

融合关系类型的图卷积网络关系抽取模型

赵良福

上海工程技术大学, 电子电气工程学院, 上海

收稿日期: 2023年9月12日; 录用日期: 2023年11月6日; 发布日期: 2023年11月13日

摘要

关系抽取是信息抽取和检索中的一项重要任务, 它旨在从运行的文本中提取给定实体之间的关系。以前的研究表明, 要想在这个任务中取得良好的表现, 需要对上下文信息进行良好的建模, 其中输入句子的依存树可以成为不同类型上下文信息中的一个有益来源。然而, 这些研究大多集中在词与词之间的依赖关系上, 对依赖类型的利用关注有限。现有的研究大多存在依存树中存在噪声的问题, 尤其是在自动生成依存树时, 大量利用依赖信息可能会给关系分类带来混乱, 因此对依存树进行必要的修剪非常重要。此外, 他们在建模时经常平等地对待不同的依赖连接, 因此在自动生成的依存树中会受到干扰(不准确的依赖解析)。该论文提出了一种用于关系提取的注意力机制的图卷积神经网络方法, 该方法将基于图卷积网络的注意力机制应用于依赖解析器获得的依存树中不同上下文单词, 以区分不同单词依赖的重要性。又加入一个新定义的模块, 将其命名为Key-Value Slot (简称KV Slot)。对于实体中的每个单词, KV Slot模块将所有关联的单词及其之间的依赖性进行映射, 然后根据对关系提取的贡献为其分配一个权重。该方法不仅利用了单词之间的依赖连接和类型, 而且还将可靠的依赖信息与嘈杂的信息区分开来, 在此基础上并对它们进行适当的建模。在SemEval2010-Task8和KBP37数据集上的实验证明了我们的方法的有效性, 模型在性能上有了较大提升。

关键词

图卷积神经网络, 依赖关系, 依存树, 依赖连接, 关系抽取

Relation Extraction Model Based on Graph Convolutional Network with Relations Types

Liangfu Zhao

School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai

Received: Sep. 12th, 2023; accepted: Nov. 6th, 2023; published: Nov. 13th, 2023

Abstract

Relation extraction is an important task in information extraction and retrieval, which aims to extract relations between given entities from running text. Previous studies have shown that good performance in this task requires good modeling of contextual information, where the dependency tree of the input sentence can be a beneficial source among different types of contextual information. However, most of these studies focus on word-to-word dependencies and pay limited attention to the exploitation of dependency types. Most of the existing studies have the problem of noise in the dependency tree, especially when the dependency tree is generated automatically, a large amount of dependency information may bring confusion to the relation classification, so it is very important to do necessary pruning of the dependency tree. Moreover, they often treat different dependency connections equally when modeling, and thus suffer from interference (inaccurate dependency parsing) in the automatically generated dependency tree. This paper proposes a graph convolutional neural network method for the attention mechanism of relation extraction, which applies the attention mechanism based on graph convolutional network to different context words in the dependency tree obtained by the dependency parser to distinguish the importance of different word dependencies. This paper adds a new module called the Key-Value Slot (KV Slot for short). For each word in an entity, the KV Slot module maps all associated words and dependencies between them, and then assigns it a weight based on its contribution to relation extraction. The proposed method not only exploits the dependency connections and types between words, but also distinguishes reliable dependency information from noisy information, builds on it and models them appropriately. Experiments on Semeval2010-task8 and KBP37 datasets prove the effectiveness of our method, and the performance of the model has been greatly improved.

Keywords

Graph Convolutional Neural Networks, Dependency Relationship, Dependency Tree, Dependency Connection, Relation Extraction

Copyright © 2023 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

关系抽取(RE)是信息提取和检索中最重要的任务之一,旨在识别出从原始文本中提及的实体间的关系,在支持文本挖掘等下游自然语言处理(NLP)应用,例如文本挖掘[1],情感分析[2],问答系统[3],和概要[4]等起着至关重要的作用。

最近,基于神经网络的RE方法[5][6][7][8][9]使用强大的编码器(如CNN,RNN和Transformers)显著提高了RE模型性能,而不需要任何精心设计的系统或手动构造的特征。这些方法在捕获上下文信息方面更胜一筹,从而使RE系统能够更好地理解文本并识别给定文本中实体之间的关系。采用神经网络的模型帮助RE不仅简单有效,而且有望将更多样化和更丰富的知识整合到RE系统中。在所有不同的知识来源中,句法信息,特别是依存树,已在许多研究中被研究人员证明是有益的[10][11][12][13],因为它们提供了有用单词之间的长距离单词连接,从而相应地指导系统更好地提取实体对之间的关系。

然而,大量利用依赖项信息并不总是能带来良好的RE性能,因为依赖项树中的噪声可能会给关系

分类带来混淆[6][14],特别是当这些树是自动生成的时候。因此,以往的研究在通过特定的模型(如 LSTM [6]或图卷积网络(GCN)) [11]对依赖信息进行编码之前,总是需要必要的修剪策略。因为固定的修剪策略不能保证得到包含所有重要上下文信息并过滤掉所有噪声的子树,所以有必要设计一种适当的方法来识别依存树中的噪声并相应地建模。

针对以上问题,本文所提出的基于注意力的改进图卷积神经网络关系抽取模型能够利用基于注意力机制的软剪枝策略来挖掘依存树中的有效信息,同时可以过滤掉无用信息;另外,本文加入 KV Slot 模块,对于实体中的每个单词, KV Slot 模块将所有关联的单词及其之间的依赖性映射到内存槽,然后根据每个槽对关系提取的贡献为其分配一个权重。本文的方法不仅利用了单词之间的依赖连接和类型,而且还将可靠的依赖信息与嘈杂的信息区分开来,并对它们进行适当的建模,让模型反复学习利用依赖信息和依赖类型。

2. 相关工作

依赖分析在 NLP 的语言表达与意义之间起着重要的中间作用。它已广泛应用于 NLP 任务,如情感分析[15][16],意见提取[17],知识推断[18][19]。对于 RE 任务,依存树可以提供具有更多语义关系特征 [6][20][21][22]的深度学习模型。然而,一个基本的问题是需要将多少依存树信息整合到基于神经网络的 RE 模型中才能获得令人满意的结果。未修剪的依存树对神经模型带来计算负担和不可避免的干扰,而过于激进的修剪策略会导致关系提取器学习不足。

图卷积网络(GCN)在依赖解析句的编码和并行计算中都显示出了它的有效性。Zhang 等人[11]在研究中首次将 GCN 引入 RE 任务。他们提出的 C-GCN 模型包括一个双向 LSTM 层和一个编码依赖结构的 GCN 层。作者采用了预定义的剪枝策略,其中当一个依存树节点与 SDP 的距离大于固定 k 时,将其从树中移除,从而从神经网络模型中排除。他们的模型在 $k=1$ 的情况下表现最好[11]。激发了一系列基于 GCN 的 RE 框架的后续工作。Zhou 等人[23]提出了一种改进的 C-GCN 模型在 GCN 操作中用距离加权的邻接矩阵代替常用的邻接矩阵,更好地表示树形结构,同时采用实体注意机制和剩余网络块等深度学习技术。为了摆脱那些预定义邻接矩阵带来的不灵活性,Guo 等人[24]和 Chen 等人[25]都使用了通过多头自注意机制学习到的动态矩阵,并采用其他深度学习技术,如密集连接层。但是,由于注意机制缺乏可解释性,他们的方法既没有关注 SDP 中的关键节点,也没有对依存树进行全面的处理。除了使用 C-GCN 框架进行句子级 RE 外, Sun 等人[26]和 Hong 等人[27]都将 GCN 应用于端到端提取,而 Sahu 等人[28]在跨句 RE 中使用了 GCN;这些应用证实了 GCN 在更一般的 RE 任务中的普遍适用性。

本文针对上述问题,第一,改进了用于关系抽取的注意力机制的图卷积神经网络方法(A-GCN),改进后的方法将基于图卷积网络的注意机制应用于依赖解析器获得的依存树中的不同上下文单词,以区分不同单词依赖的重要性。第二,在 A-GCN 基础上又加入一个新的模块,即 KV Slot 模块,对于实体中的每个单词, KV Slot 模块将所有关联的单词及其之间的依赖性映射到内存槽,然后根据每个槽对关系提取的贡献为其分配一个权重。增加的 KV Slot 模块不仅利用了单词之间的依赖连接和类型,而且还将可靠的依赖信息与嘈杂的信息区分开来,并对它们进行适当的建模。第三,本文将提出的模型在 SemEval2010-Task8 和 KBP37 数据集上进行实验,取得了很好的效果;为验证本文所提出的基于注意力机制的剪枝策略与 KV Slot 模块的有效性,还进行了消融实验。

3. 模型描述及实现

本章主要介绍基于改进注意力机制的图卷积神经网络,该模型框架如图 1 所示。下面将详细介绍各部分模块结构及功能。

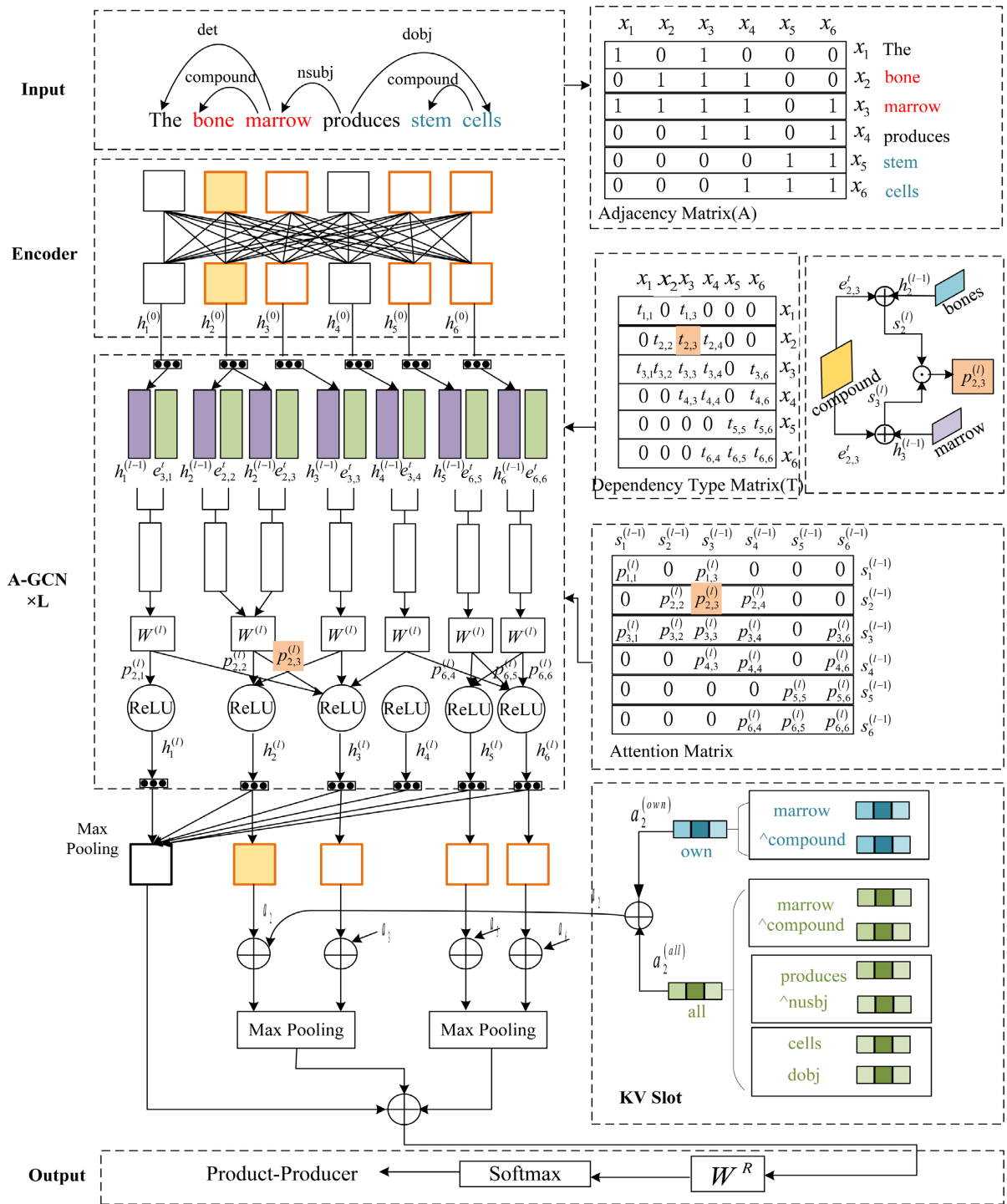


Figure 1. Improved A-GCN relational extraction model framework

图 1. 改进的 A-GCN 关系抽取模型框架

3.1. 模型总体框架介绍

本文用预训练模型 BERT 将输入句子进行编码，再使用一种改进的基于注意力的图卷积神经网络方法。具体来说，首先从现成的工具包中获得一个输入句子的依存树，然后在依存树上构建图，并对任意

两个词之间的不同标记的依赖连接赋予不同的权重，根据连接及其依赖类型计算权重。将改进的 A-GCN 网络的输出输入进新的 KV Slot 模块，增加的 KV Slot 模块不仅利用了单词之间的依赖连接和类型，而且还将可靠的依赖信息与嘈杂的信息区分开来，并对它们进行适当的建模。最后将输出结果进行 softmax 操作进行预测。

3.2. Transformer 和 BERT

如图 2 所示，本文使用的 BERT 预训练模型是以 Transformer 模型的编码器作为基模型的一种多头注意力机制构建的模型。在以往的研究中，BERT 和 RNN 等模型不同，在并行性方面、语义理解方面的表现更加突出。BERT 模型只是用了 Transformer 模型的编码器，而其编码器由 6 个相同的层组成，每一层又由两个子层组成，接下来本文对编码器子层的主要部分进行介绍，其结构如图 3 所示。

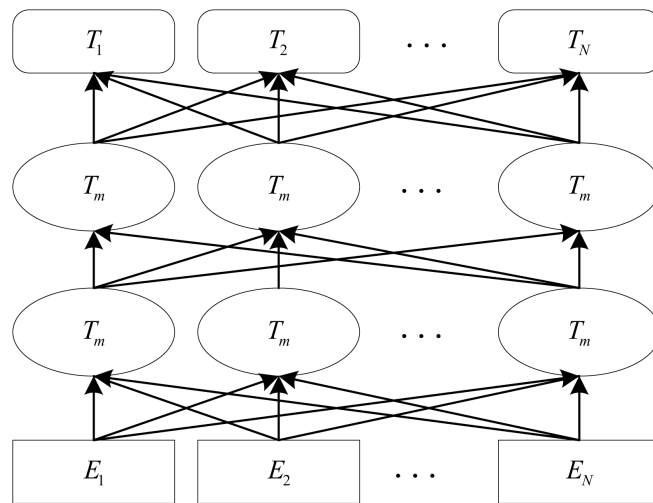


Figure 2. BERT model structure diagram
图 2. BERT 模型结构图

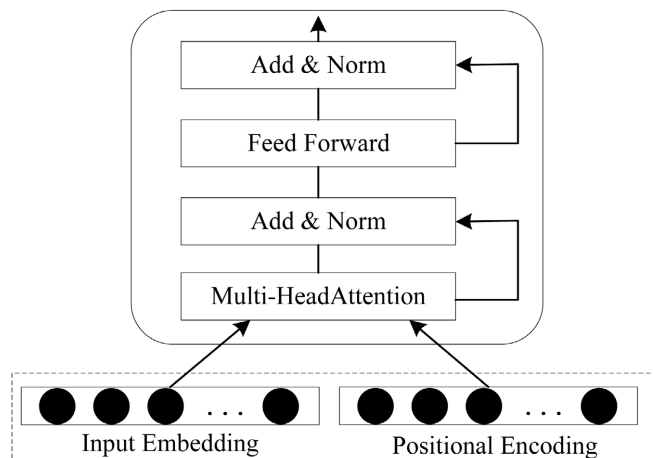


Figure 3. Transformer encoder structure diagram
图 3. Transformer 编码器结构图

Transformer 模型采用自注意力机制和多头注意力机制。设自注意力机制中查询向量 Q 、键向量 K 、值向量 V 的维度相同，注意力的计算采用了 scale dot-product。值向量 d_k 的权重为

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

多头注意力机制则是通过 h 个不同的线性变换对 Q, K, V 进行投影, 再将得到的自注意力中不同的注意力结果拼接起来。其表示为

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (2)$$

$$MultiHead(Q, K, V) = Concat(head_1, head_2, \dots, head_h) \quad (3)$$

自注意力层的结果会输入到前馈神经网络层中, 该层主要提供非线性变化, 每个解码器都有同样的层级结构。区别在于自注意力层和前馈神经网络层之间有一个注意力层。

3.3. 基于注意力机制的图卷积神经网络 A-GCN

为了增强文本表示, 从而获得对运行文本的良好理解, 许多研究[6] [25] [29] [30] [31]试图通过不同的模型架构利用上下文特性, 例如 **n-gram** 和语法信息。在这些结构选择中, 图卷积网络(GCN)是一种被广泛应用的对图中的信息进行编码的结构, 在每一层的 GCN 中, 每个节点的信息通过它们之间的连接与它的邻居进行通信。GCN 模型在输入句子图上编码上下文信息的有效性已被许多以前的研究证明。标准 GCN 模型中的图由单词依赖关系构建, 用邻接矩阵 $A = (a_{i,j})_{n \times n}$ 表示。如果 $i = j$ 或依存树中两个单词 x_i 和 x_j 之间存在依赖连接, $a_{i,j} = 1$; 否则 $a_{i,j} = 0$ 。对于句子中每一个单词 x_i , 第 l 层 GCN 层把其邻接单词集中在一起, 对于 x_i 计算它的输出表示 $h_i^{(l)}$:

$$h_i^{(l)} = \sigma\left(\sum_{j=1}^n a_{i,j} (W^{(l)} \cdot h_j^{(l-1)} + b^{(l)})\right) \quad (4)$$

其中公式里的 $h_j^{(l-1)}$ 表示 GCN 的 $(l-1)$ 层中 x_j 的输出表示, $W^{(l)}$ 和 $b^{(l)}$ 是 GCN 的 l 层可训练矩阵和偏置, σ 是 Relu 激活函数。

考虑到在标准 GCN 网络中, 单词之间的连接是平等看待的(即 $a_{i,j}$ 要么为 1, 要么为 0)。因此用于关系抽取的标准 GCN 模型不能识别不同连接的重要性。因此, 本文提出了一个用于关系抽取的改进的 A-GCN 模型, 这个模型使用注意力机制来计算不同连接的权重, 从而使模型相应利用不同依赖连接。除此之外, 标准 GCN 模型和大多数以前的研究都忽略了与依赖连接相关的依赖类型, 其中这些依赖类型对于关系抽取任务来说包含非常有用的信息, 本文将这些信息引入改进 A-GCN 模型。

具体来说, 本文将依存树中的依赖类型用一个类型矩阵 $T = (t_{i,j})_{n \times n}$ 表示, 其中 $t_{i,j}$ 是 x_i 和 x_j 之间有向依赖连接相关的依赖类型(比如 nsubj)。然后将每个依赖类型 $t_{i,j}$ 映射到它的嵌入 $e_{i,j}^t$ 。再之后, 在 GCN 的第 l 层, x_i 和 x_j 之间连接的权重可以计算为:

$$p_{i,j}^{(l)} = \frac{a_{i,j} \cdot \exp(s_i^{(l)} \cdot s_j^{(l)})}{\sum_{j=1}^n a_{i,j} \cdot \exp(s_i^{(l)} \cdot s_j^{(l)})} \quad (5)$$

其中 $a_{i,j} \in A$, “ \cdot ” 代表内积操作, $s_i^{(l)}$ 和 $s_j^{(l)}$ 分别是 x_i 和 x_j 的中间向量, 其计算公式如下:

$$s_i^{(l)} = h_i^{(l-1)} \oplus e_{i,i}^t \quad (6)$$

和

$$s_j^{(l)} = h_j^{(l-1)} \oplus e_{i,j}^t \quad (7)$$

其中 \oplus 表示向量连接操作。本文将权重 $p_{i,j}^{(l)}$ 作为 x_i 和 x_j 之间的关联依赖连接, 从而得到 x_i 的输出表示:

$$h_i^{(l)} = \sigma \left(\frac{\sum_{j=1}^n p_{i,j}^{(l)} \cdot W^{(l)} \cdot \tilde{h}^{(l-1)}}{d_i} + b^{(l)} \right) \quad (8)$$

其中 $\tilde{h}_j^{(l-1)}$ 是 x_j 的类型加强表示, d_i 为节点 i 的度, $\tilde{h}_j^{(l-1)}$ 的计算表示如下:

$$\tilde{h}_j^{(l-1)} = h_j^{(l-1)} + W_T^{(l)} \cdot e_{i,j}^t \quad (9)$$

其中 $W_T^{(l)}$ 将依赖类型嵌入 $e_{i,j}^t$ 映射到和 $h_j^{(l-1)}$ 相同的维度。

和标准 GCN 相比, 本文的方法用了数值加权(即 $p_{i,j}^{(l)} \in [0,1]$), 而不是对于 $a_{i,j}$ 的一个二元选择, 以此来区分不同连接的重要性, 从而可以更好利用。除此之外, 本文将依赖类型信息加入到计算的权重(即 $p_{i,j}^{(l)}$) 和 x_i 的输出表示($h_i^{(l)}$), 这些在以前的大多数研究中都没有考虑到。

3.4. KV Slot 模块

受到论文[13]的启发, 为了更好的利用依存树中的依赖信息, 本文增加一个新的模块(Key-Value Slot, 简称 KV Slot), 根据不同依赖关系对关系提取任务的贡献来确定它们的权重, 合并携带键(keys)和值(values)的依赖信息, 使最终的关系抽取任务效果提高。下面来具体介绍增加的新模块:

在解析树中, 输入句子中的每个单词都与它的 governor 和它的 dependents 用标记的依赖关系连接起来。对于输入句子中的任意两个单词, 它们之间只有一条连接路径。对于一个实体中的每个单词, 比如, 实体 E_u 中的 x_{i_u} (x_{i_u} 为实体中的一个单词, i_u 是 x_{i_u} 在输入句子中的索引 $u \in \{1,2\}$) 对于实体中每一个单词, 本文都考虑两种依赖信息, 即“单独项”(own)和“集合项”(all)的依赖信息。

单独项(own)通过依赖解析记录了和 x_{i_u} 相连的单词的信息。为了构造单独项, 首先从句子的依存树中找到 x_{i_u} 作为 governor 的所有 dependents。然后将 governor 和 dependents 作为键, 它们与 x_{i_u} 的依赖关系作为值。本文将 x_{i_u} 的第 j 个单独项信息记作 $s_{i_u,j}^{(own)} = (k_{i_u,j}^{(own)}, v_{i_u,j}^{(own)})$, 其中 $k_{i_u,j}^{(own)}$ 为与 x_{i_u} 有依赖连接的单词, $v_{i_u,j}^{(own)}$ 为它们之间的依赖关系类型。

集合项(all)的作用是沿着两个实体之间的依赖路径合并上下文信息。为了构建集合项, 对于 E_u 中的每一个 x_{i_u} (将另一个实体记作 $E_{\bar{u}}$), 首先找到 x_{i_u} 与 $E_{\bar{u}}$ 的最后一个单词之间的依赖路径。使用最后一个单词的原因是英语中的名词短语(实体总是名词短语)往往是中心语后置的。然后类似于构建单独项的过程, 提取该路径上的所有单词(包括 $E_{\bar{u}}$ 的最后一个单词)以及对应的依赖关系类型。最后, 将单词作为键, 将依赖关系类型作为值。本文将 x_{i_u} 的第 j 个集合项信息记作 $s_{i_u,j}^{(all)} = (k_{i_u,j}^{(all)}, v_{i_u,j}^{(all)})$, 其中 $k_{i_u,j}^{(all)}$ 为与 x_{i_u} 有依赖连接的单词, $v_{i_u,j}^{(all)}$ 为它们之间的依赖关系类型。

在实体 E_u 中的单词 x_{i_u} (其中 i_u 是 x_{i_u} 在输入句子中的索引, $u \in \{1,2\}$), 将其第 j 个单独项和集合项分别记为: $s_{i_u,j}^{(own)} = (k_{i_u,j}^{(own)}, v_{i_u,j}^{(own)})$ 和 $s_{i_u,j}^{(all)} = (k_{i_u,j}^{(all)}, v_{i_u,j}^{(all)})$ 。以单独项举例, 首先用两个矩阵将键 $k_{i_u,j}^{(own)}$ 和值 $v_{i_u,j}^{(own)}$ 映射到它们的嵌入, 分别用 $e_{i_u,j}^{k,(own)}$ 和 $e_{i_u,j}^{v,(own)}$ 表示。之后, 本文通过 $e_{i_u,j}^{k,(own)}$ 和 GCN 模型输出的 $h_{i_u}^{(l)}$ 来计算 $q_{i_u,j}$:

$$q_{i_u,j} = \frac{\exp(h_{i_u}^{(l)} \cdot e_{i_u,j}^{v,(own)})}{\sum_{j=1}^{n_{i_u}^{(own)}} \exp(h_{i_u}^{(l)} \cdot e_{i_u,j}^{k,(own)})} \quad (10)$$

其中 $n_{i_u}^{(own)}$ 是 x_{i_u} 的单独项中包含的项数, 然后通过以下公式得到键和值的加权和 $a_{i_u}^{(own)}$:

$$a_{i_u}^{(own)} = \sum_{j=1}^{n_u^{(own)}} q_{i_u,j} \left(e_{i_u,j}^{k,(own)} + e_{i_u,j}^{v,(own)} \right) \quad (11)$$

其中+代表向量的元素和。

集合项也和单独项计算过程类似，可得到一个加权和 $a_{i_u}^{(all)}$ ，继而得到 a_{i_u} ：

$$a_{i_u} = a_{i_u}^{(own)} + a_{i_u}^{(all)} \quad (12)$$

a_{i_u} 即为 KV Slot 对于单词 x_{i_u} 的输出。将实体 E_u 的每一个单词 x_{i_u} 都进行以上操作，最终通过最大池化操作得到实体 E_u 的向量化表示：

$$O_u = MaxPooling \left(h_{i_u}^{(l)} \oplus a_{i_u} \right) \quad (13)$$

因此本文得到两个实体的表示(O_1 代表 E_1 ， O_2 代表 E_2)。先通过 BERT 将输入句子 X 编码，将 x_i 编码记为 $h_i^{(0)}$ 。然后将 $h_i^{(0)}$ 作为输入，输入到 l 层改进的 A-GCN 模型，输出记为 $h_i^{(l)}$ 。本文通过以下公式得到整个输入句子向量表示 h_x ：

$$h_x = MaxPooling \left(\left\{ h_1^{(l)}, h_2^{(l)}, \dots, h_n^{(l)} \right\} \right) \quad (14)$$

最后用一个可训练矩阵 W_R 将其映射到输出空间：

$$O = W_R \cdot (h_x \oplus O_1 \oplus O_2) \quad (15)$$

其中 O 是一个 $|R|$ 维向量，每一个值代表关系类型集 R 中的一种关系类型，用 Softmax 函数来预测 E_1 和 E_2 之间的关系 \hat{r} ：

$$\hat{r} = \arg \max_{u=1}^{|R|} \frac{\exp(O^u)}{\sum_{u=1}^{|R|} \exp(O^u)} \quad (16)$$

其中 O^u 代表向量 O 在第 u 维的值。

4. 实验与结果分析

4.1. 实验数据与评估指标

本文在两个标准关系抽取数据集上进行实验，分别为：

1) SemEval2010-Task8 数据集。该数据集包含 10717 个句子实例，其包括 8000 个训练实例和 2717 个测试实例，关系类型包括 9 类关系和 1 个 other 类，关系类别及数据分布如表 1 所示。

2) KBP37 数据集。该数据集使用了 2010 年和 2013 年的 KBP 文档数据集，以及对 2013 年 Wikipedia 进行注释的文本数据集，包括训练实例 15,917 个，测试实例 3405 个，包含 19 种不同的关系，其中丢弃了低频关系，每种关系的训练实例超过 100 条。

在两个数据集上本文均采用官方评价指标宏观平均(Macro)F1 值评估模型。Macro 首先对每一个类计算 F1 值，然后对所有类求算术平均值。表 2 为关系分类结果的混淆矩阵。

在计算 F1 指标值之前，根据混淆矩阵求得查准率 P 和查全率 R ，计算如式(17)、(18)所示。

$$P = TP / (TP + FP) \times 100\% \quad (17)$$

$$R = TP / (TP + FN) \times 100\% \quad (18)$$

F1 值定义为查准率和查全率的调和平均数，如式(19)所示。

$$F_1 = 2PR / (P + R) \quad (19)$$

Table 1. SemEval2010-Task8 dataset relation type and corpus distribution
表 1. SemEval2010-Task8 数据集关系类型及语料分布

关系类型	关系描述	训练集		测试集	
		实例数	占比/%	实例数	占比/%
Cause-Effect(C-E)	因果关系	1003	12.54	328	12.07
Component-Whole(C-W)	整体与部分	941	11.76	312	11.48
Entity-Destination(E-D)	实体与目的	845	10.56	292	10.75
Product-Producer(P-P)	实体与来源	717	8.96	231	8.50
Entity-Origin(E-O)	产品与生产者	716	8.95	258	9.50
Member-Collection(M-C)	成员与组织	690	8.63	233	8.58
Message-Topic(M-T)	信息与主题	634	7.92	261	9.61
Content-Container(C-C)	内容与包含	540	6.75	192	7.07
Instrument-Agency(I-A)	工具与机构	504	6.30	156	5.74
Other(0)	其他	1410	17.63	454	16.71
	总计	8000	100	2717	100

Table 2. Classification result confusion matrix
表 2. 分类结果混淆矩阵

真实情况	预测结果	
	真(正例)	假(反例)
真	TP (真正例)	FN (假反例)
假	FP (假正例)	TN (真反例)

4.2. 依赖图的构建过程

为了构造 A-GCN 的图，本文使用 Standard CoreNLP Toolkits (SCT) 获取每个输入句子 X 的依存树 T_X 。虽然本文的方法能够通过注意机制区分不同依赖连接的重要性，但如果能够通过特定的修剪策略找出那些给关系抽取带来困惑的依赖连接，它仍然是有益的。受先前研究[11] [12] [15] [32] [33] [34] [35] 的启发，我们通过包含两组依赖连接，即本地连接(local connection)和全局连接(global connection)，来构建 A-GCN 网络模型的图。其中，本地连接包括直接连接两个实体头部的所有依赖项，全局连接包括两个实体头部之间的最短依赖路径(最短依赖路径 SDP)上的所有依赖项，通常情况下，不直接连接两个实体的单词也会被涉及。通过一个包含两个实体(即“company”和“restrictions”)的句子，图 4 展示了两组依赖连接以及用这两组连接构建的邻接矩阵。当 SDP 比较短，本地组的连接数可能比全局组的连接数多。

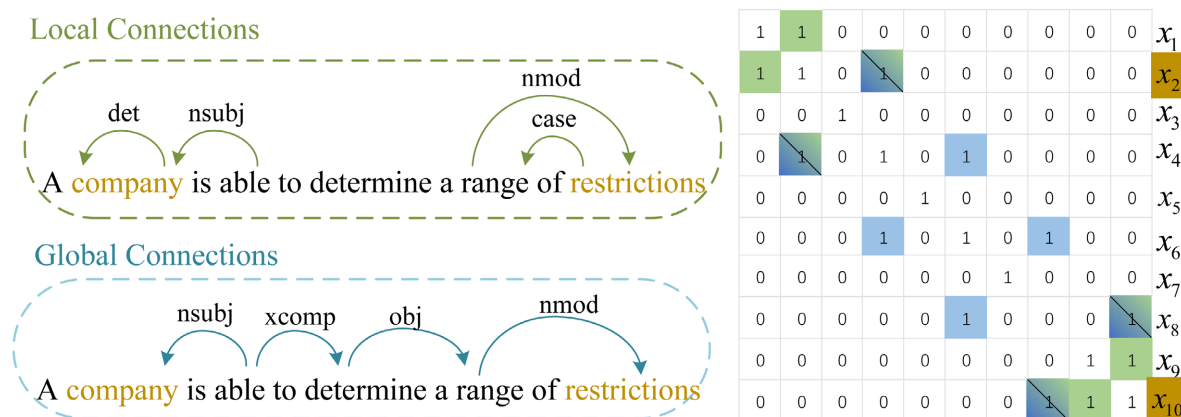


Figure 4. Local connection and global connection

图 4. 局部连接和全局连接

4.3. 参数设置

本文引入四个特殊 token (“<e1>”, “</e1>”, “<e2>”, “</e2>”)用来标记两个实体的边界,方便编码器在编码时能够区分实体的位置,从而提高模型性能。对于编码器,本文使用 BERT,因为它是一个强大的预训练语言模型,其变体在许多 NLP 任务中取得了最先进的性能。本文使用 BERT-base-uncased 和 BERT-large-uncased 编码器(对于 BERT-base,使用了带有 768 维隐藏向量的 12 层多头注意力机制;对于 BERT-large,使用 1024 维隐藏向量的 24 层多头注意力机制)。对于 A-GCN 和 KV Slot,本文随机初始化所有可训练参数和依赖类型嵌入,它们的维数与 BERT 中隐藏向量的维数匹配。对于训练模型需要设置的其他超参数(例如,学习率(learning rate)、预热学习率(warmup rate)、dropout rate 和批大小(batch_size)),将这些数据记录在表 3 中,在其中测试每个模型的所有组合,并使用最终实验中 F1 值达到最高的一个(它们的最佳组合以粗体显示)。

Table 3. Experiment parameter

表 3. 实验参数

超参数	值
Learning rate	5e-6, 1e-5, 2e-5, 3e-5
Warmup rate	0.06 , 0.1
Dropout rate	0.1
Batch Size	16, 32 , 64, 128

4.4. 总体结果

在本实验中,本文使用 BERT-base 和 BERT-large 编码器在带有或不带有依赖修剪策略的图上运行改进的 A-GCN 模型,对应于基于局部和全局连接(“L+ G”),以及由完全依赖图(“Full”)构造。本文还使用标准 GCN 和标准 GAT 运行基线。对于标准 GCN 和 A-GCN,本文尝试不同数量的层(即 1 到 3 层)。此外,本文将不使用任何依赖信息的 BERT-base 和 BERT-large 作为基线模型。表 4 显示了 A-GCN 模型在 SemEval 2010-Task8 和 KBP37 测试集上的 F1 值。

通过表 4 可以得出几个观察结果。首先,当使用 BERT-base 或 BERT-large 作为编码器时,可以观察到与仅使用 BERT 的基线(ID: 1)相比,即使 BERT 基线已经取得了良好的性能, A-GCN 在数据集上的功

能更加突出。其次，当有相同的图(即“L+G”或“Full”)和相同的层数时，A-GCN 优于 GAT (ID: 2,3) 和标准 GCN (ID: 4、6、8、10、12、14)。可以将这种观察结果解释为用于权衡不同依赖连接的注意力机制，该机制允许 A-GCN 区分图中的噪声，从而相应地利用有用的依赖信息。第三，在 A-GCN 层数不同的模型中，(比如 ID: 11 用于 BERT-base 和 ID: 11 用于 BERT-large)，两层 A-GCN 获得最高 F1 值。此外，发现本文的 A-GCN 模型(以及标准的 GCN 基线)具有本地和全局连接图(即“L+G”)始终优于具有完整依赖图(即“Full”)。这些观察结果是相对直观的，因为当依赖信息以密集的方式被利用时(例如，通过使用更多的层或不进行修剪的完整依存树)，它可能会给 RE 引入更多的噪声。

本文在 2 个 A-GCN 层输出后加入 KV Slot 模块，继续提高模型精度。选择加入或不加入 KV Slot 模块，若选择加入 KV Slot 模块，应用单独项和集合项依赖信息的不同组合(即仅单独项依赖信息，仅集合项依赖信息，以及两者都使用)。表 5 显示了改进的 A-GCN 模型加上 KV Slot 模块后在 SemEval 2010-Task8 和 KBP37 测试集上的最终 F1 值。

通过观察发现，用相同剪枝图的两个 A-GCN 层输出加入 KV Slot 模块后，当单独项(own)和集合项(all)依赖信息同时使用时，F1 值最高，即+KV Slot(both)。从而表明了单独项和集合项包含的依赖信息对于关系抽取任务的作用，本文将它们结合在一起以提高模型性能的方法是可行的。与仅使用集合项依赖信息的模型(即+KV Slot(all))相比，仅使用单独项依赖信息的模型(即+KV Slot(own))在大多数情况下获得了更好的结果。一种可能的解释如下：单独项依赖关系和集合项依赖关系之间存在重叠。如果只使用单独项依赖关系，模型不仅可以利用与实体本身直接关联的上下文信息，而且可以部分的利用依赖路径上的上下文信息，而具有集合项依赖关系的模型则不能利用与实体直接关联的上下文信息，这使得 KV Slot (own)可以实现更好的效果。

Table 4. F1 value of the improved A-GCN model

表 4. 改进的 A-GCN 模型的 F1 值

ID	Models	SemEval	KBP37	ID	Models	SemEval	KBP37
1	BERT-Base	87.87	68.30	1	BERT-Large	89.02	69.10
2	+GAT (Full)	88.39	68.62	2	+GAT (Full)	89.39	68.72
3	+GAT (L + G)	88.53	68.81	3	+GAT (L + G)	89.44	68.93
4	+1GCN Layer (FULL)	87.58	68.90	4	+1 GCN Layer (FULL)	88.98	68.96
5	+1A-GCN Layer (FULL)	88.36	69.12	5	+1 A-GCN Layer (FULL)	89.56	69.21
6	+1 GCN Layer (L + G)	88.64	69.15	6	+1 GCN Layer (L + G)	89.11	69.34
7	+1 A-GCN Layer (L + G)	89.05	69.23	7	+1 A-GCN Layer (L + G)	89.69	69.42
8	+2 GCN Layer (FULL)	88.66	69.15	8	+2 GCN Layer (FULL)	89.43	69.62
9	+2 A-GCN Layer (FULL)	88.72	69.30	9	+2 A-GCN Layer (FULL)	89.71	70.10
10	+2 GCN Layer (L + G)	88.62	70.21	10	+2 GCN Layer (L + G)	89.42	70.53
11	+2 A-GCN Layer (L + G)	89.18	70.62	11	+2 A-GCN Layer (L + G)	89.86	70.72
12	+3 GCN Layer (FULL)	88.54	69.34	12	+3 GCN Layer (FULL)	89.62	69.76
13	+3 A-GCN Layer (FULL)	88.65	69.76	13	+3 A-GCN Layer (FULL)	89.46	69.90
14	+3 GCN Layer (L + G)	88.33	70.12	14	+3 GCN Layer (L + G)	89.19	70.24
15	+3 A-GCN Layer (L + G)	88.72	70.23	15	+3 A-GCN Layer (L + G)	89.57	70.32

Table 5. Comparison of F1 values of the model after adding KV Slot module
表 5. 入 KV Slot 模块后模型的 F1 值对比

ID	Models	SemEval	KBP37
1	BERT-Base	87.87	68.30
2	+2 A-GCN Layer (FULL)	88.72	69.30
3	+2 A-GCN Layer (FULL) + KV Slot (own)	88.96	69.97
4	+2 A-GCN Layer (FULL) + KV Slot (all)	88.82	69.72
5	+2 A-GCN Layer (FULL) + KV Slot (both)	89.72	70.02
6	+2 A-GCN Layer (L + G)	89.18	70.62
7	+2 A-GCN Layer (L + G) + KV Slot (own)	89.96	70.84
8	+2 A-GCN Layer (L + G) + KV Slot (all)	89.34	70.71
9	+2 A-GCN Layer (L + G) + KV Slot (both)	90.12	70.93
ID	Models	SemEval	KBP37
1	BERT-Large	89.02	69.10
2	+2 A-GCN Layer (FULL)	89.71	70.10
3	+2 A-GCN Layer (FULL) + KV Slot (own)	90.12	70.42
4	+2 A-GCN Layer (FULL) + KV Slot (all)	89.94	70.25
5	+2 A-GCN Layer (FULL) + KV Slot (both)	90.32	70.54
6	+2 A-GCN Layer (L + G)	89.86	70.72
7	+2 A-GCN Layer (L + G) + KV Slot (own)	90.34	70.92
8	+2 A-GCN Layer (L + G) + KV Slot (all)	90.14	70.85
9	+2 A-GCN Layer (L + G) + KV Slot (both)	90.67	70.98

4.5. 现有方法

本文最好的模型，使用 BERT-large 编码器和两个 A-GCN 层 + KV Slot 模块，将其与以前的研究作对比，结果记录在表 6 中，模型介绍如下：

- 1) CNN + PF (CNN Position Feature) [36]: 模型为基本的 CNN，引入实体位置特征。
- 2) RNN + PF [37]: 将 CNN + PF 中的 CNN 替换为基本的 RNN。
- 3) SDP-LSTM [6]: 通过剪枝策略，选取依存树中的最短依赖路径作为输入，利用 LSTM 提取异构信息。
- 4) SPTree (Shortest Path Tree) [10]: 应用剪枝策略，将整棵树缩减为实体最低公共祖先下的子树，并利用双向树状结构的 LSTM 捕获句子高阶特征。
- 5) Att-Bi-LSTM (Attention Bi-LSTM) [38]: 利用注意力机制作用于 LSTM 输出层捕获句子中重要的语义特征。
- 6) SA-Bi-LSTM-LET [39]: 将实体感知注意力机制与潜在实体类型相结合，充分利用实体信息进行关系抽取。
- 7) A-GCN [40]: 使用注意力机制来计算不同连接的权重，使模型利用不同依赖连接。

从表 6 可以看出，本文模型的 F1 值均高于对比模型。本文模型增加的 KV Slot 模块不仅利用了单词

之间的依赖连接和类型，而且还将可靠的依赖信息与嘈杂的信息区分开来，并对它们进行适当的建模，从而进一步提高了关系抽取任务的性能。

Table 6. F1 values for the best model and previous models are used
表 6. 使用最佳模型和以往模型的 F1 值

Models	SemEval	KBP37
CNN + PF	82.7	51.3
RNN + PF	82.5	54.3
SDP-LSTM	83.7	58.3
SPTree	84.4	59.1
Att-Bi-LSTM	84.0	-
SA-Bi-LSTM-LET	85.1	59.0
A-GCN (FULL)	89.70	68.7
A-GCN (L + G)	89.85	69.2
A-GCN (FULL) + KV Slot	90.32	70.54
A-GCN (L + G) + KV Slot	90.67	70.98

4.6. 改进 A-GCN 的作用

在关系抽取中依赖信息是有益的，因为它包含长距离的单词 - 单词关系，当给定的两个实体在输入句子中相距很远时，这可能非常有用。为了探索 A-GCN 和新增的 KV Slot 模块在捕获这种长距离单词关系上帮助关系抽取任务的效果，本文根据实体的距离(即两个实体之间的单词数量)将测试实例分成不同的组，并在这些组上运行模型以测试它们的性能。图 5 展示了在 SemEval2010-Task8 测试集中的三组测试实例上，性能最好的 A-GCN + KV Slot 模型使用 BERT-large 及其相应的标准 GCN 和 BERT-large 基线的性能，其中横坐标表示两实体距离的范围，两实体之间距离用两实体之间词的数量表示。可以观察到，A-GCN + KV Slot 在所有组测试实例上的性能都优于两个基线。这个观察结果证实了本文的方法能够利用依赖信息并捕获长距离的单词 - 单词关系来改进关系抽取任务。并且 A-GCN + KV Slot 模型同样优于 A-GCN 模型，表明本文新增的 KV Slot 模块编码依赖信息以改进关系抽取的方法的有效性。

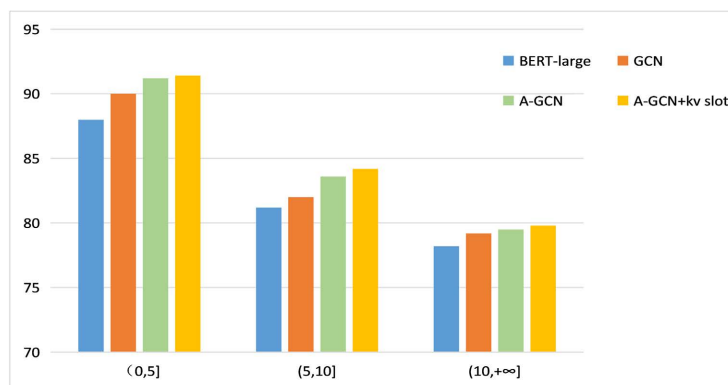


Figure 5. In the instance Test model performance based on the distance (i.e, number of words) between two entities
图 5. 根据实例中两个实体之间的距离(即单词数)测试模型性能

4.7. 图构建的影响

本文使用带有本地连接和全局连接(“L + G”)的 A-GCN 模型。为了研究本地连接和全局连接对 A-GCN 的影响,本文使用两个 A-GCN 层和由本地连接(“L”)或全局连接(“G”)构建的图来运行方法。表 7 为 BERT-base 和 BERT-large 编码器下不同模型的实验结果(F1 值)。和 A-GCN (L + G)相比,由局部连接(即 A-GCN (L))或全局连接(即 A-GCN (G))构建的图模型的性能较低,这符合预期,因为两组连接都包含重要的用于关系抽取的上下文特征。除此之外还发现不论用 BERT-base 还是 BERT-large 编码器, A-GCN (L)都比 A-GCN (G)效果更好。一个比较合理的解释是:本地连接和全局连接之间存在重叠(例如,图 4 中的“range”和“restrictions”之间的连接)。因此, A-GCN (L)不仅可以利用与实体本身相关的上下文信息,而且部分得益于两个实体在 SDP 上的重叠连接,使得 A-GCN (L)比 A-GCN (G)具有更好的效果。

Table 7. F1 values for models built with graphs of different styles

表 7. 使用不同方式的图构建的模型的 F1 值

ID	Models	SemEval	KBP37
1	BERT-base	87.87	68.30
2	+A-GCN (L)	88.89	69.26
3	+A-GCN (G)	88.86	68.86
4	+A-GCN (L + G)	89.16	70.62
5	+A-GCN (Full)	88.70	69.30
6	BERT-large	89.02	69.10
7	+A-GCN (L)	89.70	69.98
8	+A-GCN (G)	89.38	69.54
9	+A-GCN (L + G)	89.85	70.72
10	+A-GCN (Full)	89.70	70.10

4.8. KV Slot 模块的影响

之前的实验证明了本文的模型与 KV Slot 在一阶词相关性上的有效性。本文还使用 KV Slot (all)通过 BERT-large 实现二阶和三阶依赖。表 8 记录了实验结果(包括来自一阶依赖项的分数)。分析结果如下:首先,在所有设置下使用 KV Slot 的模型都优于只使用 A-GCN 的模型。其次,模型在相同的设置下, KV Slot (own)的性能始终优于 KV Slot (all),这表明单独项依赖能够给模型带来更大的改进。第三,在所有三个设置下合并不同类型的依赖关系信息(即单独项(own)、集合项(all)或两者都有(both)),本文的新模型优于 BERT 基线,并且当使用单独项和集合项依赖信息时, F1 得分最高(即+ KV Slot (both))。这一观察结果证实了单独项和集合项依赖信息的贡献,以及本文的方法的有效性。第四,对于 KV Slot (both),使用更高阶的依赖关系通常会导致较差的结果;而 KV Slot (own)则正好相反。一种可能的解释是,对于 KV Slot (both),两个实体之间的大多数基本单词依赖关系已经编码,更高阶的依赖关系有时会引入噪声而不是有用的信息;而对于 KV Slot (own),利用更高阶依赖关系允许模型沿两个实体之间的依赖路径覆盖更多上下文信息。

Table 8. F1 values measured using dependencies of different orders
表 8. 利用不同阶数依赖项测得的 F1 值

Models	Order	SemEval	KBP37
A-GCN	N/A	89.86	70.72
+KV Slot (own)	1st	89.92	70.76
	2nd	89.87	70.75
	3rd	89.94	70.78
+KV Slot (both)	1st	90.08	70.96
	2nd	89.96	70.87
	3rd	89.93	70.79

4.9. 消融实验

与标准的 GCN 相比, A-GCN 从两个方面进行了改进: (1) 使用注意机制对不同的依赖连接进行权衡; (2) 在过程中引入依赖类型, 对更详细的依赖信息进行编码。为了更好地研究每个个体增强(即注意机制或依赖类型信息)的影响, 本文对最佳模型(即带有 BERT-base 和 BERT-large 编码器的双层 A-GCN (L + G)) 进行了消融研究。表 9 记录了不同模型的实验结果, 其中仅使用 BERT 基准和标准 GCN 基准(不使用注意机制和依赖类型)的性能也记录下来作为参考。结果表明, 弱化增强信息(即注意机制或依赖类型信息)会导致更差的结果。在这两种增强中, 注意力机制对 A-GCN 的影响更大, 这表明 A-GCN 区分重要连接并相应地利用它们的能力在关系抽取中起更重要的作用。如表 10 所示, 在 A-GCN 模型基础上, 增加 KV Slot 模块之后, 显然使得在 SemEval2010-Task8 和 KBP37 数据集上 F1 值增加, 效果更好, 对原有模型性能上有提升, 所以加入 KV Slot 模块。

Table 9. Results of ablation of attentional mechanisms (ATT.) and dependent types (TYPE) in the A-GCN model
表 9. 在 A-GCN 模型中注意力机制(ATT.)和依赖类型(TYPE)的消融研究结果

	ATT.	TYPE	SemEval	KBP37
BERT-base		Baseline	87.87	68.30
	✓	✓	89.16	69.87
	×	✓	88.07	68.96
	✓	×	88.50	69.42
		GCN	88.62	70.21
BERT-large		Baseline	89.02	69.10
	✓	✓	89.85	70.83
	×	✓	89.26	69.98
	✓	×	89.37	70.12
		GCN	89.13	69.84

Table 10. On the basis of A-GCN model, the ablation results of whether to add KV Slot module are carried out
表 10. 在 A-GCN 模型基础上对于是否加入 KV Slot 模块进行消融研究结果

Models	加入 KV Slot 模块	SemEval	KBP37
A-GCN (BERT) (FULL)	√	90.10	70.72
	×	89.70	70.10
A-GCN (BERT) (L + G)	√	90.15	70.96
	×	89.85	70.83

5. 结束语

在本文中，提出了利用依赖信息进行关系抽取的 A-GCN + KV Slot 方法，该方法通过对依赖连接应用注意力机制，对连接和类型同时施加重权，从而更好地进行区分重要的依赖项信息，并相应地利用它们。通过这样做，A-GCN 能够动态地从不同的依赖连接中学习，从而修剪信息较少的依赖。KV Slot 的键和值建立在从现成工具包中获取的输入句子的依存树上。特别是对于句子中的每个实体，根据输入句子的依赖解析及其对应的依赖关系类型，提取与之相关的单词。然后，对这些信息进行编码和加权，并将其用于关系提取任务中。在数据集上的实验结果和分析表明，本文的方法是有效的，在两个英文数据集上都表现出了很好的性能。本文模型是在两个英文数据集上训练测试的，在之后的研究中，我们将进一步将模型扩展到中文数据集上。

参考文献

- [1] Distiawan, B., Weikum, G., Qi, J.Z. and Zhang, R. (2019) Neural Relation Extraction for Knowledge Base Enrichment. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, July 2019, 229-240.
- [2] Sun, K., Zhang, R., Mensah, S., Mao, Y.Y. and Liu, X.D. (2019) Aspect-Level Sentiment Analysis via Convolution over Dependency Tree. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, November 2019, 5683-5692. <https://doi.org/10.18653/v1/D19-1569>
- [3] Xu, K., Reddy, S., Feng, Y.S., Huang, S.F. and Zhao, D.Y. (2016) Question Answering on Freebase via Relation Extraction and Textual Evidence. *Proceedings of the Association for Computational Linguistics*, Berlin, August 2016, 2326-2336. <https://doi.org/10.18653/v1/P16-1220>
- [4] Wang, L. and Cardie, C. (2012) Focused Meeting Summarization via Unsupervised Relation Extraction. *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, Seoul, July 2012, 304-313.
- [5] Zeng, D.J., Liu, K., Lai, S.W., Zhou, G.Y. and Zhao, J. (2014) Relation Classification via Convolutional Deep Neural Network. *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, Dublin, August 2014, 2335-2344.
- [6] Xu, Y., Mou, L.L., Li, G., Chen, Y.C., Peng, H. and Jin, Z. (2015) Classifying Relations via Long Short Term Memory Networks along Shortest Dependency Paths. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, September 2015, 1785-1794. <https://doi.org/10.18653/v1/D15-1206>
- [7] Zhang, D.X. and Wang, D. (2015) Relation Classification via Recurrent Neural Network. arXiv: 1508.01006.
- [8] Wang, L.L., Cao, Z., De Melo, G. and Liu, Z.Y. (2016) Relation Classification via Multi-Level Attention CNNs. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, August 2016, 1298-1307. <https://doi.org/10.18653/v1/P16-1123>
- [9] Zhang, Y.H., Zhong, V., Chen, D.Q., Angeli, G. and Manning, C.D. (2017) Positionaware Attention and Supervised Data Improve Slot Filling. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, September 2017, 35-45. <https://doi.org/10.18653/v1/D17-1004>
- [10] Miwa, M. and Bansal, M. (2016) End-to-End Relation Extraction Using LSTMs on Sequences and Tree Structures. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*,

- Berlin, August 2016, 1105-1116. <https://doi.org/10.18653/v1/P16-1105>
- [11] Zhang, Y.H., Qi, P. and Manning, C.D. (2018) Graph Convolution over Pruned Dependency Trees Improves Relation Extraction. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, October-November 2018, 2205-2215. <https://doi.org/10.18653/v1/D18-1244>
- [12] Sun, K., Zhang, R.C., Mao, Y.Y., Mensah, S. and Liu, X.D. (2020) Relation Extraction with Convolutional Network over Learnable SyntaxTransport Graph. *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**, 8928-8935. <https://doi.org/10.1609/aaai.v34i05.6423>
- [13] Chen, G.M., Tian, Y.H., Song, Y. and Wan, X. (2021) Relation Extraction with Type-aware Map Memories of Word Dependencies. In: Zong, C.Q., Xia, F., Li, W.J. and Navigli, R., Eds., *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021*, Association for Computational Linguistics, Ohio, 2501-2512. <https://doi.org/10.18653/v1/2021.findings-acl.221>
- [14] Yu, B.W., Xue, M.G., Zhang, Z.Y., Liu, T.W., Wang, Y.B. and Wang, B. (2020) Learning to Prune Dependency Trees with Rethinking for Neural Relation Extraction. *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona, 8-13 December 2020, 3842-3852.
- [15] Nasukawa, T. and Yi, J. (2003) 'Sentiment Analysis: Capturing Favorability Using Natural Language Processing. *Proceedings of the 2nd International Conference on Knowledge Capture*, Sanibel Island, 23-25 October 2003, 70-77. <https://doi.org/10.1145/945645.945658>
- [16] Poria, S., Cambria, E., Winterstein, G. and Huang, G.B. (2014) 'Sentic Patterns: Dependency-Based Rules for Concept-Level Sentiment Analysis. *Knowledge-Based Systems*, **69**, 45-63. <https://doi.org/10.1016/j.knsys.2014.05.005>
- [17] Yang, B. and Cardie, C. (2013) Joint Inference for Fifine-Grained Opinion Extraction. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, Sofia, August 2013, 1640-1649.
- [18] Niu, F., Zhang, C., Ré, C. and Shavlik, J.W. (2012) DeepDive: Web-Scale Knowledge-Base Construction Using Statistical Learning and Inference. *VLDS'12*, Istanbul, 31 August 2012, 25-28.
- [19] Toutanova, K. and Chen, D. (2015) Observed versus Latent Features for Knowledge Base and Text Inference. *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, Beijing, July 2015, 57-66. <https://doi.org/10.18653/v1/W15-4007>
- [20] Xu, K., Feng, Y., Huang, S. and Zhao, D. (2015) Semantic Relation Classification via Convolutional Neural Networks with Simple Negative Sampling. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, September 2015, 536-540. <https://doi.org/10.18653/v1/D15-1062>
- [21] Cai, R., Zhang, X. and Wang, H. (2016) Bidirectional Recurrent Convolutional Neural Network for Relation Classification. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, August 2016, 756-765. <https://doi.org/10.18653/v1/P16-1072>
- [22] Ren, F., Zhou, D., Liu, Z., Li, Y., Zhao, R., Liu, Y. and Liang, X. (2018) Neural Relation Classification with Text Descriptions. *Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe, August 2018, 1167-1177.
- [23] Zhou, L., Wang, T., Qu, H., Huang, L. and Liu, Y. (2020) A Weighted GCN with Logical Adjacency Matrix for Relation Extraction. *Proceedings of the 24th European Conference on Artificial Intelligence*, Santiago de Compostela, August/September 2020, 2314-2321.
- [24] Guo, Z., Zhang, Y. and Lu, W. (2019) Attention Guided Graph Convolutional Networks for Relation Extraction. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, July 2019, 241-251. <https://doi.org/10.18653/v1/P19-1024>
- [25] Chen, H., Liu, L., Zhou, X., Qing, L. and Wang, M. (2020) A Robust Graph Convolutional Network for Relation Extraction by Combining Edge Information. *2020 IEEE 5th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*, Chengdu, 10-13 April 2020, 150-156. <https://doi.org/10.1109/ICCCBDA49378.2020.9095628>
- [26] Sun, C., Gong, Y., Wu, Y., Gong, M., Jiang, D., Lan, M., Sun, S. and Duan, N. (2019) Joint Type Inference on Entities and Relations via Graph Convolutional Networks. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, July 2019, 1361-1370. <https://doi.org/10.18653/v1/P19-1131>
- [27] Hong, Y., Liu, Y., Yang, S., Zhang, K., Wen, A. and Hu, J. (2020) Improving Graph Convolutional Networks Based on Relation-Aware Attention for End-to-End Relation Extraction. *IEEE Access*, **8**, 51315-51323. <https://doi.org/10.1109/ACCESS.2020.2980859>
- [28] Sahu, S.K., Christopoulou, F., Miwa, M. and Ananiadou, S. (2019) Inter-Sentence Relation Extraction with Document-Level Graph Convolutional Neural Network. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, July 2019, 4309-4316. <https://doi.org/10.18653/v1/P19-1423>
- [29] Song, Y. and Xia, F. (2013) A Common Case of Jekyll and Hyde: The Synergistic Effect of Using Divided Source Training Data for Feature Augmentation. *Proceedings of the Sixth International Joint Conference on Natural Lan-*

-
- guage Processing*, Nagoya, 14-19 October 2013, 623-631.
- [30] Mandya, A., Bollegala, D. and Coenen, F. (2020) Graph Convolution over Multiple Dependency Sub-Graphs for Relation Extraction. *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona, December 2020, 6424-6435. <https://doi.org/10.18653/v1/2020.coling-main.565>
- [31] Nie, Y.Y., Tian, Y.H., Song, Y., Ao, X. and Wan, X. (2020) Improving Named Entity Recognition with Attentive Ensemble of Syntactic Information. In: Cohn, T., He, Y.L. and Liu, Y., Eds., *Findings of the Association for Computational Linguistics: EMNLP 2020*, Association for Computational Linguistics, Dublin, 4231-4245.
- [32] Guo, Z.J., Zhang, Y. and Lu, W. (2019) Attention Guided Graph Convolutional Networks for Relation Extraction. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, July 2019, 241-251.
- [33] Chen, G.M., Tian, Y.H. and Song, Y. (2020) Joint Aspect Extraction and Sentiment Analysis with Directional Graph Convolutional Networks. *Proceedings of the 28th International Conference on Computational Linguistics*, Barcelona, December 2020, 272-279. <https://doi.org/10.18653/v1/2020.coling-main.24>
- [34] Tian, Y.H., Song, Y. and Xia, F. (2020) Supertagging Combinatory Categorical Grammar with Attentive Graph Convolutional Networks. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, November 2020, 6037-6044. <https://doi.org/10.18653/v1/2020.emnlp-main.487>
- [35] Tian, Y.H., Chen, G.M. and Song, Y. (2021) Aspect-Based Sentiment Analysis with Type-aware Graph Convolutional Networks and Layer Ensemble. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, June 2021, 2910-2922. <https://doi.org/10.18653/v1/2021.naacl-main.231>
- [36] Zeng, D., Liu, K., Lai, S., *et al.* (2014) Relation Classification via Convolutional Deep Neural Network. *Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers*, Dublin, August 2014, 2335-2344.
- [37] Zhang, D. and Wang, D. (2015) Relation Classification via Recurrent Neural Network. <https://arxiv.org/pdf/1508.01006.pdf>
- [38] Zhou, P., Shi, W., Tian, J., *et al.* (2016) Attention-Based Bidirectional Long Short-Term Memory Network for Relation Classification. *Proceeding of 54th Annual Meeting of the Association for Computational Linguistics*, Berlin, August 2016, 207-212. <https://doi.org/10.18653/v1/P16-2034>
- [39] Soares, L.B., FitzGerald, N., Ling, J. and Kwiatkowski, T. (2019) Matching the Blanks: Distributional Similarity for Relation Learning. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, July 2019, 2895-2905. <https://doi.org/10.18653/v1/P19-1279>
- [40] Tian, Y.H., Chen, G.M., Song, Y., *et al.* (2021) Dependency-Driven Relation Extraction with Attentive Graph Convolutional Networks. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 4458-4471.