基于动态句法剪枝机制的中文语义角色标注

费 豪[1)] 姬东鸿[1)] 任亚峰[2)]

[1)](武汉大学国家网络安全学院 武汉 430072)

[2)](广东外语外贸大学高级翻译学院 广州 510420)

摘 要 语义角色标注(Semantic Role Labeling,SRL)旨在识别给定句子中所包含的谓词及对应的语义论元,从而为信息抽取、自动问答和阅读理解等任务的语义理解提供帮助.构建句法特征作为实现语义角色标注任务的关键步骤,在很大程度上影响着任务的性能.针对现有的神经网络模型未能有效构建句法特征,例如现有研究采取离线式的人工定式句法裁剪方案,不可避免地造成关键句法信息丢失或者裁剪效果减弱等问题,本文提出基于动态句法剪枝机制的端到端神经网络模型,并将其用于中文语义角色标注任务.具体地,我们提出两种创新的动态句法剪枝机制:基于递归神经网络模型的动态句法剪枝机制(Recur-DSP)和基于带句法标签的图卷积网络模型的句法剪枝机制(SGCN-DSP).Recur-DSP 采用递归神经网络模型进行句法结构编码与融合,并对句法树的每一个连接处通过 Gumbel-Softmax 函数离散化实现动态句法裁剪.SGCN-DSP 采用图卷积神经网络模型为句法依存树的依存弧结构以及对应的标签进行统一建模,并提出对应的动态句法裁剪机制.在基准数据集上的实验结果显示所提方法超过当前的最好模型,获得当前中文语义角色标注的最优性能.通过整合预训练语言模型 BERT,基于 CoNLL09 数据集,提出的模型 SGCN-DSP 在角色论元识别上获得了 90.4%的 $F1$ 值,在谓词识别上获得 90.8%的 $F1$ 值.

关键词 自然语言处理;语义角色标注;句法剪枝;神经网络;深度学习

中图法分类号 TP391 **DOI 号** 10.11897/SP.J.1016.2022.01746

Chinese Semantic Role Labeling Based on Dynamic Syntax Pruning

FEI Hao[1)] JI Dong-Hong[1)] REN Ya-Feng[2)]

[1)](*School of Cyber Science and Engineering, Wuhan University, Wuhan* 430072)

[2)](*School of Interpreting and Translation Studies, Guangdong University of Foreign Studies, Guangzhou* 510420)

Abstract Semantic role labeling (SRL), as the shallow semantic parsing task, which has received extensive research attention in recent years and plays a core role in the natural language processing (NLP) community. The SRL task aims to identify the corresponding argument roles for the predicates of a given sentence, which can facilitate the downstream NLP tasks, such as information extraction, question answer system and reading comprehension, etc. A great number of methods have been proposed for the task, and the existing studies can be divided into two main categories: machine learning based methods with hand-crafted discrete features and deep learning methods with automatic distributed features. The early studies largely separate SRL into two individual subtasks, i.e., predicate disambiguation and argument role labeling. More recently, great efforts have been paid for constructing various end-to-end SRL architectures, solving two pipeline steps in one shot via one unified model. Recent studies also show that integrating external syntactic features, such as syntactic dependency trees, are important for the SRL task highly. So

收稿日期:2021-03-07;在线发布日期:2022-01-25.本课题得到国家重点研发计划项目(2017YFC1200500)、国家自然科学基金项目(61702121,61772378)和广州市科技计划项目(202102020607)资助.**费 豪**,博士研究生,中国计算机学会(CCF)会员,主要研究方向为自然语言处理和句法语义结构解析.E-mail:hao.fei@whu.edu.cn.**姬东鸿**,博士,教授,主要研究领域为自然语言处理、机器学习和数据挖掘.**任亚峰**(通信作者),博士,副教授,主要研究方向为自然语言处理和数据挖掘.E-mail:renyafeng@whu.edu.cn.

designing a novel neural model, which can capture syntactic features effectively, has become a heated research topic. Recently, He et al. (2018) find that only a part of syntactic structure information can offer valuable information for the SRL task, which calls for pruning the syntactic structure features. However, the existing work adopts the offline syntactic pruning strategy, which can inevitably lead to either the loss of key syntactic information or the weakening of pruning effectiveness. Extracting syntactic features, as an important step of the SRL task, will largely affect the final performance of the task. However, the existing neural network methods fail to effectively model syntactic features. For example, the existing studies adopt the offline syntactic pruning strategy with fixed human labor, which inevitably leads to the loss of key syntactic information or the weakening of pruning effectiveness. To address the above issues, we propose an end-to-end neural network model for the Chinese SRL task based on dynamic syntactic pruning mechanism. Specifically, we propose two novel methods: recursive neural network model with dynamic syntactic pruning (Recur-DSP) and syntax-label graph convolutional network with dynamic syntactic pruning (SGCN-DSP). Recur-DSP uses a recursive neural network model to encode and fuse syntactic structure knowledge, and applies the Gumbel-Softmax function to realize dynamic syntactic pruning. SGCN-DSP exploits a graph convolutional neural network model that can simultaneously encode syntactic arcs and labels, based on which we introduce the corresponding dynamic syntactic pruning strategy. Experimental results on multiple benchmark datasets show the effectiveness of the proposed methods. Our proposed methods outperform the current best method by a large margin, giving the state-of-the-art performances for the Chinese SRL task. Specifically, our proposed model SGCN-DSP achieves 86.9% $F1$ score in argument role labeling and 89.1% $F1$ score in predicate identification based on the CoNLL09 dataset. By integrating the current pre-trained language model BERT (Bidirectional Encoder Representation from Transformers, BERT), the task performance can be further improved. The proposed SGCN-DSP gives 90.4% $F1$ score in argument role labeling, and 90.8% $F1$ scores in predicate identification, respectively.

Keywords　natural language processing; semantic role labeling; syntax pruning; neural network; deep learning

《计算机学报》

1　引　言

自然语言处理/理解是人工智能领域的热门研究方向，其关键在于对语言的语义进行深层次的理解和表示. 语义角色标注(Semantic Role Labeling, SRL)通常被认为是自然语言处理/理解的核心任务之一，旨在分析出输入文本句子中所包含的谓词以及对应的语义论元，如“谁在何时何地对谁做了什么”的结构化信息. 这种语义结构信息可以用于自然语言处理的很多下游任务，如信息抽取[1-2]、自动问答[3-4]、阅读理解[5]、机器翻译[6-7]以及语言模型[8]等. 因此，语义角色标注任务近年来受到了研究者们持续的研究与关注. 早期的工作主要采用传统的统计模型，结合离散的人工构造特征来实现语义角色标注[9-11]. 最近的工作尝试设计各种不同的深度学习模型来自动完成特征的学习，避免了手工的特征设计[12-17]. 然而，已有研究主要集中于英文语义角色标注，其原因在于其有充分多的标注数据. 相比之下，受限于有限的标注数据[18]，中文语义角色标注的研究进展较为缓慢.

语义角色标注通常可划分为四个子任务，包括谓词识别、谓词消歧、论元识别和论元分类[19]. 早期的研究通常把任务建模为一个流水线过程，即先进行谓词的抽取[20]，再完成语义角色论元的标注[9]. 然而，基于流水线的方法因为两个离散阶段容易引入噪声和错误传播，导致任务的性能受限[21]. 最近，研究者们试图构建端到端的语义角色标注方法，即一个模型同时完成以上四个子过程一次性得到最终的所有结果[22]. 图 1 中给出了一个语义角色标注的例子，给定样例句子“市民也对即将诞生的欧元表现出极大兴趣.”，端到端模型旨在一次性全部识别出

五套浅层语义结构①:{表现出,市民,*A*0},{表现出,也对,*AM-ADV*},{表现出,兴趣,*A*1},{诞生,即将,*AM-ADV*},{诞生,欧元,*A*0}.由于其连续的过程可避免错误传播,端到端的方法极大地提升了语义角色标注的性能[23].目前绝大部分的端到端模型都是基于 span 图神经网络模型,穷尽式地遍历出所有可能的谓词以及对应的论元[19,22].

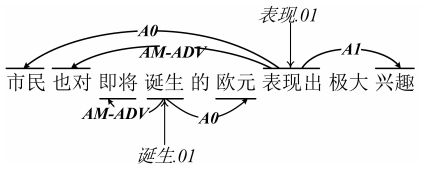

图 1 语义角色标注的例子

另一方面,大量的研究工作证实,外部句法结构信息可为语义角色标注模型提供高效的特征[24-27].直觉上,一个句子中的语义角色标注的"谓词-论元"结构与句法依存树的"依附-头部"结构有着密切的联系.以图 2 中的句子为例,可以观察到,"谓词-论元"结构({使,增长,*A*0},{使,中国,*A*1})与句法依存树结构({使,增长,*sbj*},{使,中国,*comp*})有着高度的重合与对应关系.因此,若将句法依存树结构作为输入特征编码到语义角色标注模型中可以一定程度上提升模型的推理能力.先前的基于句法结构的研究主要使用基于神经网络模型的句法编码器,如注意力机制[15,23]、树循环神经网络模型[28-29]、图卷积网络模型[14,28,30]等.然而相关研究指出,并不是外部句法结构特征中的所有信息都能起到帮助作用[31-32].以图 2 为例,实际上只有与语义角色标注结构高度相关的句法依存树结构(如高亮的句法依存对)才能提供真正有用的信息,而其他剩余的句法依存信息并不能提供有价值的信息(如暗灰色部分的句法依存对).基于此,He 等人(2018)[31]通过 k 阶的句法树裁剪技术,裁剪掉一些不相关的依存树分支从而使得输入的句法特征更为有效.但该方法采取了离线式的人工句法剪枝方案,即通过固定的 k 阶剪枝来裁剪掉所有的 k 阶之外的分支,这种方式会造成关键信息丢失或者裁剪效果减弱,从而限制了任务的性能.

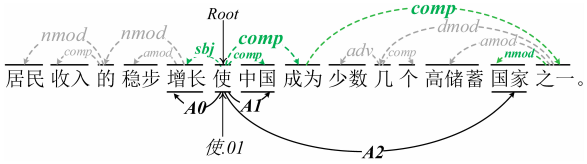

图 2 依存句法结构(上)与语义角色标注结构(下)

为解决上述问题,本文提出构建基于动态句法剪枝机制的端到端神经网络模型,实现中文语义角色标注.图 3 展示了所提出模型的整体框架,主要包括表征编码层、句法融合层和解码层.首先,利用基于自注意力机制的 Transformer 模型[33]作为基础表征编码器来学习输入句子的上下文表征.接着,使用句法融合层为外部句法依存树信息进行编码,在信息传播的同时考虑句法树的动态裁剪.本文提出两种方法进行句法融合以及对应的动态句法裁剪机制,分别为基于递归神经网络模型的动态句法剪枝机制(Recur-DSP)和基于带句法标签的图卷积网络模型的句法剪枝机制(SGCN-DSP).同时我们观察到,目前的句法依存编码器仅仅编码依存树中的依存弧结构,而没有利用依存树结构所对应的句法标签信息.实际上,不同依存弧下相邻节点的信息可提供不同程度的贡献,而句法标签恰能表达这种贡献差异,从而为语义角色标注任务提供重要的推理证据.以图 2 为例,同样是"*comp*"类型的依存弧,依存对{的,收入,*comp*}和{使,中国,*comp*}提供了截然不同的信息,而仅仅后者的信息是有帮助的.因此,基于目前的图卷积网络模型提出带句法标签的图卷积网络依存句法编码模型,同时提出对应的句

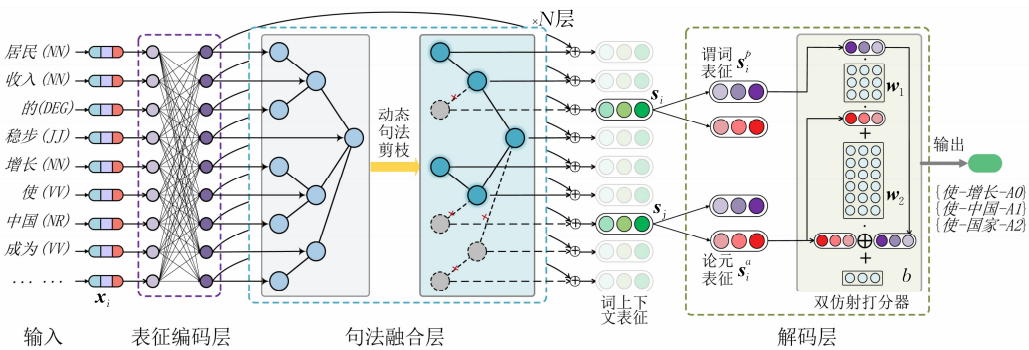

图 3 本文所提模型的整体框架

① 本文将一个语义结构表述为一个三元组结构:{谓词,论元,角色}.

法剪枝机制.最后,基于获得的词表征分别构建谓词表征和论元表征,并通过双仿射打分器对每一对谓词和论元进行迭代式配对打分,最终输出所有合法的结果.

为验证所提方法的有效性,我们在中文语义角色标注的基准数据集上进行了实验,包含 CoNLL09[34] 和 Universal Proposition Bank[35-36].实验结果显示本文所提出的基于动态句法剪枝机制的端到端神经网络模型超过了目前最先进的基线模型,取得了当前中文语义角色标注任务的最优性能.同时,消融实验量化了所提模型的各部分的作用和贡献.后续的分析实验,进一步验证了动态句法剪枝机制对语义角色标注任务的影响.除此之外,我们还基于所提出的带句法标签的图卷积网络依存句法编码模型分析了语义角色结构与句法结构的依赖关系.本文的贡献可归纳为以下三点:

(1)本文首次提出了基于 Gumble-Softmax 函数离散化原理的语义角色标注动态句法剪枝机制,从而弥补前驱研究中的固定离线式的句法裁剪方法的短板,进一步提升中文语义角色标注任务的性能.我们基于递归神经网络编码器和图卷积网络编码器分别提出两类对应的动态剪枝模型.

(2)我们改进了图卷积网络编码器,提出了一个带句法标签的图卷积网络句法编码器,即同时将依存弧与依存标签信息统一建模于图卷积网络中,可更加充分地结合外部句法知识.基于此,我们实现了基于带句法标签的图卷积网络编码器的动态剪枝模型,实现了更为高效的动态剪枝效果.

(3)所提模型在中文语义角色标注的基准数据集上超过了目前最先进的基线模型,取得了当前最佳的中文语义角色标注性能.我们提供了后续的分析实验,深入探究了动态句法剪枝机制对语义角色标注任务的影响.

2　相关工作

语义角色标注任务由 Jurafsky 等人(2000)[37] 提出,旨在分析出文本中句子所包含的谓词以及对应的语义论元,诸如"谁在何时何地对谁做了什么"的语义结构.作为自然语言处理的基本研究任务之一,语义角色标注一直是自然语言处理领域的热门研究话题.下面从语义角色标注资源,模型与方法,基于句法特征的语义角色标注和中文语义角色标注四个方面来分析和总结相关研究进展.

2.1　语义角色标注资源

基于句子的句法结构,研究人员构建了一些代表性的语义角色标注资源,例如 FrameNet[38] 和 PropBank[39].FrameNet 以格语法理论[40] 为基础,分类依据为词汇的出现场景,以框架为基本组成单元.为了区分不同的框架语义,每个框架独自定义了语义角色,称为框架元素.FrameNet 的谓词包含名词、动词及形容词,需要指出的是,不同词汇上的谓词可能会引发相同的框架,因此每个框架具有独自的元素或者角色标签.基于 Penn Treebank[41],PropBank 对动词性谓词的论元进行了语义标注,其动词分类体系来源于 VerbNet[42],这里动词性谓词也有若干语义角色被定义.这里,核心的论元角色主要包含 6 种,标记为 $A0\sim A5$,其中施事者用 $A0$ 表示,受事者用 $A1$ 表示,$A2\sim A5$ 基于谓词的不同从而表达不同的语义.附属论元角色包含各种修饰性的语义,诸如"何时何地、何种方式",比如 *AM-ADV*、*AM-LOC*、*AM-MNR*、*AM-TMP* 等.PropBank 具有局部性和全局性特点,由于其为论元增加了描述信息.但框架元素为 FrameNet 框架所特有的,同时随着框架数量而增加.PropBank 中的框架根据具体的谓词的意态而具有不同的意义.另外,PropBank 认为相同的语义角色,其语法特征应该具有一致性,因此其语义角色中更多考虑的是语法层面.而 FrameNet 偏重从语义层面来标注框架元素,其认为相同的框架元素可能有不同的语法特征.

2.2　模型与方法

从模型构架上,已有模型主要分为两类,即基于传统的统计机器学习模型和基于深度学习的模型.早期的研究主要采用基于传统的统计机器学习模型实现语义角色标注任务[9-11].这类方法一般涉及到特征工程,即通过人工定义特征模板构造离散的特征,诸如 n-gram 特征作为统计模型的输入.近几年,随着深度神经网络模型的快速发展,研究人员尝试使用各种深度学习模型从输入文本中自动提取有效的特征[12-16].深度学习模型有效地避免了人工特征工程所导致的劳力消耗,同时又能极大地提升任务的性能.现有的绝大部分语义角色标注方法都为深度神经网络模型,并且在各类数据集上获得了优越的任务性能[14,17,26].

已有研究通常将语义角色标注任务划分为四个子过程,包括谓词识别、谓词消歧、论元识别和论元分类[19,22].前期的语义角色标注研究,受限于方法的进展,通常采用一种流水线范式的任务建模.他们

考虑先离线式地完成谓词部分的预测，包含谓词识别与谓词消歧[20]，再基于预测出来的谓词信息完成语义角色论元部分的预测，包含论元识别和论元分类[9]. 然而，传统的流水线方法的任务性能往往会受到两方面因素的限制. 第一，流水线方法隔离了谓词信息与论元信息之间的动态交互；第二，谓词预测和论元预测两个阶段的分离会引入噪声和错误传播[21]. 另外，一大部分的工作集中于探索第二阶段的语义角色论元的标注，即他们考虑直接将标准正确的谓词标注信息作为给定条件，从而简化了任务[15,19,30]. 这时，任务被建模为一个标准的序列标注问题，一般通过条件随机场模型或者长短时记忆网络模型等方法即可达到较好的效果[13].

最近，越来越多的研究兴趣转移到构建端到端的语义角色标注框架. 具体地，研究者们尝试将语义角色标注中的四个子任务通过一个学习模型同时进行建模，并一次性输出所有的结果，即所有可能的谓词以及其对应的论元[22]. 由于端到端的预测是一个连续的过程而不会造成额外的损耗，该方法极大地提升了任务的性能[23]. 目前绝大部分的端到端模型都是基于 span 图的神经网络模型，穷尽式地遍历出所有可能的谓词以及对应的论元[19,22]. 例如，He 等人(2018)[22]通过使用片段短语图的网络框架，为语义角色标注任务实现了首次端到端的识别，在 CoNLL05 数据集上超越了所有基于流水线的方法. 最近，Fei 等人(2020)[43]基于图网络构架模型提出高阶提纯机制，使得模型能够自动迭代更新，学到全局的高阶特征，从而提升语义角色标注的性能. 本研究考虑采用这种端到端的语义角色标注模式，并以此为基准模型，并进一步改进其性能.

2.3 基于句法特征的语义角色标注

已有的大量工作证实，传统的统计模型结合有效的句法特征，能帮助推导语义角色的约束和抽取，提升任务的性能[10,37]. 尽管基于深度学习的模型在语义角色标注任务获得较为显著的性能，但如何设计创新的神经网络模型来构建句法特征仍然是一个具有挑战性的研究话题[24-27]. 基于此，Roth 和 Lapata(2016)[13]将句法依存树路径嵌入表征引入到神经网络模型中. Marcheggiani 和 Titov(2017)[25]在双向长短时记忆网络编码器的基础上采用了图卷积神经网络来编码句法信息. He 等人(2018)[31]提出了一种基于句法依存树的 k 阶论元剪枝算法. Strubell 等人(2018)[15]提出了基于自注意机制的神经网络模型，该模型通过多任务学习框架将句法依存解析任务作为语义角色标注的辅助任务来提升主任务的性能. Xia 等人(2019)[29]研究并比较了几种基于广度的 SRL 句法感知方法，揭示了整合句法信息的有效性. 总的来说，现有的基于句法结构的语义角色标注工作主要采用基于神经网络模型的句法编码器，包括注意力机制[15,26]、树循环神经网络模型[28-29]、图卷积网络模型[14,28,30]等.

然而，最近的研究发现并非所有的外部句法结构特征知识都能帮助模型进行语义角色推理[31-32]. 实际上，只有一部分句法结构信息能为语义角色标注模型提供真正有价值的信息，比如与语义角色标注结构高度相关部分的句法依存结构，而与其无关的句法依存信息的价值是非常有限的，甚至有时会影响到模型的推理. 基于此，He 等人(2018)[31]通过 k 阶的句法树裁剪技术，发现裁减掉一些不相关的依存树分支可使得输入的句法特征更为有效. 但我们注意到，He 等人(2018)采取了离线式人工句法剪枝方案，通过固定的 k 阶剪枝来裁剪掉所有的 k 阶之外的分支，这种人工定式裁剪方式会造成关键信息的丢失或者裁剪效果的减弱. 因此，为提升语义角色标注任的性能，本文提出基于动态句法剪枝机制的端到端神经网络模型，来弥补已有语义角色标注研究中的人工离线式的句法裁剪方法的短板.

2.4 中文语义角色标注

对语义角色标注任务来说，已有的工作大多集中于英语，相比之下，中文语义角色标注工作受限于有限的标注数据[18]，研究相对较少，任务性能相对较低. 为此，一部分研究工作考虑利用外部句法结构知识来扩充模型的能力，提升中文语义角色标注的性能. 例如，Sun 等人(2009)[44]将任务建模成序列标注问题，使用支持向量机，同时整合构造的词法和句法特征，实现了中文语义角色标注. Wang 等人(2015)[45]构建双向长短时记忆网络模型，并介绍一种通过共享单词嵌入机制来注入多源异构数据从而提升任务性能. Xia 等人(2017)[46]提出了从异构语义角色标注数据中学习和转移知识的渐进模型. 之后，Xia 等人(2019)[18]提出一个多任务学习框架，通过从依存句法解析模型中提取出隐式语法表示并作为语义角色标注模型的外部输入，从而提升任务性能. 然而我们注意到，上述工作将完整的句法结构特征作为输入，而没有考虑到句法特征的剪枝，未删除句法结构特征中对语义角色标注无效或者冗余的特征，这将一定程度上影响任务预测的效果. 基于此，本文提出基于动态句法剪枝机制的神经网络框架，

所提框架可为任务捕捉到有效的句法特征，以此来提升任务的性能.

3 方　法

本节将详细介绍所提方法的具体技术细节. 首先介绍端到端中文语义角色标注任务的建模，接着介绍语义角色标注整体架构，包含编码和解码过程. 然后详细介绍所提出的两种融合句法知识的编码器，以及所对应的动态句法剪枝机制. 最后给出具体的模型训练思路和细节.

3.1 任务建模

首先对任务进行建模，用数学公式进行形式化表达. 我们将其建模为一个“谓词-论元-角色”三元组预测任务[19,22]. 给定一个输入句子 $S=\{w_1,\cdots,w_n\}$，端到端语义角色标注系统最终需预测并输出一个三元组列表，$\mathcal{Y}\in\mathcal{P}\times\mathcal{A}\times\mathcal{R}$，其中 $\mathcal{P}=\{p_1,\cdots,p_m\}$ 为所有可能的谓词，$\mathcal{A}=\{a_1,\cdots,a_l\}$ 为所有与谓词关联的论元词，而 $\mathcal{R}=\{l_1,\cdots,l_g\}$ 为对应的论元 a 的语义角色标签，如 *A0*、*A1*、*AM-ADV*、*AM-LOC*、*AM-TMP* 等，其中包含一个空标签 ϵ 以表示目前的备选谓词与论元之间不能形成合理配对.

3.2 语义角色标注框架

如图 3 所示，本文提出的语义角色标注模型包括表征编码层，句法融合层和解码层. 首先，表征编码层作为第一层将句子进行编码以初步得到句子的上下文信息. 这里采用 Transformer 模型作为该表征编码器. 接着在第二层即句法融合层，句法编码器将融合句法结构信息到句子上下文表征中，在此期间，句法结构将会被动态裁剪以获得最佳的句法表征. 最后，基于所获得的词表征分别构建谓词表征和论元表征，解码层通过双仿射打分器对每一对谓词和论元进行迭代式配对打分，最终输出所有合法的结果.

3.2.1 输入特征

本文所提框架的输入特征包含两部分内容，一方面是基于输入句子的特征表示，另一方面是基于外部句法结构的特征表示. 对于基于输入句子的特征，首先通过一个词向量表为每个中文词 w_i 获取其向量表示 $\boldsymbol{x}_i^w$，即词向量. 然后再考虑将词性标注信息作为支撑特征. 具体地，首先获取每个词的词性标注(Part-of-Speech，POS)标签，再通过向量表获取到对应的嵌入式表示 $\boldsymbol{x}_i^{pos}$. 为了充分利用中文词的结构特征，我们再利用一个卷积神经网络模型(Convolutional Neural Networks，CNN)来编码每个中文词语所包含的字信息，即可得到字级别特征 $\boldsymbol{x}_i^c$. 然后我们将以上所有的特征向量拼接起来，构成了整体的句子输入特征：

$$\boldsymbol{x}_i=[\boldsymbol{x}_i^w;\boldsymbol{x}_i^{pos};\boldsymbol{x}_i^c] \tag{1}$$

对于基于外部句法结构的特征，我们主要考虑依存句法树. 具体的依存句法特征表示方法将在下一个小节中结合句法编码器给出具体的介绍.

3.2.2 表征编码层

这里采用基于自注意力机制的 Transformer 编码器. Transformer 已在众多的自然语言处理任务中证实其句子表征能力，能高效地学习到句子中任意两个词之间的互相作用[33]. 具体地，在 Transformer 编码器中，输入的句子表征 $\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n\}$ 首先通过三个线性层被分别映射为查询向量 $\boldsymbol{Q}$，值向量 $\boldsymbol{V}$ 和键向量 $\boldsymbol{K}$. 然后通过一个扩展点积对齐函数计算出 $\boldsymbol{K}$ 与 $\boldsymbol{Q}$ 的相关程度值，再乘以 $\boldsymbol{V}$：

$$\boldsymbol{\beta}=\mathrm{Softmax}\left(\frac{\boldsymbol{Q}\cdot\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\cdot\boldsymbol{V} \tag{2}$$

其中 d_k 为扩展因子. 在具体实现中，查询向量 $\boldsymbol{Q}$，值向量 $\boldsymbol{V}$ 和键向量 $\boldsymbol{K}$ 都源自于句子表征 $\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n\}$. 不同的平行的注意力头可关注到输入句子特征的不同频道，从而实现最大化的特征交互. 最后，m 个不同的平行注意力头所产生的输出特征向量拼接在一起形成 Transformer 特征表征：

$$[\boldsymbol{\beta}_1;\cdots;\boldsymbol{\beta}_m]\cdot\boldsymbol{W}+b_{\boldsymbol{\beta}} \tag{3}$$

为了方便表述，我们将 Transformer 的表征编码过程简化为以下公式：

$$\{\boldsymbol{r}_1,\cdots,\boldsymbol{r}_n\}=\mathrm{Trm}(\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n) \tag{4}$$

其中，$\boldsymbol{r}_i$ 是每个词 w_i 所对应的上下文表征.

3.2.3 句法融合层

基于表征编码层所输出的上下文表征向量 $\boldsymbol{r}_i$，我们进一步利用句法编码器将外部句法知识编码进来，得到蕴含句法结构的隐状态表征. 在句法信息聚合过程中，系统将执行动态句法剪枝操作，只传播和融合对任务最有帮助的句法知识. 我们将在第 3.3 节中详细介绍句法融合层，这里不妨将句法融合层的计算过程表述为

$$\{\boldsymbol{s}_1,\cdots,\boldsymbol{s}_n\}=\mathrm{DSP}(\{\boldsymbol{r}_1,\cdots,\boldsymbol{r}_n\}) \tag{5}$$

其中 $\mathrm{DSP}(\cdot)$ 为句法融合层过程. 具体地，$\boldsymbol{s}_i$ 在下一小节为 $\boldsymbol{s}_i^{\mathrm{recur}}$ 或者 $\boldsymbol{s}_i^{\mathrm{sgcn}}$. 为充分利用到上一步得到的上下文特征，考虑将上下文表征向量 $\boldsymbol{r}_i$ 与融合句法特征向量 $\boldsymbol{s}_i$ 做一个拼接操作，

$$\boldsymbol{s}_i:=[\boldsymbol{s}_i;\boldsymbol{r}_i] \tag{6}$$

$\boldsymbol{s}_i$ 是每个词 w_i 所对应的融合句法知识的上下文表征.

《计算机学报》

3.2.4 解码层

接下来，基于所产生的对应每个词 w_i 融合句法知识的上下文表征 $\boldsymbol{s}_i$，分别通过单层前馈神经网络(Feed-Forward Network，FFN)衍化出谓词表征以及论元表征：

$$\boldsymbol{s}_i^p;\boldsymbol{s}_i^a=\mathrm{FFN}^p(\boldsymbol{s}_i);\ \mathrm{FFN}^a(\boldsymbol{s}_i) \tag{7}$$

其中上标 p 与 a 分别表示与得到谓词和论元的表征的前馈神经网络非线性变换的相关操作.

为了有效度量出潜在的谓词-论元对，本文考虑两种打分策略：一元打分器与二元打分器. 首先讨论一元打分器，即用一个标量分数来度量每个词成为谓词或者论元的可能性：

$$S_i^p=\boldsymbol{W}_p^{\mathrm{T}}\mathrm{FFN}^p(\boldsymbol{s}_i^p),\ \phi_i^p=\exp(S_i^p) \tag{8}$$

$$S_j^a=\boldsymbol{W}_a^{\mathrm{T}}\mathrm{FFN}^a(\boldsymbol{s}_j^a),\ \phi_j^a=\exp(S_j^a) \tag{9}$$

其中，S_i^p 与 S_j^a 分别为谓词与论元的非线性变换过程，ϕ_i^p 与 ϕ_j^a 分别为谓词与论元的打分器的输出分数，具体分别通过一个对数函数 $\exp(*)$ 实现，$\boldsymbol{W}_p\in\mathbb{R}^{d_p}$ 与 $\boldsymbol{W}_a\in\mathbb{R}^{d_a}$ 是可学习更新的模型参数. 然后再采用一个二元打分器基于每个角色标签 l 度量每个谓词-论元对之间可成功组合的可能性. 具体地，我们采用了一个双仿射打分器[47]：

$$S_{ij}^l=\begin{bmatrix}\boldsymbol{s}_j^a\\1\end{bmatrix}^{\mathrm{T}}\boldsymbol{W}_l\boldsymbol{s}_i^p,\quad \phi_{ij}^l=\exp(S_{ij}^l) \tag{10}$$

这里 S_{ij}^l 和 ϕ_{ij}^l 分别表示角色标签的非线性变换过程与打分，$\boldsymbol{W}_l\in\mathbb{R}^{(d_l+1)\times d_l}$ 为二维学习参数，其中第一维包含了一项偏置项，因此，论元表征向量 $\boldsymbol{s}_j^a$ 与 1 进行拼接保证了与该参数中第一维偏置项维度的对齐.

基于上述的打分器的分数，我们为每个三元组“谓词-论元-角色标签”结构针对每个所输入的句子 S 构造其计算因子 $P_\theta(\boldsymbol{Y}|S)$：

$$P_\theta(\boldsymbol{Y}|S)=\prod_{p\in\mathcal{P},a\in\mathcal{A}}P_\theta(y(p,a)|S) \tag{11}$$

其中，

$$P_\theta(y(p,a)|S)=\frac{\Phi(p,a,l)}{\sum_{\hat{l}\in R}\Phi(p,a,\hat{l})} \tag{12}$$

$$\Phi(p,a,l)=\phi_{ij}^l+\phi_i^p+\phi_j^a \tag{13}$$

这里 $\Phi(p,a,l)$ 为“谓词-论元-标签”的打分总和，$P_\theta(y(p,a)|S)$ 为“谓词-论元”的局部因子，θ 为模型的整体参数. 对于每一对“谓词-论元”(p_i,a_j)，可通过式(12)计算与之对应的角色标签 l 组合下的合法性. 即如果得到的角色标签不为空标签($l\neq\epsilon$)，模型则输出该三元组(p_i,a_j,l).

3.3 句法融合与动态句法剪枝机制

本文构建了两种句法融合与动态句法剪枝方法，包括(1)基于递归神经网络模型的依存句法编码器以及句法剪枝机制，如图 4 所示；(2)基于带句法标签的图卷积网络模型的句法编码器以及句法剪枝机制，如图 5 所示. 下面分别具体介绍这两种方法.

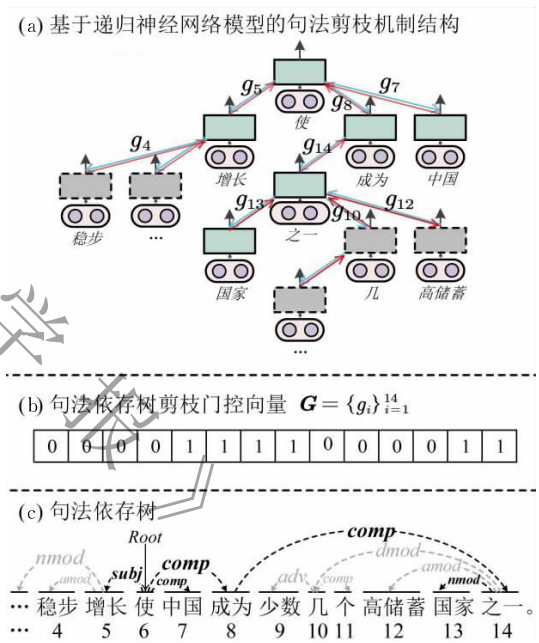

图 4 基于递归神经网络模型的依存句法编码器和对应的句法剪枝机制

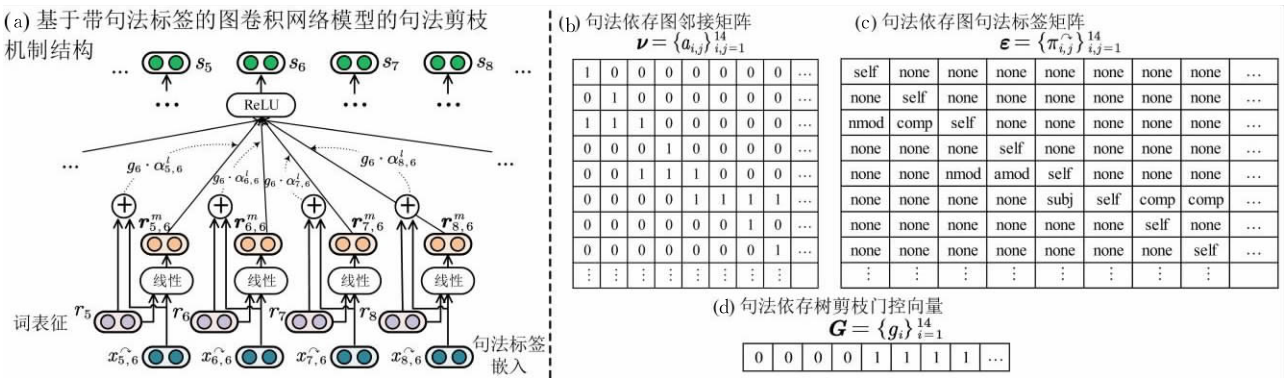

图 5 基于带句法标签的图卷积网络模型的句法编码器和对应的句法剪枝机制

3.3.1　基于递归神经网络模型的句法剪枝机制

受已有工作的启发[48]，首先采用递归神经网络模型(Recursive Neural Network)编码依存句法结构.具体地，考虑用一个孩子规约(Child-Sum)结构的树递归模型来编码依存树，采用长短时记忆网络(Long Short-Term Memory，LSTM)为基本单元.同时，考虑一个双向的连通结构来充分捕获树结构中的信息交互，即自下而上和由上至下的递归树结构.

基于给定的输入句子的依存句法结构，如图 4(c)所示，构建出递归神经网络结构.递归神经网络中每个节点 j 以 Transformer 表征编码层的上下文输出为输入特征，对应到每个输入的词 w_j.在这个双向的树传播结构中，其中自下而上的过程表述为

$$\begin{aligned}
&\bar{\boldsymbol{h}}_i^{\uparrow}=\sum_{j\in C(i)}\boldsymbol{h}_j^{\uparrow},\\
&\boldsymbol{i}_i=\sigma(\boldsymbol{W}^{(i)}\boldsymbol{r}_i'+\boldsymbol{U}^{(i)}\bar{\boldsymbol{h}}_i^{\uparrow}+b^{(i)}),\\
&\boldsymbol{f}_{ij}=\sigma(\boldsymbol{W}^{(f)}\boldsymbol{r}_i'+\boldsymbol{U}^{(f)}\bar{\boldsymbol{h}}_j^{\uparrow}+b^{(f)}),\\
&\boldsymbol{o}_i=\sigma(\boldsymbol{W}^{(o)}\boldsymbol{r}_i'+\boldsymbol{U}^{(o)}\bar{\boldsymbol{h}}_i^{\uparrow}+b^{(o)}),\\
&\boldsymbol{u}_i=\mathrm{Tanh}(\boldsymbol{W}^{(u)}\boldsymbol{r}_i'+\boldsymbol{U}^{(u)}\bar{\boldsymbol{h}}_i^{\uparrow}+b^{(u)}),\\
&\boldsymbol{c}_i=\boldsymbol{i}_i\odot\boldsymbol{u}_i+\sum_{j\in C(i)}\boldsymbol{f}_{ij}\odot\boldsymbol{c}_j,\\
&\boldsymbol{h}_i^{\uparrow}=\boldsymbol{o}_i\odot\tanh(\boldsymbol{c}_i)
\end{aligned}\tag{14}$$

其中，$\boldsymbol{W}$，$\boldsymbol{U}$ 和 b 均为模型参数.$C(i)$是节点 i 的子节点集合.$\boldsymbol{h}_j$，$\boldsymbol{i}_j$，$\boldsymbol{o}_j$ 以及 $\boldsymbol{c}_j$ 分别表示节点 j 的隐状态、输入门控、输出门控及记忆门控.$\boldsymbol{f}_{jk}$ 是节点 j 的各个子节点 k 的忘记门.$\sigma(\cdot)$为 Sigmoid 激活函数，$\odot$ 为元素级别的乘法.相同地，由上至下的递归神经网络结构与上述自下而上过程有相同的转移方程式，我们将其得到的节点表征记为 $\boldsymbol{h}_i^{\downarrow}$.将两个方向的树结构表征在每个对应词的位置进行拼接即可得到最终的基于递归神经网络模型的句法编码器输出向量：

$$\boldsymbol{s}_i^{\mathrm{recur}}=[\boldsymbol{h}_i^{\uparrow};\boldsymbol{h}_i^{\downarrow}]\tag{15}$$

基于递归神经网络模型进行动态句法结构裁剪，记为 Recur-DSP.在式(14)中，由于采用标准的 Sigmoid 激活函数，输出门控的输出值 $\boldsymbol{o}_i$ 为一个连续的概率值.考虑 $p_{i,1}=\boldsymbol{o}_i$ 为门控开放的概率，$p_{i,0}=1-\boldsymbol{o}_i$ 为门控关闭的概率.为了达到动态的剪枝效果，需要一个离散的控制机制，即进行[0，1]取值的决策.为此，采用 Gumbel-Softmax 函数[49]来完成此离散控制过程.具体地，Gumbel-Softmax 函数完成从 $\boldsymbol{o}_i$ 连续分布变换到 0-1 分布的过程：

$$g_i=\frac{\exp(\log p_{i,1}+\epsilon)/\tau}{\sum_{t=0,1}\exp(\log p_{i,t}+\epsilon)/\tau}\tag{16}$$

其中，τ 为协调因子，在训练过程中，当 τ 逐渐下降趋向于 0 时，上式的 g_i 将得到离散的 0-1 分布近似值.ϵ 为从 Gumbel 分布中随机抽样而得的变量，其中 Gumbel 函数为

$$\mathrm{Gumbel}(0,1)=-\log(-\log(\mathrm{Uniform}(0,1)))\tag{17}$$

此处 Uniform(0，1)即为标准正态分布.

对基于句法树的递归神经网络中的每对节点之间的连接，裁剪机制 Recur-DSP 都会进行一次离散的门槛判断，即可得到 g_i.遍历整棵树后将得到整体的剪枝门控向量 $\boldsymbol{G}=\{g_i\}_{i=1}^{n}$，如图 4(b)所示.接着，将 g_i 取代式(14)中输出门控 $\boldsymbol{o}_i$ 值再进行递归神经网络计算，相应的为每个词 w_i 得到裁剪后的句法表征.对于自下而上的剪枝过程有 $\boldsymbol{h}_i^{\uparrow}=g_i\odot\tanh(\boldsymbol{c}_i)$，而对于由上至下过程有 $\boldsymbol{h}_i^{\downarrow}=g_i\odot\tanh(\boldsymbol{c}_i)$.最终将两个方向的每个对应词的位置进行拼接，即通过式(15).本裁剪过程如图 4(a)所示.

3.3.2　基于带句法标签的图卷积网络模型的句法剪枝机制

利用不同的句法编码模型可得到异质的句法结构特征表征.相比于递归神经网络模型的依存句法编码器，基于图卷积神经网络模型(Graph Convolutional Network，GCN)的句法编码近年来受到了更多的研究热度[14].主要原因在于，图卷积神经网络模型的计算效率更高，对句法树的计算仅有 $O(1)$ 的常数级复杂度.然而我们注意到，在现有的基于图卷积神经网络模型编码句法树的语义角色标注工作中，只考虑了依存弧结构的编码，而忽略了对依存树结构的句法标签信息的利用.在本研究中，我们考虑同时将句法依存弧以及对应的标签信息进行建模.为此，提出一个基于带句法标签的图卷积网络模型句法编码器(Syntax-label Graph Convolutional Network，SGCN).SGCN 将依存弧与标签信息统一建模，同时被归一化为一个分布来更好的表示句法树中的节点之间的邻接强度.

图 5(a)展示了所提出的基于带句法标签的图卷积网络模型句法编码器.给定一个句子以及对应的依存句法树，为该句法树构建一个依存图$(\mathcal{V},\mathcal{E})$，其中$\mathcal{V}$为实词节点集合，$\mathcal{E}$为依存对(依赖节点与头节点)之间的带有弧标签的双向边的集合.在$\mathcal{V}$中，$a_{i,j}=1$ 表示节点 w_i 与节点 w_j 之间存在着弧，反之则 $a_{i,j}=0$.在$\mathcal{E}$中，除了预定义好的依存标签(如“nmod”，“comp”，“sbj”等)，我们额外为每一个节点增加一个“self”标签来表示其自环边，以促进信息的充分传播与利用.同时，用一个“none”标签表示所有的不形成逻辑连接的节点对之间的边.以图 2 中的

《计算机学报》

句子为例，其对应的依存图的实词节点矩阵以及句法标签矩阵如图 5(b)和(c)所示.

我们为每一个边标签 $\pi_{i,j}^{\curvearrowright}$ 创建其对应的嵌入式表征 $\boldsymbol{x}_{i,j}^{\curvearrowright}$. 基于带句法标签的图卷积网络模型句法编码器的输入为句子中词表征 $\boldsymbol{r}_i$ 以及边标签的嵌入式表征 $\boldsymbol{x}_{i,j}^{\curvearrowright}$，输出的是编码句法信息的表征 $\boldsymbol{s}_i$：

$$\boldsymbol{r}_{i,j}^{m}=\boldsymbol{W}_1\cdot\boldsymbol{r}_j+\boldsymbol{W}_2\cdot\boldsymbol{x}_{i,j}^{\curvearrowright}+b,$$

$$\boldsymbol{s}_i^{\text{sgcn}}=\text{ReLU}\Big(\sum_{j=1}^{n}\boldsymbol{\alpha}_{i,j}\boldsymbol{r}_{i,j}^{m}\Big) \tag{18}$$

这里，$\boldsymbol{r}_{i,j}^{m}$ 为线性变换的中间过程表征，$\boldsymbol{\alpha}_{i,j}$ 为图卷积网络模型的句法图邻接矩阵，由下式计算而得：

$$\boldsymbol{\alpha}_{i,j}=\frac{a_{i,j}\cdot\exp(\boldsymbol{r}_i^{n}\cdot\boldsymbol{r}_j^{n})}{\sum_{j=1}^{n}a_{i,j}\cdot\exp(\boldsymbol{r}_i^{n}\cdot\boldsymbol{r}_j^{n})} \tag{19}$$

其中 $\boldsymbol{r}_i^{n}$ 是通过为 $\boldsymbol{r}_i$ 和 $\boldsymbol{x}_{i,j}^{\curvearrowright}$ 进行元素级别的加和所得到的表征，即 $\boldsymbol{r}_i^{n}=\boldsymbol{r}_i+\boldsymbol{x}_{i,j}^{\curvearrowright}$. 句法依存图邻接矩阵 $\boldsymbol{\alpha}_{i,j}$ 编码了丰富的依存弧与标签信息，因此可以充分反映出句法信息在语义角色标注任务的属性.

下一步，我们为基于带句法标签的图卷积网络模型句法编码器进行句法剪枝. 具体地，考虑将句法依存图邻接矩阵 $\boldsymbol{\alpha}_{i,j}$ 装配离散化门控机制，即：

$$\hat{\boldsymbol{\alpha}}_{i,j}=\frac{g_j\cdot\boldsymbol{\alpha}_{i,j}}{\eta+\sum_{j'=1}^{n}g_{j'}\cdot\boldsymbol{\alpha}_{i,j'}} \tag{20}$$

其中，分母中的 η 为极小值常量用以避免数值计算不稳定性，g_j 即为剪枝门控，从式(16)的 Gumbel-Softmax 函数计算获得，其中 $p_{i,1}$ 门控开放的概率由下式获得：

$$p_{i,1}=\sigma(\boldsymbol{V}^{\mathrm{T}}\text{Tanh}(\boldsymbol{W}\boldsymbol{r}_i)) \tag{21}$$

其中 $\boldsymbol{V}$ 和 $\boldsymbol{W}$ 为可训练参数. 图 5(d)展示了对应的句法依存树剪枝门控向量.

为了进一步保证句法剪枝裁剪过程中的剪枝力度可控性，考虑再对基于递归神经网络模型的句法剪枝机制以及基于带句法标签的图卷积网络模型的句法剪枝机制的门控开放概率 $p_{i,1}$ 乘以一个伸缩控制因子 $\lambda>0$：

$$p_{i,1}:=\text{Min}(\lambda\cdot p_{i,1},1) \tag{22}$$

3.4 训练与优化

语义角色标注模型的训练目标旨在根据标准的结构去最小化计算因子 $P_\theta(\hat{\boldsymbol{Y}}|\boldsymbol{S})$(式(11))的负对数似然结构：

$$\mathcal{L}=-\log P_\theta(\boldsymbol{Y}|\boldsymbol{S})+\zeta\|\theta\|_2^2 \tag{23}$$

其中公式的第二项为 ℓ_2 正则项，用以防止模型参数过拟合.

由于本文所提的端到端模型需要穷尽式的遍历所有可能的谓词-论元-语义标签三元组，导致昂贵的计算复杂度代价，即 $O(n^2g)\approx O(n^2)$，其中 $g=\|\mathcal{R}\|\ll n$ 为所有的标签的数量. 借鉴先前的工作[22]，我们定义了两组用来缓存候选谓词和论元的容器. 谓词的缓存容器大小为 $\mu_p\cdot n$，论元的缓存容器大小为 $\mu_a\cdot n$. 首先所有的候选谓词和论元会根据式(8)与式(9)得到的一元分数按照分数的由高到低进行排列，然后那些排在容器的容量之外的候选将会被舍弃掉.

4 实验设置

第 4 节介绍实验阶段所需的环境设置、数据集、对比的基线模型以及评估方法.

4.1 实验数据集

本文所有实验基于中文语义角色标注基准数据集，包括 CoNLL09 的中文数据集① 和 Universal Proposition Bank(UPB)的中文集②. 每个数据集都有着对应的训练集、开发集以及测试集. 数据集中的中文已经进行了分词，无需人工再次预处理. 此外，每条句子都给定了标准的句法标注信息，包括每个词的词性标注标签和依存句法解析树，无需做额外的句法分析处理. 表 1 给出了数据集的统计信息.

表 1 数据集的统计信息

	句子	词	谓词	论元
CoNLL09				
训练集	22277	609060	104019	231869
开发集	1762	49620	8202	18554
测试集	2556	73153	12438	27712
UPB				
训练集	3997	98608	10495	21322
开发集	500	12663	1304	2681
测试集	500	12012	1302	2610

4.2 模型参数

采用了预训练的中文 Fasttext 词向量③作为系统的初始化词嵌入层. Transformer 编码器的层数为 2. 例外，实验还用到了中文 BERT 预训练语言模型④来初始化 Transformer 编码器，这里 Transformer 的层数为 12. 词向量的维度为 300 维. 编码字的卷积神经网络的卷积核分别为[3,4,5]，字编码嵌入维度为 50 维. 依存树标签嵌入 $\boldsymbol{x}_{i,j}^{\curvearrowright}$ 的维度为 100. 根据实验结果，Recur-DSP 的层数为 2，SGCN-DSP 的层

① https://catalog.ldc.upenn.edu/LDC2012T03
② https://github.com/System-T/UniversalPropositions
③ https://fasttext.cc
④ https://github.com/google-research/bert，Chinese-base 版本

计算机学报

数为 3. 句法剪枝机制的伸缩控制因子 λ：Recur-DSP 中取值为 1.6，SGCN-DSP 中取值为 0.8. 实验中采用反向传播算法进行参数优化. 使用 Adam 优化器来训练模型，其学习率初始值为 1e-5. 采用小批次(mini-batch)的方式训练样本，每个批次大小为 CoNLL09 数据集为 32，UPB 数据集为 16. 模型在训练时不固定迭代数量，而是采用提前停止策略(early-stop)确定迭代次数. 实验的环境为 i9 Intel CPU 以及 NVIDIA RTX 3090Ti GPU. 其他的模型参数设置如表 2 所示.

表 2　模型的超参数设置

参数名	值	参数名	值	参数名	值
词向量维度	300	依存树标签嵌入维度	100	词性标注标签嵌入维度	50
字嵌入维度	50	字卷积过滤器	50	字卷积核	[3,4,5]
Transformer 编码器层数	2	Transformer 表征 $\boldsymbol{r}_i$ 维度	768	词上下文表征维度	1068
Recur-DSP 层数	2	Recur-DSP 表征 $\boldsymbol{s}_i^{\text{recur}}$ 维度	300	Recur-DSP 协调因子 τ 初始值	0.60
SGCN-DSP 层数	3	SGCN-DSP 表征 $\boldsymbol{s}_i^{\text{sgcn}}$ 维度	300	SGCN-DSP 协调因子 τ 初始值	0.75
BERT 维度	768	一元打分器维度	300	双仿射打分器维度	300
Recur-DSP 的伸缩因子 λ	1.60	SGCN-DSP 的伸缩因子 λ	0.80	ζ	0.48
μ_p	0.20	μ_a	0.45	特征 Dropout 率	0.10
优化器	Adam	学习率	1e-5	学习率衰减率	1e-3
提前停止容忍值	20	CoNLL09 小批次样本数	16	UPB 小批次样本数	32

4.3　对比方法

为了验证本文所提出基于动态句法剪枝机制的端到端中文语义角色标注系统的有效性，我们从系统建模方式、编码句法结构知识、句法剪枝三个角度，选择了目前最先进的、具有代表性的语义角色标注系统作为对比模型. 模型的具体介绍如下：

(1) Zhao 等人(2009)[50] 和 Björkelund 等人(2009)[51]. 基于传统的统计模型结合人工构建的离散特征，诸如依存结构信息、短语句法信息等.

(2) Roth 等人(2016)[13]. 将句法依存树路径嵌入表征引入到神经网络模型中以改进中文语义角色标注性能. 该方法采用了流水线方式，默认谓词是给定的.

(3) Marcheggiani 等人(2017)[14]. 利用图卷积网络模型编码句法依存知识作为语义角色标注系统的基本编码器，该模型基于流水线的工作方式.

(4) He 等人(2018)[31]. 提出基于长短时记忆网络的流水线语义角色标注系统. 该方法首先进行人工离线的句法剪枝，再将剪枝后的句法特征进行编码以更好的提升任务性能.

(5) He 等人(2018)[22]. 采用基于 span 图的网络构架从而首次实现了端到端的语义角色标注.

(6) Cai 等人(2018)[21]. 将端到端的语义角色标注任务建模为依存解析任务. 该系统未考虑使用句法结构信息但取得较好的效果.

(7) Li 等人(2019)[19]. 基于 He 等人(2018)[22] 提出的端到端网络架构，利用双仿射打分器来度量谓词-论元对，从而提升端到端的性能. 这里，我们重新实现了他们的系统，并且将句法信息编码进模型.

(8) Xia 等人(2019)[18]. 基于多任务学习框架实现将外部句法结构信息集成到端到端中文语义角色标注过程.

(9) Fei 等人(2020)[43]. 基于图网络构架的端到端模型提出高阶提纯机制，使得模型能够学到全局的特征. 该模型是当前最优的端到端中文语义角色标注模型.

4.4　评估方法

采用精确率(Precision，P)、召回率(Recall，R)和 $F1$ 值来度量实验中各个模型的性能. 无论是流水线模型还是端到端模型，我们统一对比每个模型在论元角色标注(简记为 Arg.)和谓词识别(简记为 Prd.)上的效果. 实验过程中所提模型经过 10 次训练，每次训练以不同的随机种子进行初始化，呈现最终的平均值. 利用 Dan Bikel 随机验证对比器①进行显著性测试. 对比当前最好模型，所提模型经过了显著性的验证，即 $p \leq 0.05$.

5　实验结果与分析

本节详细介绍了所提模型在不同数据集上的实验结果和分析. 首先，通过与基线系统的对比得出相关结论. 然后，通过消融实验分析所提出的方法的各个机制. 最后，对模型的各个功能、优点进行细粒度的验证与分析.

① http://www.cis.upenn.edu/dbikel/software.html\#comparator

5.1 主实验结果

在 CoNLL09 上的结果. 表 3 呈现了各个流水线方法和端到端方法在 CoNLL09 数据集上的实验结果. 通过比较各个方法的结果,可以得出以下几个结论. 首先可以观察到,端到端方法要好于流水线方法,例如,最好的流水线模型(He 等人(2018)[31])比最差的端到端模型(He 等人(2018)[22])在角色论元识别上少了 0.2% 的 $F1$ 值(83.0～82.8). 其次,比较不同的流水线方法,可发现传统的统计模型(Zhao 等人(2009)[50]和 Björkelund 等人(2009)[51])在性能上要低于基于深度学习的模型(Roth 等人(2016)[13]、Marcheggiani 等人(2017)[14]和 He 等人(2018)[31]). 进一步可观察到,采取句法剪枝策略后的方法(He 等人(2018)[31])要比直接编码整体全量的句法结构特征(Roth 等人(2016)[13])的效果更好. 这证实了句法剪枝对语义角色标注任务的有效性.

表 3 模型在 CoNLL09 数据集上的实验结果

	Arg.			Prd.
	P	R	$F1$	$F1$
流水线方法				
Zhao 等人(2009)[50]	80.4	75.2	77.7	—
Björkelund 等人(2009)[51]	82.4	75.1	78.6	—
Roth 等人(2016)[13]	83.2	75.9	79.4	—
Marcheggiani 等人(2017)[14]	84.6	80.4	82.5	—
He 等人(2018)[31]	84.2	81.5	82.8	—
端到端方法				
He 等人(2018)[22]	82.6	83.6	83.0	85.7
Li 等人(2019)[19]	84.9	84.6	84.8	86.9
Xia 等人(2019)[18]	84.6	85.7	85.1	87.2
+BERT	88.0	89.1	88.5	89.6
Fei 等人(2020)[43]	85.7	86.2	85.9	88.6
+BERT	87.4	89.3	88.8	90.3
Recur-DSP	86.2	86.6	86.4	88.8
+BERT	87.8	89.8	89.0	90.6
SGCN-DSP	86.8	87.1	86.9	89.1
+BERT	**88.6**	**90.7**	**90.4**	**90.8**

再具体观察各个端到端方法. 通过对 He 等人(2018)[22]的模型与 Li 等人(2019)[19]的模型进行比较,可以发现:利用句法结构信息的方法在语义角色标注效果(同时包含角色论元识别以及谓词识别)上要优于没有编码句法结构信息的方法. 通过进一步采用更高效的句法融合方式,如 Xia 等人(2019)[18]的多任务学习框架,可以获得更好的效果,达到了 85.1% 的角色论元识别 $F1$ 值,以及 87.2% 的谓词识别 $F1$ 值.

最关键的是,本文所提的基于动态句法剪枝策略的端到端模型 Recur-DSP 和 SGCN-DSP,皆取得了优异的性能,均超过了当前中文端到端语义角色标注最强的最强模型,该模型由 Fei 等人(2020)[43]提出. 这证明执行有效的动态句法剪枝对任务的重要性. 其中可以注意到,所提出的 SGCN-DSP 方案给出了最优的结果,带来了 86.9% 的角色论元识别 $F1$ 值,以及 89.1% 的谓词识别 $F1$ 值. 最后可以发现,结合最近几年流行的预训练模型 BERT[52],各个系统的效果可以得到非常明显的提升. 而本文所提方法依然获得比所有基线模型更优的性能,比如 SGCN-DSP+BERT 取得了 90.4% 的角色论元识别 $F1$ 值,以及 90.8% 的谓词识别 $F1$ 值.

在 UPB 上的结果. 在 UPB 数据集上我们只展示了端到端系统的实验结果,具体的结果如表 4 所示. 各个系统的趋势基本与 CoNLL09 数据集上保持一致,有以下几个观察. 首先,通过比较 Cai 等人(2018)[21]的句法信息无关方法与 Li 等人(2019)[19]的基于句法信息的方法可知,融合句法信息的系统要表现的更好. 其次,本文所提模型带来了最优的性能,SGCN-DSP 取得了 68.7% 的角色论元识别 $F1$ 值,超过目前最强的方法(Fei 等人(2020)[43]的模型). 这再次证明了执行句法动态剪枝的重要性. 最后,通过整合预训练模型 BERT,任务性能得到进一步的提升. 上述实验结果和分析证实本文所提模型的有效性.

表 4 端到端系统在 UPB 数据集上对语义角色论元抽取的表现

	P	R	$F1$
He 等人(2018)[22]	64.8	65.3	64.9
Cai 等人(2018)[21]	65.0	66.4	65.8
Li 等人(2019)[19]	65.4	67.2	66.0
Fei 等人(2020)[43]	67.5	68.8	67.9
+BERT	70.0	73.0	72.4
Recur-DSP	67.8	68.5	68.0
+BERT	70.4	72.8	71.7
SGCN-DSP	68.2	69.1	68.7
+BERT	**71.9**	**73.3**	**72.7**

5.2 混淆矩阵分析

为更加直观地理解所提出的两种动态句法裁剪机制 Recur-DSP 与 SGCN-DSP 与所对应的基础原型句法编码器(即 Recur 与 SGCN)的提升能力的差异性,对这两套句法编码器模型关于语义角色论元抽取性能的混淆矩阵进行对比分析. 本实验基于 CoNLL09 的测试集,混淆矩阵结果如图 6 所示. 通过混淆矩阵热力图的颜色深浅可以推出,句法编码器 Recur 与 SGCN 结合了所提出的动态句法裁剪机制后得到了非常明显的性能提升. 可看到在原始句法编码器 Recur 与 SGCN 中大量错误预测的论元角色标

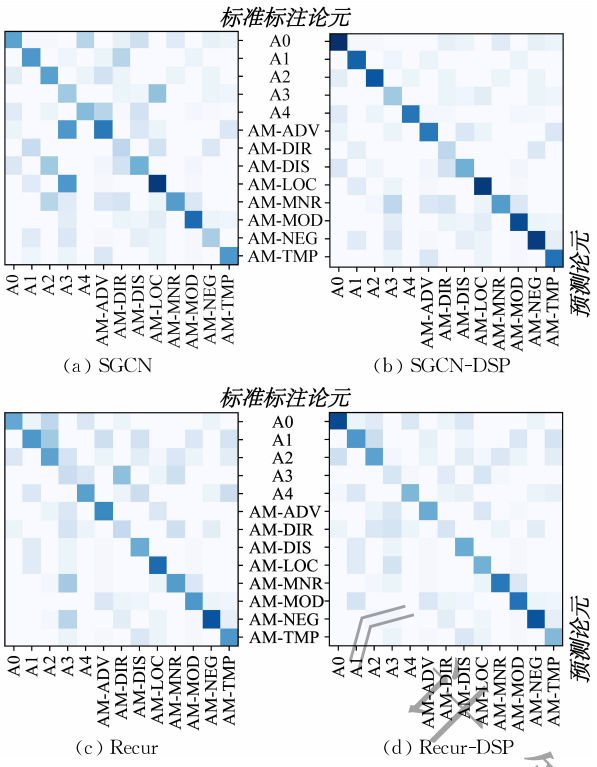

图 6　语义角色论元抽取性能的混淆矩阵可视化的对比(颜色越深表示数量越多)

签在进行了动态句法裁剪后得到了纠正并成为正确的预测. 比如,Recur 与 SGCN 模型较多地将 *A*0 预测为其他标签,而结合了动态句法裁剪后,能够极大减少这种错误. 另外可以观察到,SGCN-DSP 比 Recur-DSP 具有更强的能力,所预测的正确结果更多.

5.3　消融实验

本节基于消融实验来进一步细粒度地分析和验证本文所提方法的在各个方面以及各个机制的优缺点. 实验基于 CoNLL09 的测试集,结果如表 5 所示.

表 5　消融实验结果(*F*1 值)

	Recur-DSP		SGCN-DSP	
	Arg.	Prd.	Arg.	Prd.
完整模型	**86.4**	**88.8**	**86.9**	**89.1**
输入特征				
摘除预训练词向量	86.0	88.0	86.3	88.4
摘除词性标注特征	85.2	87.0	85.8	87.7
摘除字信息特征	86.2	88.4	86.6	88.9
表征编码层				
替换为双向 LSTM	85.1	87.2	85.7	87.8
句法融合层				
摘除句法特征	84.0	85.4	84.0	85.4
摘除依存句法标签	—	—	84.8	86.7
摘除动态句法剪枝	84.6	86.5	85.1	87.0
解码层				
摘除一元打分器	85.9	88.0	86.4	88.4
摘除二元打分器	80.7	81.8	81.0	82.1
训练优化				
摘除谓词缓存机制	86.2	88.5	86.7	88.9
摘除论元缓存机制	86.1	88.6	86.6	89.0

(1) 输入特征. 首先观察模型的输入特征对模型的影响. 具体地,依次摘除掉预训练词向量(替换为随机初始化的词向量)、词性标注特征以及字信息特征,可看到模型在任务上的性能都有一定程度的降低. 通过对比可知这三种特征的重要性由强至弱的排列为词性标注特征>字信息特征>预训练词向量.

(2) 表征编码层. 若考虑将表征编码层的 Transformer 模型替换为双向长短时记忆网络模型,则可看到 SGCN-DSP 与 Recur-DSP 两套句法剪枝方案在谓词识别和角色论元识别两个子任务上都出现了较为明显的性能下降. 这验证了采用基于自注意力机制的 Transformer 模型作为上下文表征编码层的有效性. 主要原因在于,Transformer 能充分捕获到句子中词之间的互相作用,提供更为丰富的句子语义表征[53-54].

(3) 句法融合层. 进一步考察句法融合层的各个机制. 首先考虑直接去除掉对于句法特征的融合,即直接让表征编码层的输出作为每个词 w_i 所对应的上下文表征 $\boldsymbol{s}_i$. 可以观察到 SGCN-DSP 与 Recur-DSP 两个系统都出现了大幅的性能下降①,具体地,角色论元识别 *F*1 值下降 2.9%(86.9～84.0),谓词识别 *F*1 值下降 3.7%(89.1～85.4). 若去掉 SGCN-DSP 系统中对依存句法标签特征 $\boldsymbol{\pi}^{\curvearrowright}_{i,j}$ 的编码,即只考虑用图卷积网络模型编码依存句法的连接结构,可观察到其性能同样有一定程度的下降. 这证明了将依存句法标签特征与依存弧进行统一建模编码的重要性. 接着,我们考虑去掉动态句法剪枝机制,可以看到 SGCN-DSP 与 Recur-DSP 两个系统都出现了非常明显的性能下降,这再次验证了本文所提的动态句法剪枝机制对语义角色标注任务的有效性.

(4) 解码层. 在解码阶段,我们采用了结合一元打分器与二元打分器的策略. 若将一元打分器为谓词与论元的打分值从 $\Phi(p,a,l)$(式(13))中移除,可看到结果出现比较明显的下降. 而如果直接去掉二元打分器,可看到整体色性能下降幅度非常大. 其原因在于二元打分器直接度量了“谓词-论元-角色标签”三元组之间合理配对的可能性,是解码层的核心部件.

(5) 训练优化. 这里,我们分别摘除掉为优化训

① SGCN-DSP 与 Recur-DSP 两个系统降级为相同的句法无关系统.

《计算机学报》

练而提出的谓词缓存容器与论元缓存容器，可以观察到 SGCN-DSP 与 Recur-DSP 两个系统在性能上出现了微弱的下降. 再次证实了所设计的缓存容器对模型训练的有效性.

5.4 句法融合模块层数分析

我们进一步研究不同层数的句法融合模块对最终性能的影响. 针对 SGCN-DSP 与 Recur-DSP 两种系统我们验证在不同句法融合模块层数下的关于论元角色标签抽取性能. 图 7 展示了实验结果. 可观察到，并非层数越多性能越好. 其中，Recur-DSP 采用 2 层的句法融合模块时表现最佳；而 SGCN-DSP 采用 3 层的句法融合模块时表现最佳. 可能的原因在于，太少的层数会导致欠充分的句法信息的传播，而太多层数的句法编码反而会带来过多的参数，从而导致过拟合. 并且在两个数据集上 SGCN-DSP 与 Recur-DSP 有着相似的规律. 另外，之所以 SGCN-DSP 需要更多层数的句法融合，其原因主要在于两点：(1) 编码句法标签信息需要额外的参数开销. (2) 图卷积神经网络基于图模型的工作方式需要更多的层数来充分传播信息[55-57].

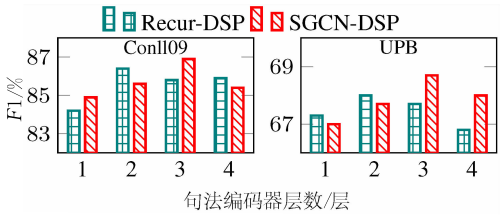

图 7 不同层数下的 SGCN-DSP 与 Recur-DSP 系统在两个数据集上关于论元角色标签抽取的性能

5.5 谓词论元距离的影响

语义角色标注任务中的一个常见问题为长距离依存问题[23,43]，即当谓词与角色论元之间的距离增大后，对角色论元的识别将会造成越来越大的困难. 已有的相关研究表明，外部句法信息可以有效地缓解长距离依存问题，因为直观上理解，依存句法结构的依存弧可以跨越较长的距离为谓词与论元之间提供直接的线索，突破句子的词顺序的限制. 因此我们探索进行句法裁剪是否可进一步帮助解决长距离依存问题，这里基于 CoNLL09 数据集进行实验. 具体地，选择：(1) He 等人(2018)[22] 的句法无关方法，(2) Li 等人(2019)[19] 的融合了全量句法信息的模型作为两个基线方法，同本文提出的 SGCN-DSP 和 Recur-DSP 模型进行对比. 图 8 显示了各个模型在不同谓词-论元对间距离下论元角色抽取的表现.

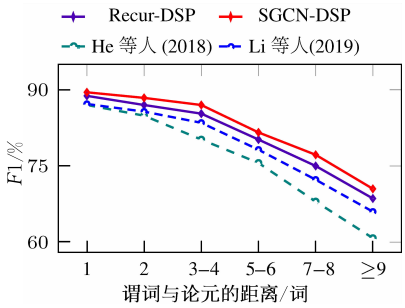

图 8 不同谓词-论元对间距离下论元角色抽取的表现

首先可以明显观察到，随着谓词与论元的距离的增大，对论元角色抽取的性能开始逐渐下降. 而通过对比 He 等人(2018)[22] 与 Li 等人(2019)[19] 的模型可知，外部句法信息可以有效地缓解长距离依存问题. 具体地，当距离逐渐增加至 9 时，基于句法信息的 Li 等人(2019)[19] 的模型亦能表现出较为不错的性能，而 He 等人(2018)[22] 此时的性能下降非常明显. 最关键的是，本文所提出的 SGCN-DSP 与 Recur-DSP 方法无论在谓词与论元有多远距离下都能表现出最好的效果. 即使面对超远距离的谓词-论元对，我们的模型都要强于 Li 等人(2019)[19] 的模型. 这证明了动态句法裁剪能一定程度上解决长距离依存问题. 另外可观察到 SGCN-DSP 模型要优于 Recur-DSP 模型.

5.6 动态句法裁剪分析

本节深入分析动态句法裁剪对基于句法信息的语义角色标注系统的影响. 由于在式(22)为 Recur-DSP 与 SGCN-DSP 各自装配了一个伸缩控制因子 λ 以灵活地控制动态句法剪枝机制对依存结构信息的裁剪力度. 现考虑人工调整 λ 为不同取值下的实验结果，这里基于 CoNLL09 数据集进行实验. 图 9 与图 10 分别展现了不同伸缩控制因子下 Recur-DSP 模型以及 SGCN-DSP 模型中的句法结构信息的覆盖率以及对应的论元角色抽取的结果.

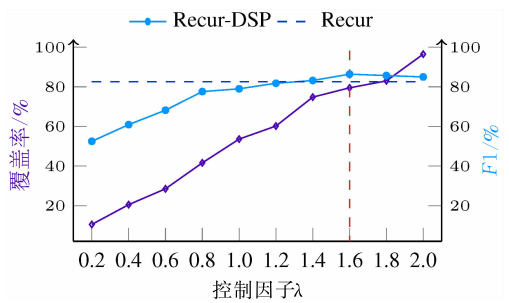

图 9 不同伸缩控制因子 λ 下 Recur-DSP 模型中的句法结构信息的覆盖率和对应的论元角色抽取的表现

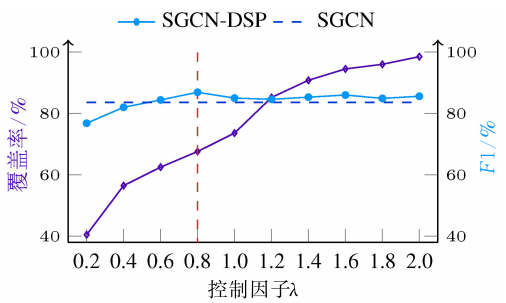

图 10　不同伸缩控制因子 λ 下 SGCN-DSP 模型中的句法结构信息的覆盖率和对应的论元角色抽取的表现

首先可以看到，随着控制因子的值的提高，剪枝力度会减弱，从而使得句法结构信息的覆盖率得以提高. 当 λ 接近于 2 时，Recur-DSP 与 SGCN-DSP 皆达到了接近 100% 的句法信息覆盖率. 而 λ 的提高并不能为论元角色抽取的效果带来持续地提升. 对于 Recur-DSP，当 $\lambda=1.6$ 时，任务效果达到最佳；而对于 SGCN-DSP，当 $\lambda=0.8$ 时，任务效果达到最佳. 究其原因，对于基于树结构的 Recur 模型，节点之间的连接更为稀疏，使得系统学习偏向于让句法剪枝门控为开（$p_{i,1}$），即降低句法裁剪力度. 然而较弱的裁剪力度使得模型编码了更多的与任务帮助无关的句法枝干信息，导致裁剪低效. 当伸缩控制因子 λ 提升到 1.6 时既能有效地保证句法裁剪达到一定的数量，又能保证裁剪后保留下来的句法信息最有利于任务.

而对于 SGCN-DSP，由于将句法结构建模为图结构，加上为每个节点人工增加的自环结构，以及双向联通结构，相对 Recur-DSP 而言，图结构更为稠密，自然系统学习到的句法剪枝门控为开（$p_{i,1}$）的概率趋向于偏大. 这使得模型会加大裁剪力度，导致过多的句法信息被排除掉. 当 λ 降低到 0.8 时，恰好可以让 SGCN-DSP 在句法覆盖率与任务性能上取得最优平衡. 另外还可观察到，去掉动态句法剪枝机制后，Recur 与 SGCN 的论元角色抽取性能弱于对应的最佳地结合了动态句法剪枝机制的模型，相当于完整全量的句法结构信息帮助下的表现，而全量信息必然会带来一些无关信息等噪音弱化了效果.

为了进一步直观地验证“动态句法剪枝策略与静态的句法剪枝或者不进行句法剪枝相比的结果”，我们进行了一个更为公平的实验. 我们考虑基于所提出的两类句法编码器，在统一的端到端框架下，分别实现动态剪枝、静态剪枝、不剪枝这三种情况. 表 6 基于 CoNLL09 数据集在公平的实验条件下对比了各个剪枝策略分别在论元识别与谓词检测子任务上的 $F1$ 值. 可以看到，实现了动态剪枝后的效果在总体上保持了最佳水平，其次是静态的剪枝，最后是不进行剪枝的方法. 但是，在 SGCN 模型下，不进行句法剪枝的效果反而强于利用了静态剪枝的方法. 这意味着，由于 SGCN 编码了句法标签信息，其反而不适合进行静态离线式的句法剪枝. 其背后深层的原因在于，这会潜在地引起句法信息丢失问题.

表 6　不同剪枝策略的效果（*F1*）对比

	动态剪枝	静态剪枝	不剪枝
Arg.			
Recur	86.4	85.1	84.6
SGCN	**86.9**	84.8	85.1
Prd.			
Recur	88.8	86.8	86.5
SGCN	**89.1**	86.6	87.0

5.7　语义角色结构与句法结构依赖关系

由于所提的 SGCN-DSP 系统同时编码依存树的依存弧结构和对应的标签信息，从而带来了更好的任务性能. 这里进一步探索了语义角色结构与句法结构的依赖关系. 通过分析系统对语义角色标注的预测结果与句法依存图邻接矩阵 $\boldsymbol{\alpha}_{i,j}$（式（19））的对应关系，进而可以确定语义角色结构与句法结构的依赖关系. 我们将 SGCN-DSP 所发现的语义角色结构与句法结构的依赖关系以热力图的方式进行可视化，如图 11 所示.

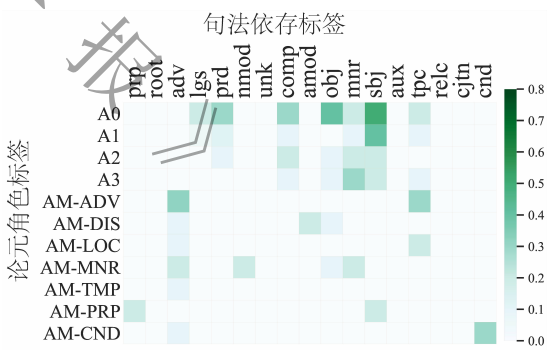

图 11　SGCN-DSP 系统发现的语义角色结构与句法结构的依赖

首先可以推断出，SGCN-DSP 系统可正确地学习到语义角色结构与句法结构的依赖关系，这也为 SGCN-DSP 表现出更强的任务性能提供了直接的证据. 另外，通过该可视化图像可观察到，并非所有的句法依存结构（依存对）对语义角色标注任务都有帮助作用. 比如，“*aux*”与“*relc*”等句法结构对任何论元角色标签都无明显的帮助效果. 另外，这些带有“*sbj*”标签的句法结构对核心语义角色标签“*A0*”与“*A1*”的贡献最大；而修饰性语义角色标签如“*AM-ADV*”与

《计算机学报》

句法标签“*adv*”,“*AM-CND*”则与带有“*cnd*”标签的句法结构有着最为密切的联系.这证明,一方面句法标签信息对语义角色标注任务有着非常明显的帮助作用,另一方面,进行动态句法裁剪帮助对语义角色标注任务进一步的性能提升极其关键.同时,这种语义与依法结构信息的依赖关系能够为相关研究,诸如基于句法信息的无监督语义角色标注任务,提供重要的理论支撑.

5.8 个案分析

最后,为了更加直观地理解本文所提出的动态句法剪枝机制的优势,我们基于测试数据集中随机挑选的一个实例,分别用四种不同的方法对其进行预测,并通过式(19)中的图邻接矩阵 $\boldsymbol{\alpha}_{i,j}$ 解析其句子中各个词在句法结构中所激活的状态.图 12 分别展示了(a)无句法剪枝机制的 GCN 模型的结果、(b)无句法剪枝机制的 SGCN 模型的结果、(c)基于静态句法剪枝机制的 SGCN 模型的结果以及(d)基于动态句法剪枝机制的 SGCN 模型的结果.通过图中所展现的现象,我们主要可以得到以下三点观察与发现.

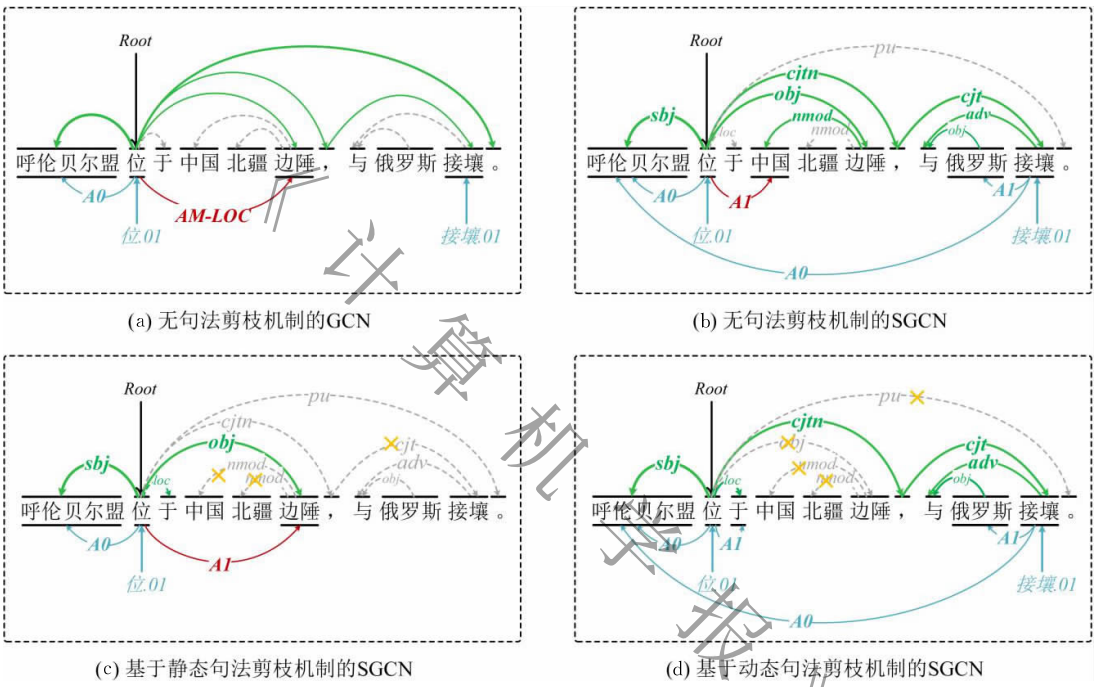

图 12 个案分析结果(可视化中,绿色的弧表示在图模型中获得激活,绿色越深表示越高的权重;灰色的弧表示未被图模型激活.黄色的叉号表示该句法弧被裁剪过.蓝色与红色分别表示正确与错误的任务预测结果.其中我们所提出的方法,即(d)基于动态句法剪枝机制的 SGCN,预测的结果全部正确且完整)

(1)通过对比图 12(a)和(b)可以明显观察到,将句法标签信息与弧信息一并建模到图结构中可进一步帮助正确地判断任务所依赖的句法边的情况.比如,在图 12(b)中,SGCN 学到了对于带有“*pu*”类型的依存弧,即表示句子结尾的元素,会降低其被激活的概率.同时,相比 GCN,SGCN 还可以通过句法标签信息学到依存弧的多跳激活,比如,模型可使得“俄罗斯”(*A*1)被正确解析出来.

(2)通过对比图 12(c)和(d)可以发现,基于动态句法剪枝机制的 SGCN 结果要远优于基于静态剪枝机制的结果.通过图中可视化可知其原因在于,静态句法剪枝会造成关键信息的丢失,比如,图 12(c)中由于离线地将“*cjt*”类型的依存弧(1 阶剪枝)裁剪掉,导致第二个谓词“接壤”以及其对应的论元的预测造成全部丢失.相比之下,动态句法剪枝机制灵活地决定哪些句法结构对当前的预测有帮助,从而将其保留,并促进获得正确的预测结果.

(3)通过对比图 12(b)和(c)可知,对于 SGCN 模型,实际上静态剪枝的效果要差于无剪枝的效果.这表明,由于 SGCN 利用了句法标签信息,其已经获得了更高效的句法特征选择能力.

6 总结与展望

本文提出基于动态句法剪枝机制的端到端神经网络模型,并将其成功用于中文语义角色标注.本文

提出了两种动态剪枝方法：基于递归神经网络模型的动态句法剪枝机制和基于带句法标签的图卷积网络模型的句法剪枝机制. 基于中文语义角色标注的基准数据集的实验结果证实本文所提两种方法均超越了当前最先进的基线系统，成为目前最佳的中文语义角色标注系统. 通过后续大量的实验分析验证了本文所提出的动态句法裁剪机制对任务性能的提升作用. 同时，通过为句法依存树的依存弧结构以及对应的标签进行统一建模，可以发现语义角色结构与句法结构之间的依赖关系.

未来工作中将考虑以下几个方面的探索. 首先，除了依存句法树结构外，可考虑融合其他类型的句法结构信息，比如句法成分树. 其次，考虑采用不同的句法编码器进行句法信息的编码，如基于自注意力机制的 Transformer 模型. 最后，本文没有考虑动态句法裁剪过程中所导致的潜在的句法子结构的不连通问题，即进行动态句法裁剪后将使得完整的句法树分解. 未来工作可考虑完善句法裁剪机制，使得剪枝后的结构依然保持全连通.

致　谢　衷心感谢各位审稿专家对本文工作的指导!

参　考　文　献

[1] Christensen J, Mausam, Soderland S, et al. An analysis of open information extraction based on semantic role labeling//Proceedings of the International Conference on Knowledge Capture. Banff, Canada, 2011: 113-120

[2] Bastianelli E, Castellucci G, Croce D, et al. Textual inference and meaning representation in human robot interaction//Proceedings of the Joint Symposium on Semantic Processing. Trento, Italy, 2013: 65-69

[3] Shen D, Lapata M. Using semantic roles to improve question answering//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Prague, Czech Republic, 2007: 12-21

[4] Berant J, Chou A, Frostig R, et al. Semantic parsing on Freebase from question-answer pairs//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Seattle, USA, 2013: 1533-1544

[5] Guo S, Guan Y, Li R, et al. Incorporating syntax and frame semantics in neural network for machine reading comprehension //Proceedings of the 28th International Conference on Computational Linguistics. Barcelona, Spain (Online), 2020: 2635-2641

[6] Xiong D, Zhang M, Li H. Modeling the translation of predicate-argument structure for SMT//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Jeju Island, Korea, 2012: 902-911

[7] Shi C, Liu S, Ren S, et al. Knowledge-based semantic embedding for machine translation//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Berlin, Germany, 2016: 2245-2254

[8] Zhang Z, Wu Y, Zhao H, et al. Semantics-aware BERT for language understanding//Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence. New York, USA, 2020: 9628-9635

[9] Pradhan S, Ward W, Hacioglu K, et al. Semantic role labeling using different syntactic views//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Michigan, USA, 2005: 581-588

[10] Punyakanok V, Roth D, Yih W. The importance of syntactic parsing and inference in semantic role labeling. Computational Linguistics, 2008, 34(2): 257-287

[11] Zhao H, Chen W, Kazama J, et al. Multilingual dependency learning: Exploiting rich features for tagging syntactic and semantic dependencies//Proceedings of the Conference on Computational Natural Language Learning. Boulder, USA, 2009: 61-66

[12] Fitzgerald N, Täckström O, Ganchev K, et al. Semantic role labeling with neural network factors//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Lisbon, Portugal, 2015: 960-970

[13] Roth M, Lapata M. Neural semantic role labeling with dependency path embeddings//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Berlin, Germany, 2016: 1192-1202

[14] Marcheggiani D, Titov I. Encoding sentences with graph convolutional networks for semantic role labeling//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark, 2017: 1506-1515

[15] Strubell E, Verga P, Andor D, et al. Linguistically-informed self-attention for semantic role labeling//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Brussels, Belgium, 2018: 5027-5038

[16] Wang Ming-Xuan, Liu Qun. A simple and effective deep model for semantic role labeling. Journal of Chinese Information Processing, 2018, 32(2): 50-57(in Chinese)
(王明轩，刘群. 基于深度神经网络的语义角色标注. 中文信息学报，2018，32(2)：50-57)

[17] Fei H, Zhang M, Ji D. Cross-lingual semantic role labeling with high-quality translated training corpus//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Online Conference, 2020: 7014-7026

[18] Xia Q, Li Z, Zhang M. A syntax-aware multi-task learning framework for Chinese semantic role labeling//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Hong Kong, China, 2019: 5382-5392

《计算机学报》

[19] Li Z, He S, Zhao H, et al. Dependency or span, end-to-end uniform semantic role labeling//Proceedings of the AAAI Conference on Artificial Intelligence. Honolulu, USA, 2019: 6730-6737

[20] Scheible C. An evaluation of predicate argument clustering using pseudo-disambiguation//Proceedings of the International Conference on Language Resources and Evaluation. Valletta, Malta, 2010: 1187-1194

[21] Cai J, He S, Li Z, et al. A full end-to-end semantic role labeler, syntactic-agnostic over syntactic-aware?//Proceedings of the International Conference on Computational Linguistics. Santa Fe, USA, 2018: 2753-2765

[22] He L, Lee K, Levy O, et al. Jointly predicting predicates and arguments in neural semantic role labeling//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Melbourne, Australia, 2018: 364-369

[23] Fei H, Zhang M, Li F, et al. Cross-lingual semantic role labeling with model transfer. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2020, 28: 2427-2437

[24] Yuan Xiao-Hong, Wang Hong-Ling, Wang Bu-Kang, et al. Dependency-based semantic role labeling in Chinese language for nominal predicate. Computer Applications and Software, 2011, 28(5): 31-33(in Chinese)
(袁晓虹, 王红玲, 王步康等. 基于依存关系的中文名词性谓词语义角色标注研究. 计算机应用与软件, 2011, 28(5): 31-33)

[25] Marcheggiani D, Frolov A, Titov I. A simple and accurate syntax-agnostic neural model for dependency-based semantic role labeling//Proceedings of the Conference on Computational Natural Language Learning. Vancouver, Canada, 2017: 411-420

[26] Zhang Y, Wang R, Si L. Syntax-enhanced self-attention-based semantic role labeling//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Hong Kong, China, 2019: 616-626

[27] Yang Feng-Ling, Zhou Qiao-Li, Cai Dong-Feng, et al. Semantic role labeling combined with phrase structure parsing. Journal of Chinese Information Processing, 2018, 32(6): 1-11(in Chinese)
(杨凤玲, 周俏丽, 蔡东风等. 结合短语结构句法的语义角色标注. 中文信息学报, 2018, 32(6): 1-11)

[28] Li Z, He S, Cai J, et al. A unified syntax-aware framework for semantic role labeling//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Brussels, Belgium, 2018: 2401-2411

[29] Xia Q, Li Z, Zhang M, et al. Syntax-aware neural semantic role labeling//Proceedings of the AAAI Conference on Artificial Intelligence. Honolulu, USA, 2019: 7305-7313

[30] Marcheggiani D, Titov I. Graph convolutions over constituent trees for syntax-aware semantic role labeling//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Virtual Event, 2020: 3915-3928

[31] He S, Li Z, Zhao H, et al. Syntax for semantic role labeling, to be, or not to be//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Melbourne, Australia, 2018: 2061-2071

[32] Yu B, Mengge X, Zhang Z, et al. Learning to prune dependency trees with rethinking for neural relation extraction//Proceedings of the 28th International Conference on Computational Linguistics. Barcelona, Spain(Online), 2020: 3842-3852

[33] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need//Proceedings of the Advances in Conference on Neural Information Processing Systems. Long Beach, USA, 2017: 5998-6008

[34] Hajič J, Ciaramita M, Johansson R, et al. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages//Proceedings of the Conference on Computational Natural Language Learning. Boulder, USA, 2009: 1-18

[35] Akbik A, Chiticariu L, Danilevsky M, et al. Generating high quality proposition banks for multilingual semantic role labeling// Proceedings of the Annual Meeting of the Association for Computational Linguistics. Beijing, China, 2015: 397-407

[36] Akbik A, Li Y. Polyglot: Multilingual semantic role labeling with unified labels//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Berlin, Germany, 2016: 1-6

[37] Gildea D, Jurafsky D. Automatic labeling of semantic roles//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Hong Kong, China, 2000: 512-520

[38] Baker C F, Fillmore C J, Lowe J B. The Berkeley FrameNet project//Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics. Montreal, Canada, 1998: 86-90

[39] Palmer M, Kingsbury P R, Gildea D. The proposition bank: An annotated corpus of semantic roles. Computational Linguistics, 2005, 31(1): 71-106

[40] Bach E, Harms R. Universals in linguistic theory. Indogermanische Forschungen, 1970, 75(1): 230-242

[41] Marcus M P, Santorini B, Marcinkiewicz M A. Building a large annotated corpus of English: The Penn Treebank. Computational Linguistics, 1993, 19(2): 313-330

[42] Kipper K, Dang H T, Palmer M S. Class-based construction of a verb lexicon//Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence. Austin, USA, 2000: 691-696

[43] Fei H, Ren Y, Ji D. High-order refining for end-to-end

Chinese semantic role labeling//Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing. Suzhou, China, 2020: 100-105

[44] Sun W, Sui Z, Wang M, et al. Chinese semantic role labeling with shallow parsing//Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. Singapore, 2009: 1475-1483

[45] Wang Z, Jiang T, Chang B, et al. Chinese semantic role labeling with bidirectional recurrent neural networks//Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Lisbon, Portugal, 2015: 1626-1631

[46] Xia Q, Sha L, Chang B, et al. A progressive learning approach to Chinese SRL using heterogeneous data//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver, Canada, 2017: 2069-2077

[47] Dozat T, Manning C D. Deep biaffine attention for neural dependency parsing//Proceedings of the International Conference on Learning Representations. Toulon, France, 2017: 1-10

[48] Tai K S, Socher R, Manning C D. Improved semantic representations from tree-structured long short-term memory networks//Proceedings of the Annual Meeting of the Association for Computational Linguistics. Beijing, China, 2015: 1556-1566

[49] Gumbel E J. Statistical theory of extreme values and some practical applications: A series of lectures: Volume 33. Washington, USA: US Government Printing Office, 1954

[50] Zhao H, Chen W, Kit C. Semantic dependency parsing of NomBank and PropBank: An efficient integrated approach via a large-scale feature selection//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Singapore, 2009: 30-39

[51] Björkelund A, Hafdell L, Nugues P. Multilingual semantic role labeling//Proceedings of the Conference on Computational Natural Language Learning. Boulder, USA, 2009: 43-48

[52] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding//Proceedings of the North American Chapter of the Association for Computational Linguistics. Minneapolis, USA, 2019: 4171-4186

[53] Lei T, Zhang Y, Márquez L, et al. High-order low-rank tensors for semantic role labeling//Proceedings of the North American Chapter of the Association for Computational Linguistics. Denver, USA, 2015: 1150-1160

[54] Chen X, Lyu C, Titov I. Capturing argument interaction in semantic role labeling with capsule networks//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Hong Kong, China, 2019: 5415-5425

[55] Lyu C, Cohen S B, Titov I. Semantic role labeling with iterative structure refinement//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Hong Kong, China, 2019: 1071-1082

[56] Cai R, Lapata M. Semi-supervised semantic role labeling with cross-view training//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Hong Kong, China, 2019: 1018-1027

[57] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation, 1997, 9(8): 1735-1780

FEI Hao, Ph. D. candidate. His research interests include natural language processing, and semantics-syntax structure parsing.

JI Dong-Hong, Ph. D., professor. His research interests include natural language processing, machine learning and data mining.

REN Ya-Feng, Ph. D., associate professor. His research interests include natural language processing and data mining.

Background

Semantic role labeling (SRL) is a shallow semantic parsing task, which has received extensive research attention within recent years in natural language processing (NLP). The SRL task aims to extract the *predicate-argument* structure of a given sentence, which can facilitate many NLP tasks such as information extraction, question answer and read comprehension, etc. A large number of studies have been proposed for the task, and the existing studies can be divided into two main categories: machine learning methods with hand-crafted discrete features and deep learning methods with automatic distributed features. Early studies largely separate SRL into two individual subtasks, i. e., predicate

disambiguation and argument role labeling. More recently, great efforts have been paid for constructing various end-to-end SRL systems, solving two pipeline steps in one shot via one unified model.

Many studies show that integrating external syntactic features, such as syntactic dependency trees, are highly important for the SRL task. So designing a neural model, which can capture syntactic features effectively, has becoming a heated research topic. Recently, He et al. (2018) find that only a part of syntactic structure information can offer valuable information for the SRL task, which calls for pruning the syntactic structure features. However, the existing research adopts the offline syntactic pruning strategy, which inevitably leads to either the loss of key syntactic information or the weakening of pruning effectiveness.

In this paper, we propose a novel neural network model for the end-to-end Chinese SRL based on a dynamic syntactic pruning mechanism. Specifically, we propose two frameworks with different syntax encoding models and dynamic syntactic pruning strategies. We compare our methods with the current best-performing systems on multiple Chinese SRL benchmarks, and our proposed methods outperform the current best system by a large margin, giving the state-of-the-art performance. In-depth analysis shows that the proposed dynamic syntax pruning mechanism can greatly improve the performance of this task. Specifically, our proposed model SGCN-DSP achieves 86.9% $F1$ score in argument role labeling and 89.1% $F1$ score in predicate identification based on the CoNLL09 dataset. By integrating the current pre-trained language model BERT, the task performance can be further improved. The proposed SGCN-DSP gives 90.4% $F1$ score in the argument role labeling, and 90.8% $F1$ score in predicate identification, respectively.

This work is supported by the National Key Research and Development Program of China (2017YFC1200500), the National Natural Science Foundation of China (61772378, 61702121) and the Science and Technology Project of Guangzhou (202102020607).

《计算机学报》