配不同的权重, 目的是在获取依存信息的同时关注到图中其余关键节点信息。

多头自注意力机制由 Vaswani 等<sup>[5]</sup>提出, 以计算句子中每个词与其余词间的相关性。与不同的是本文未引入位置特征, 原因在于基于多头自注意力机制的 GCN 只需给依存树中的边赋予不同的权重, 学习句子的依存表示, 而无须关注节点间的位置信息。多头自注意力机制有三个输入, 分别为“Query(查询)”“Key(键)”“Value(值)”, 记为  $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ 。由于对第  $l$  层依存表示  $\mathbf{G}^l$  进行自注意力机制, 因此,  $\mathbf{Q}=\mathbf{K}=\mathbf{V}=\mathbf{G}^l$ 。

首先, 对  $\mathbf{Q}, \mathbf{K}, \mathbf{V}$  三者进行线性变换, 即分别将三者输入到三个单层神经网络层中, 得到的句子表示为  $\tilde{\mathbf{Q}}, \tilde{\mathbf{K}}, \tilde{\mathbf{V}}$ , 其输出维度和输入维度相同, 具体如式(3)所示。

$$\begin{bmatrix} \tilde{\mathbf{Q}} \\ \tilde{\mathbf{K}} \\ \tilde{\mathbf{V}} \end{bmatrix} = \mathbf{W} * \begin{bmatrix} \mathbf{Q} \\ \mathbf{K} \\ \mathbf{V} \end{bmatrix} + \mathbf{b} \quad (3)$$

然后, 将经过全连接层后的  $\tilde{\mathbf{Q}}, \tilde{\mathbf{K}}, \tilde{\mathbf{V}}$  向量的最后一维划分为三段, 即形成  $\bar{\mathbf{Q}}_{1,2,3}, \bar{\mathbf{K}}_{1,2,3}, \bar{\mathbf{V}}_{1,2,3}$ , 其目的是在不同的表示子空间中学习句子关键词; 接着计算候选词  $\bar{\mathbf{Q}}_{1,2,3}$  与其余词  $\bar{\mathbf{K}}_{1,2,3}$  之间的相关性, 经过 softmax 函数获得得分权重; 再将权重作用于  $\bar{\mathbf{V}}_{1,2,3}$  以得到最终的输出  $\bar{\mathbf{O}}_{1,2,3}$ , 如式(4)所示。

$$\bar{\mathbf{O}}_{1,2,3} = \text{softmax} \left( \frac{\bar{\mathbf{Q}}_{1,2,3} * \bar{\mathbf{K}}_{1,2,3}^T}{\sqrt{d_k}} \right) * \bar{\mathbf{V}}_{1,2,3} \quad (4)$$

其中,  $d_k$  表示  $\bar{\mathbf{K}}$  的维数, 比例因子  $\sqrt{d_k}$  是为了缓解  $d_k$  较大时, 模型易进入 softmax 函数的“饱和区”及梯度较小的问题。

最后将  $\bar{\mathbf{O}}_1, \bar{\mathbf{O}}_2, \bar{\mathbf{O}}_3$  拼接, 得到多头自注意力机制最终结果  $\bar{\mathbf{O}} = [\bar{\mathbf{O}}_1, \bar{\mathbf{O}}_2, \bar{\mathbf{O}}_3]$ 。

### 3.3 语义表示层

为获得句子的深层语义表示及上下文依赖, 本文选用双向长短时记忆网络 (bidirectional long

short-term memory, Bi-LSTM)<sup>[3]</sup>学习句子语义表示; 同时, 为让模型更加关注句中关键的词, 并减少无意义词的干扰, 本文先使用自注意力机制 (self-attention)<sup>[5]</sup>对词表征进行学习, 再将其输入至 Bi-LSTM 网络中, 以辅助 Bi-LSTM 更好地学习句子表示, 增强关键词在句中所能发挥的作用。该部分使用的自注意力机制实质上是 3.2 节提到的多头自注意力机制, 其区别在于未将  $\tilde{\mathbf{Q}}, \tilde{\mathbf{K}}, \tilde{\mathbf{V}}$  的最后一维进行划分, 减少了模型的复杂度, 也可称其为单头自注意力机制。在这里, 本文将单头自注意力机制得到的表示定义为  $\mathbf{U}$ 。

Bi-LSTM 是由前向长短时记忆网络 (long short-term memory, LSTM)<sup>[29]</sup>和后向 LSTM 组合, 适用于时序数据建模, 可以捕获双向语义依赖, 并且能有效地结合上下文信息。每个词向量  $\mathbf{u}_i$  经前向 LSTM 和后向 LSTM 编码后得到的隐状态分别表示为  $\vec{h}_i$  和  $\overleftarrow{h}_i$ , 最后, 将  $\vec{h}_i$  和  $\overleftarrow{h}_i$  拼接作为  $\mathbf{u}_i$ , 经过 Bi-LSTM 得到的最终语义表示  $\mathbf{h}_i$ , 如式(5)所示。

$$\mathbf{h}_i = [\vec{h}_i, \overleftarrow{h}_i] \quad (5)$$

### 3.4 门控机制

为将句子语义表示  $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_i, \dots, \mathbf{h}_n\}$  ( $n$  为句子长度)、依存表示  $\mathbf{G}$  动态地融合, 本文使用了门控机制, 具体如式(6)、式(7)所示。

$$\mathbf{S} = [\mathbf{W} \times \mathbf{G}, \mathbf{H}] \quad (6)$$

$$\mathbf{S}' = (\mathbf{S}) \odot \mathbf{S} \quad (7)$$

其中,  $[\cdot]$  表示拼接,  $\mathbf{W}$  为权重矩阵,  $\odot$  为元素乘,  $\sigma$  为激活函数 sigmoid。

### 3.5 输出层

为了预测最终结果, 首先使用全连接层, 计算每个候选词的事件类型得分, 然后使用 softmax 函数对其归一化, 得到其条件概率  $P(y|x, \theta)$ , 具体如式(8)所示。

$$P(y|x, \theta) = \text{softmax}(\mathbf{W} \times \mathbf{S}' + \mathbf{b}) \quad (8)$$

其中,  $\mathbf{W}$  是权重矩阵,  $\mathbf{b}$  为偏置,  $\theta$  表示模型中的所有参数。本文将每个候选词的条件概率最高的事件类型作为该候选词的最终预测结果。

### 3.6 训练

模型训练时, 目标函数为交叉熵损失, 目的是为了最小化损失值, 目标函数如式(9)所示。

$$L(\theta)=-\sum_{i=1}^l\sum_{j=1}^n\hat{y}_{ij}\log P(y_j|x_i,\theta)\tag{9}$$

其中,  $\hat{y}_{ij}$  表示语料中的第  $i$  个词触发第  $j$  个事件类型,  $P(y_j|x_i,\theta)$  表示模型预测的第  $i$  个词触发第  $j$  个事件类型的条件概率,  $l$  表示句子中所有候选词的个数,  $n$  表示事件类型的个数。本文使用 Adam 优化函数来更新模型中所有的参数  $\theta$ 。

4 实验

4.1 实验设置

数据和评估指标

本文在 ACE 2005 数据集上进行了大量实验。为了比较, 划分数据集的方式与前人一致<sup>[14-17]</sup>, 将 40 篇文章作为测试集, 30 篇文章作为开发集, 剩余 529 篇文章作为训练集。评估方法和 Li 等<sup>[23]</sup>的工作一致, 使用准确率(Precision,  $P$ ), 召回率(Recall,  $R$ ),  $F_1$  值( $F_1$ -score)作为评估指标。

超参数设置

本文词向量是利用 Skip-gram 模型训练 NYT 语料获得的, 模型使用的超参数如表 1 所示。

表 1 模型超参数

参数名	数量	参数名	数量
词向量	300 维	LSTM 隐含单元	100
实体类型向量	50 维	batch 大小	10
词性向量	50 维	句子最大长度	80
GCN 层数	3	Dropout	0.2
GCN 隐含单元	300	学习率	2E-5
多头注意力机制	3 头	Adam 权重衰减率	1E-8

4.2 与先进的方法对比

前沿的事件检测方法主要分为两类: 基于特征的方法与基于神经网络的方法。本文选用了以下几种前沿方法进行比较, 下面首先介绍基于特征的方法。

**跨事件**(cross-event)<sup>[30]</sup>由 Liao 和 Grishman 提出, 使用跨事件推理提升事件抽取任务的性能。

**跨实体**(cross-entity)<sup>[11]</sup>由 Hong 等提出, 使用跨实体推理提升事件抽取任务的性能。

**概率软逻辑**(probabilistic soft logic, PSL)<sup>[31]</sup>由 Liu 等提出, 用来对全局信息编码。

上述均为基于特征的方法。随着深度学习的热潮推进, 现阶段用于事件检测任务的方法主要采用深度学习, 下面将介绍几种基于深度学习的方法。

**动态多池卷积神经网络**(dynamic multi-pooling convolutional neural networks, DMCNN)<sup>[15]</sup>由 Chen 等提出, 使用动态多池层自动抽取文本中的词汇级和句子级特征。

**双向循环神经网络**(Bidirectional recurrent neural networks, Bi-RNN)<sup>[1]</sup>由 Nguyen 等提出, 有效捕获论元角色和触发词子类型间的依赖性。

**混合神经网络**(hybrid neural network, HNN)<sup>[18]</sup>由 Feng 等提出, 将双向长短时记忆网络和卷积神经网络结合, 捕获序列信息和块信息。

**人工神经网络**(artificial neural networks, ANN)<sup>[32]</sup>由 Liu 等提出, 通过监督的注意力方法使用论元信息辅助事件检测任务。

**联合多个事件抽取**(jointly multiple events extraction, JMEE)<sup>[2]</sup>由 Liu 等提出, 使用基于注意力机制的图卷积网络增强句子间信息流动。

**门控多语言注意力**(gated multilingual attention, GMLATT)<sup>[22]</sup>由 Liu 等提出, 利用跨语言的方法解决语义不明确的问题。

**基于门控多层注意力机制的分层和偏置标签网络**(hierarchical and bias tagging networks with gated multi-level attention mechanisms, HBTNG-MA)<sup>[19]</sup>由 Chen 等提出, 利用句子级和文档级信息识别句子中的多个事件。

本文所提的模型与上述几种方法在触发词识别和事件类型分类这两个任务上的性能对比如表 2 所示, 可发现本文提出的方法优于目前先进的方法(前人工作的实验性能数据均引入相应参考文献中的公开数据), 在触发词识别与事件类型分类任务上,  $F_1$  值分别能达到 76.3% 与 73.9%, 比 JMEE 方法高出 0.4% 和 0.2%。

本文方法性能优良的主要原因在于:

(1) 神经网络方法比传统基于特征的方法更易缓解错误传播问题, 学习深度语义信息。

(2) Bi-LSTM 比 Bi-RNN 和 CNN 更适用于本任务, 其擅长捕捉序列长距离依赖特征。另外, 基于自注意力机制的 Bi-LSTM 在捕捉句中上下文特征时, 也关注到句中的关键词。

(3) 传统使用依存树的方式是将其作为特征加入模型中, 而本文通过图卷积网络将依存信息动态融入句中, 这更利于缓解句法工具带来的错误传播。

表 2 所有方法的性能

方法	触发词识别			事件类型分类		
	P/%	R/%	F <sub>1</sub> /%	P/%	R/%	F <sub>1</sub> /%
Cross-Entity	n/a	n/a	n/a	72.9	64.3	68.3
Cross-Event	n/a	n/a	n/a	68.7	68.9	68.8
PSL	n/a	n/a	n/a	75.3	64.4	69.4
DMCNN	80.4	67.7	73.5	75.6	63.6	69.1
Bi-RNN	68.5	75.7	71.9	66.0	73.0	69.3
ANN	n/a	n/a	n/a	76.8	67.5	71.9
GMLATT	80.9	68.1	74.1	78.9	66.9	72.4
HBTNGMA	n/a	n/a	n/a	77.9	69.1	73.3
HNN	80.8	71.5	75.9	84.6	64.9	73.4
JMEE	80.2	72.1	75.9	76.3	71.3	73.7
Ours	78.3	74.5	<b>76.3</b>	75.5	72.4	<b>73.9</b>

此外,多头自注意力机制对图中所有节点分配了不同的权重,让模型不仅能关注到依存信息,还不易忽略对事件检测任务有效的节点信息,也能增强事件间的信息流动。

(4) 门控机制有效地将语义信息与依存信息动态结合,使模型能自动选择与任务相关且重要的信息,从而提升事件检测任务的性能。

4.3 自注意力机制的作用

本文将自注意力机制分别添加到语义表示方法与依存表示方法中,目的是为计算当前词与其余词间的相似度,以便模型关注到句中与事件相关的词。以“引言”中例 1 的多头自注意力机制为例,其可视化结果如图 4 所示,从图中可看出触发词“arrested”和“killings”得分最高,表明自注意力机制可关注到触发词为本任务的关键词。其次,论元“police”和

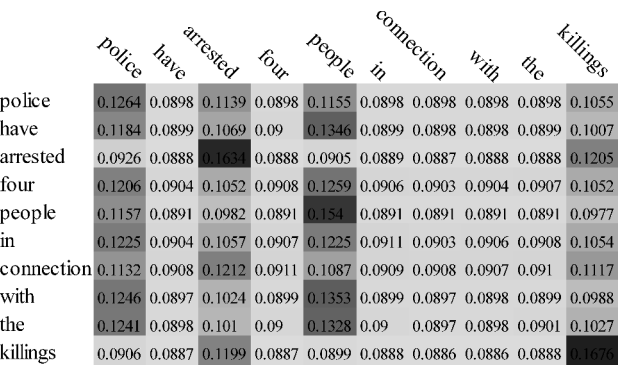


图 4 例 1 的可视化自注意力结果

“people”得分也相对较高,在依存树中,“police”和“people”与触发词“arrested”有边相连,表明自注意力机制能关注到对该任务有帮助的词。

此外,本文利用多头自注意力机制对自注意力机制进行扩展,尝试设置不同头数的自注意力机制,实验结果如表 3 所示。表 3 上方显示在语义表示方法上使用的自注意力机制不同头数的结果(Att-0 表示未使用自注意力机制)。从数据中可看出,加上单头自注意力机制会对 Bi-LSTM 起辅助作用,但是在 Bi-LSTM 前使用多头自注意力机制,性能反而下降,这表明多头自注意力机制在不同的子空间中学习到的句子表示会给 Bi-LSTM 增添负担,从而导致模型复杂度过高,不利于 Bi-LSTM 学习特征表示。表 3 下方数据为在依存表示方法上使用不同头数的自注意力机制的性能。从表中数据得知,在 GCN 网络上使用三头自注意力机制时,触发词识别和事件类型分类任务的 F<sub>1</sub> 值达到最高。这表明 GCN 能通过多头自注意力机制在不同的表示子空间中学习不同的句子表示,更关注句中的关键词,并尽可能缓解依存工具带来的噪声传播。

表 3 自注意力机制不同头的影响

方法	触发词识别			事件类型分类		
	P/%	R/%	F <sub>1</sub> /%	P/%	R/%	F <sub>1</sub> /%
Att-0 <sub>(Bi-LSTM)</sub>	<b>78.7</b>	72.9	75.7	<b>76.2</b>	70.1	73.3
Att-1 <sub>(Bi-LSTM)</sub>	78.3	<b>74.5</b>	<b>76.3</b>	75.5	<b>72.4</b>	<b>73.9</b>
Att-2 <sub>(Bi-LSTM)</sub>	77.9	72.0	74.8	75.9	70.1	72.9
Att-3 <sub>(Bi-LSTM)</sub>	75.2	73.3	74.3	72.6	70.8	71.7
Att-1(GCN)	73.4	<b>74.9</b>	74.2	71.6	<b>73.1</b>	72.4
Att-2(GCN)	75.6	72.6	74.1	73.7	70.8	72.2
Att-3(GCN)	<b>78.3</b>	74.5	<b>76.3</b>	<b>75.5</b>	72.4	<b>73.9</b>

4.4 门控机制的作用

本文利用门控机制将语义信息与依存信息动态结合。表 4 对这两种信息的融合方式进行了比较,一种是将语义表示与依存表示拼接,另一种是用门控机制将语义表示与依存表示动态融合。由表 4 可知,利用门控机制方法的触发词识别与事件类型分类的性能均高于第一种方式,并且 F<sub>1</sub> 值分别提升 0.4%和 0.3%。这表明适当的门控机制有利于语义信息和依存信息的融合。

表 4 门控机制的比较

方法	触发词识别			事件类型分类		
	P/%	R/%	F <sub>1</sub> /%	P/%	R/%	F <sub>1</sub> /%
拼接	75.5	76.3	75.9	73.2	74.0	73.6
门控机制	78.3	74.5	<b>76.3</b>	75.5	72.4	<b>73.9</b>

4.5 语义表示方法与依存表示方法的互作用

为检测语义表示方法与依存表示方法的互作用,本文分别对语义表示模型、依存表示模型独立训练,分析独立模型与两者结合的模型的性能。

**语义表示模型** 表 5 上方展示了语义表示模型的性能。以 Bi-LSTM 为基准模型,其在触发词识别和事件类型分类的 F<sub>1</sub> 值分别达到 74.7%和 71.6%。在该基准模型上加上自注意力机制,F<sub>1</sub> 值均得到提升,分别达到 75.3%和 72.7%。因此,本文选用 Self-Att+Bi-LSTM 模型作为语义表示模型。

**依存表示模型** 表 5 中间展示了依存表示模型的性能。在多头自注意力机制指导的 GCN 下,触发词识别和事件类型分类的 F<sub>1</sub> 值分别达到 73.1%和 70.3%,性能高于仅使用图卷积融合依存关系的方法。这表明多头自注意力机制能辅助图卷积动态有效地融合依存信息,同时关注到句中对事件检测任务起作用的词信息。因此,本文选用基于多头自注意力机制的 GCN 作为依存表示模型。

**两者融合(语义表示方法和依存表示方法结合)** 表 5 最后一行展示了两种方法融合的性能,均高于语义表示模型与依存表示模型取得的性能。融合后的模型在触发词识别和事件类型分类任务上,F<sub>1</sub> 值均能达到 76.3%和 73.9%。这充分表明语义表示与依存表示结合可以相互弥补各自的缺陷,最终能提升事件检测任务的性能。

表 5 语义表示模型、依存表示模型及两方法融合的性能

方法	触发词识别			事件类型分类		
	P/%	R/%	F <sub>1</sub> /%	P/%	R/%	F <sub>1</sub> /%
Bi-LSTM	76.8	72.7	74.7	73.6	69.7	71.6
Self-Att+Bi-LSTM	73.7	77.2	75.3	71.9	73.6	72.7
GCN	69.6	76.3	72.8	66.5	72.9	69.5
GCN+M-head-Att	70.0	76.6	73.1	67.2	73.6	70.3
融合模型	78.3	74.5	<b>76.3</b>	75.5	72.4	73.9

4.6 错误分析

本节对实验结果进行了分析,发现错误主要集中在以下两类。

(1) 模型识别出句子中的触发词,但触发词的事件类型被分类错误。如例 2 中“**fighting**”为触发词,事件类型为“**Demonstrate(示威)**”,但模型预测其触发词事件类型为“**Attack(攻击)**”,其主要原因在于训练集中没有事件类型为“**Demonstrate**”,触发词为“**fighting**”的样本,模型未学习到触发词“**fighting**”可以被分类为事件类型“**Demonstrate**”。未来研究中,可以尝试扩充语料或加入语言模型增强预训练词向量的语义表示。

**例 2** Secession by Iraqi Kurds could inspire Turkey’s rebel Kurds, who for 15 years have been fighting for autonomy.

(译文:伊拉克库尔德人的分裂可能会激发土耳其叛军库尔德人,他们为自治战斗已有 15 年了。)

(2) 模型未能识别出指示代词为触发词,也没能对其事件类型正确分类。如例 3 中,触发词为“**this**”,事件类型为“**Transport(运输)**”,但模型未能识别出触发词“**this**”,原因在于单个句子中给定的信息量过少。因此,在未来研究中,可以考虑利用句子在段落中的上下文信息,例如当前句子的前后句子,利用句子的上下文语义表示辅助事件检测任务,从而提升事件检测任务的性能。

**例 3** But **this** is also conditional on police approval.

(译文:但这也是要得到警方批准的条件。)

5 总结与展望

本文分别采用基于自注意力机制的 Bi-LSTM 与基于多头注意力机制的 GCN 网络学习语义表示与依存表示,让模型学习句子的语义信息与依存关系时,关注到句中的关键词;为合理分配语义信息与依存信息,提出利用门控机制动态地融合语义表示与依存表示。实验结果表明,本文提出的方法在事件检测任务上取得了较好的性能。在未来的工作中,将考虑句子的上下文信息,利用句子的上下文语义表示提升事件检测任务的性能。



## 参考文献

- [1] Nguyen Thien Huu, Cho Kyunghyun, Grishman Ralph. Joint event extraction via recurrent neural networks[C]//Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016: 300-309.
- [2] Liu Xiao, Luo Zhunchen, Huang Heyan. Jointly multiple events extraction via attention-based graph information aggregation[J]. arXiv preprint arXiv: 1809.09078, 2018.
- [3] Schuster Mike, Paliwal Kuldip K. Bidirectional recurrent neural networks[J]. IEEE Transactions on Signal Processing, 1997, 45(11): 2673-2681.
- [4] Kipf Thomas N, Welling Max. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv: 1609.02907, 2016.
- [5] Vaswani Ashish, Shazeer Noam, Parmar Niki, et al. Attention is all you need[J]. arXiv preprint arXiv: 1706.03762, 2017.
- [6] Grishman Ralph, Westbrook David, Meyers Adam. NYU's English ACE 2005 system description[J]. ACE, 2005, 5.
- [7] Ahn David. The stages of event extraction[C]//Proceedings of the Workshop on Annotating and Reasoning about Time and Events, 2006: 1-8.
- [8] Ji Heng, Grishman Ralph. Refining event extraction through cross-document inference[C]//Proceedings of ACL-08; HLT, 2008: 254-262.
- [9] Gupta Prashant, Ji Heng. Predicting unknown time arguments based on cross-event propagation[C]//Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, 2009: 369-372.
- [10] Grishman Ralph. The impact of task and corpus on event extraction systems[C]//Proceedings of LREC, 2010.
- [11] Hong Yu, Zhang Jianfeng, Ma Bin, et al. Using cross-entity inference to improve event extraction[C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2011: 1127-1136.
- [12] Liu Shulin, Chen Yubo, He Shizhu, et al. Leveraging framenet to improve automatic event detection[C]//Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, 2016, 1: 2134-2143.
- [13] Zeng Ying, Feng Yansong, Ma Rong, et al. Scale up event extraction learning via automatic training data generation[C]//Proceedings of the 32 AAAI Conference on Artificial Intelligence, 2018.
- [14] Nguyen Thien Huu, Grishman Ralph. Event detection and domain adaptation with convolutional neural networks[C]//Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, 2015, 2: 365-371.
- [15] Chen Yubo, Xu Liheng, Liu Kang, et al. Event extraction via dynamic multi-pooling convolutional neural networks[C]//Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, 2015, 1: 167-176.
- [16] Nguyen Thien Huu, Grishman Ralph. Modeling skip-grams for event detection with convolutional neural networks[C]//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016: 886-891.
- [17] Duan Shaoyang, He Ruifang, Zhao Wenli. Exploiting document level information to improve event detection via recurrent neural networks[C]//Proceedings of the 8th International Joint Conference on Natural Language Processing, 2017, 1: 352-361.
- [18] Feng Xiaocheng, Qin Bing, Liu Ting. A language-independent neural network for event detection[J]. Science China Information Sciences, 2018, 61(9): 092-106.
- [19] Chen Yubo, Yang Hang, Liu Kang, et al. Collective event detection via a hierarchical and bias tagging networks with gated multi-level attention mechanisms[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018: 1267-1276.
- [20] Zhao Yue, Jin Xiaolong, Wang Yuanzhuo, et al. Document embedding enhanced event detection with hierarchical and supervised attention[C]//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, 2018, 2: 414-419.
- [21] Hong Yu, Zhou Wenxuan, Zhang Jingli, et al. Self-regulation: Employing a generative adversarial network to improve event detection[C]//Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, 2018, 1: 515-526.
- [22] Liu Jian, Chen Yubo, Liu Kang, et al. Event detection via gated multilingual attention mechanism[C]//Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018.
- [23] Li Qi, Ji Heng, Huang Liang. Joint event extraction via structured prediction with global features[C]//Proceedings of the 51st Annual Meeting of the Asso-

- ciation for Computational Linguistics, 2013, 1: 73-82.
- [24] Sha Lei, Qian Feng, Chang Baobao, et al. Jointly extracting event triggers and arguments by dependency-bridge RNN and tensor-based argument interaction [C]//Proceedings of 32nd AAAI Conference on Artificial Intelligence, 2018.
- [25] Cao Yu, Meng Fang, Tao Dacheng. BAG: Bi-directional attention entity graph convolutional network for multi-hop reasoning question answering [J]. arXiv preprint arXiv: 1904.04969, 2019.
- [26] Guo Zhijiang, Zhang Yan, Lu Wei. Attention guided graph convolutional networks for relation extraction [J]. arXiv preprint arXiv: 1906.07510, 2019.
- [27] Doddington George R, Mitchell Alexis, Przybocki Mark A, et al. The automatic content extraction (ACE) program-tasks, data, and evaluation [C]//Proceedings of the 4th International Conference on Language Resources and Evaluation, 2004, 2: 1.
- [28] Mikolov Tomas, Chen Kai, Corrado Greg, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv: 1301.3781, 2013.
- [29] Hochreiter Sepp, Schmidhuber Jürgen. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735-1780.
- [30] Liao Shasha, Grishman Ralph. Using document level cross-event inference to improve event extraction [C]//Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010: 789-797.
- [31] Liu Shulin, Liu Kang, He Shizhu, et al. A probabilistic soft logic based approach to exploiting latent and global information in event classification [C]//Proceedings of the 30th AAAI Conference on Artificial Intelligence, 2016.
- [32] Liu Shulin, Chen Yubo, Liu Kang, et al. Exploiting argument information to improve event detection via supervised attention mechanisms [C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017:1789-1798.



陈佳丽(1994—), 硕士研究生, 主要研究领域为信息抽取。

E-mail: 20185227080@stu.suda.edu.cn



王捷(1997—), 硕士研究生, 主要研究领域为信息抽取。

E-mail: 2216525460@qq.com



洪宇(1978—), 通信作者, 博士, 教授, 主要研究领域为话题检测, 信息检索和信息抽取。

E-mail: tianxianer@gmail.com