

基于图神经网络的汉语依存分析和语义组合计算联合模型

汪凯, 刘明童, 陈圆梦, 张玉洁[†], 徐金安, 陈钰枫
北京交通大学 计算机与信息技术学院, 北京 100044
[†] 通讯作者, E-mail:yjzhang@bjtu.edu.cn

摘要

组合原则表明句子的语义由其构成成分的语义按照一定规则组合而成, 由此基于句法结构的语义组合计算一直是一个重要的探索方向, 其中采用树结构的组合计算方法最具有代表性(Tai et al., 2015)。但是该方法难以应用于大规模数据处理, 主要问题是其语义组合的顺序依赖于具体树的结构, 无法实现并行处理。本文提出一种基于图的依存句法分析和语义组合计算的联合框架, 并借助复述识别任务训练语义组合模型和句法分析模型。一方面图模型可以在训练和预测阶段采用并行处理, 极大缩短计算时间; 另一方面联合句法分析的语义组合框架不必依赖外部句法分析器, 同时两个任务的联合学习可使语义表示同时学习句法结构和语义的上下文信息。我们在公开汉语复述识别数据集LCQMC(Liu et al., 2018)上进行评测, 实验结果显示准确率接近树结构组合方法, 达到79.54%, 而预测速度提升高达30倍。

关键词: 句法分析; 语义组合; 图神经网络; 复述识别

Joint Learning Chinese Dependency Parsing and Semantic Composition based on Graph Neural Network

Kai Wang, Mingtong Liu, Yuanmeng Chen, Yujie Zhang[†]
Jinan Xu, Yufeng Chen

School of Computer and Information Technology, Beijing Jiaotong University
Beijing 10004

[†]Corresponding Author, E-mail:yjzhang@bjtu.edu.cn

Abstract

The semantics of a sentence is composed of the meaning of its constituent components and the combination method. Therefore, syntax-based semantic composition has always been an important research direction in NLP. The semantic composition method using tree structure has become the most representative method(Tai et al., 2015). However, such methods are difficult to be applied to large-scale data. The main problem is that the order of its semantic composition depends on the structure of the specific tree, and parallel computation cannot be supported. In this paper, we present a joint framework for graph-based dependency parsing and semantic composition. The model does not need to rely on an external syntax parser for providing structural information, and the semantic composition method based on graph neural network can support parallel computation, which greatly reduces the computation time. Moreover, the joint learning of two tasks enables the model to learn the syntactic structure and semantic contextual information. Experimental results on LCQMC(Liu et al., 2018) dataset show that the

国家自然科学基金 (61876198,61976015,61976016) 资助

accuracy is close to the tree-based semantics composition method, reaching 79.54%, and the prediction speed is increased by up to 30 times.

Keywords: Dependency Parsing , Semantic Composition , Graph Neural Network , Paraphrase Identification

1 引言

深度神经网络技术为自然语言处理发展带来崭新建模方式和性能上的巨大提升，成为主流的研究方法，其中语义表示是研究热点之一。已有研究表明有效的语义组合计算模型，如LSTM, CNN, Tree-LSTM等神经网络模型，可以提升自然语言处理应用的性能，如：机器翻译(Callison-Burch et al., 2006)、情感分析(Tai et al., 2015)、复述识别(Fan et al., 2018)、自然语言推理(Mou et al., 2016)等。

基于序列化结构的语义组合计算方法简单有效，被广泛采用(Mueller and Thyagarajan, 2016)，但是，这种方法没有考虑句法结构信息，难以捕获词序完全相同句法结构不同的句子之间的差异。比如句子“放弃美丽的女人让人心碎。”，可以有两种句法结构，如图1所示。在图1(a)的句法结构中，“美丽”作为形容词修饰“女人”，“放弃”的对象是“美丽的女人”；在图1(b)的句法结构中，“美丽”作为名词，“放弃”的对象是“美丽”。由此可见，句法结构决定句子语义，句子的语义表示应该考虑其句法结构。

随后，研究人员开始关注基于句法结构的语义组合计算方法(Tai et al., 2015; Chen et al., 2017; Mou et al., 2016)。组合原则表明句子的语义由其构成成分的语义按照一定规则组合而成，由此根据句法结构进行语义组合计算一直是一个重要的探索方向。在基于句法结构的语义组合计算方法中，采用树结构的组合计算方法最具有代表性，其中以Tai et al. (2015)提出的Tree-LSTM最具有代表性。这些方法在给定的一棵句法树上，从叶子节点开始，语义信息自底向上传递，最终在树的根节点获得句子的语义表示。树结构的语义组合方法虽然建模了单词在句法结构上的语义修饰关系，但和序列化方法相比，模型受句法树规定的语义组合顺序的限制，无法并行计算以支持批处理，另外由于需要额外的句法分析器，模型处理繁琐且计算效率低，难以大规模应用到自然语言处理各项任务中。

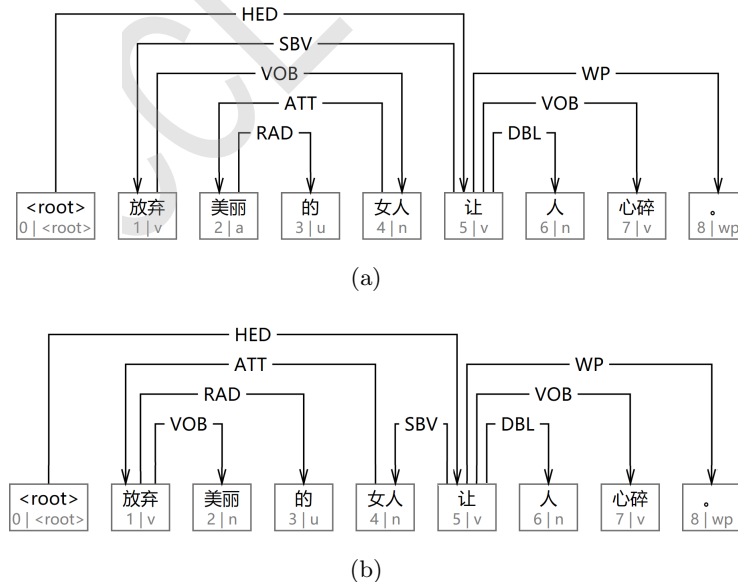


图 1: 相同句子的不同依存句法结构

为了解决上述问题，本文主要针对基于句法结构的语义组合计算方法展开研究，提出一种基于图的依存句法分析和利用图神经网络语义组合计算的联合模型。考虑依存结构描述了反映

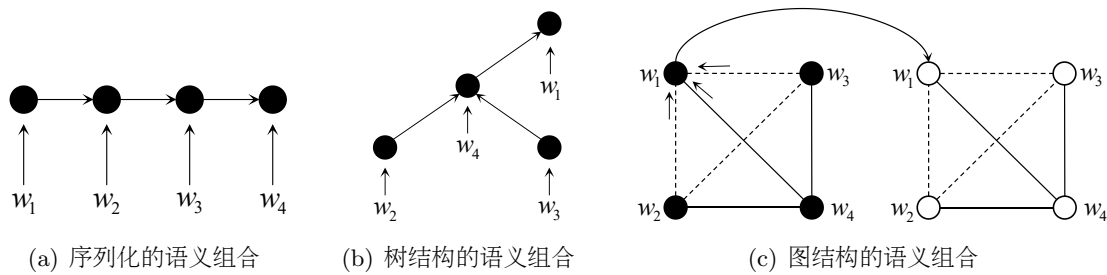


图 2: 不同的语义组合计算方法

单词间语义修饰关系的依存关系，本文采用依存结构信息指导语义组合计算。我们设计基于图的依存句法分析方法，用于生成带有概率的依存关系结构图，包含所有单词对间的有向依存弧的概率。然后，我们使用依存关系概率作为图神经网络中信息传递的权值，一方面使得语义组合计算可以按照单词间修饰关系概率结合所有单词的语义；另一方面，基于图神经网络的语义组合计算可以实现并行处理，支持训练和预测阶段的批处理，提升计算效率。本文模型与已有方法对比示意图如图2所示。其中，图2(a)是基于序列化结构的组合方法，图2(b)是基于树结构的语义组合方法，图2(c)是本文提出的基于图结构的语义组合方法。我们以复述识别作为语义组合计算的目标任务，在公开汉语复述识别数据集LCQMC(Liu et al., 2018)上的实验结果表明，本文提出的模型优于已有序列结构的语义组合计算方法，可以有效改进复述识别性能。同时，本文提出的语义组合方式支持批处理操作，在预测阶段速度是Tree-LSTM的30倍，同时能保持较高的精度。

本文的主要贡献如下：(1)提出一种基于图的依存句法分析和语义组合计算的联合框架；(2)提出一种基于图神经网络的语义组合方法，可以实现并行处理，支持训练和预测阶段的批处理，提升计算效率。

2 相关工作

语义组合计算方法主要分为基于序列化的语义组合计算方法和基于句法结构的语义组合计算方法。在序列化的语义组合计算中，如图2(a)所示，模型从左到右依次读入单词，如LSTM,RNN等(Tang et al., 2016; Mueller and Thyagarajan, 2016)。这些方法的优点是可以表示任意长度句子的上下文信息，其语义信息从左向右积累，最终将最后时刻的隐状态向量视为整个句子的语义表示。Kim (2014)利用卷积神经网络获得句子的语义表示，具体做法是使用卷积核在输入句上从左到右滑动，每次滑动捕捉句子局部区域的特征，使得CNN更能捕获n-gram特征，最后通过最大池化获得句子的语义表示。基于序列化方法为了追求运算效率，直接对句子的文本序列进行语义组合计算，未对结构信息加以利用，难以对结构不同带来的语义差异加以区分。

近年来有许多工作(Socher et al., 2012; Li et al., 2015; Tai et al., 2015)试图引入句法结构进行语义组合计算，并在情感分类(Tai et al., 2015)，自然语言推理(Bowman et al., 2016; Mou et al., 2016)等任务中验证了比序列化模型更好的性能。Tai et al. (2015)使用了树结构进行语义组合计算，从树的叶子节点开始将语义信息从底向上传递，最后在树的根节点获得句子表示。Chen et al. (2017)设计了增强树结构表示，利用短语结构树进行语义组合计算。但是这些方法受自底向上的组合顺序的限制无法实现并行计算，难以支持训练和预测阶段的批处理，导致计算时间过长难以满足实际需求。Mou et al. (2016)提出了树结构的卷积操作，通过对每个节点的孩子节点进行卷积操作获得该节点的语义表示，最后对所有节点使用最大池化获得句子语义表示，并在自然语言推理任务上验证其有效性。该方法虽然可以实现并行计算，但只计算了直接孩子的语义信息，没有考虑子孙节点的语义。

本文提出的基于图的依存分析模型和图神经网络语义组合计算联合框架，使用带有概率的依存关系结构图进行语义组合计算，一方面可以实现并行处理，另一方面可以考虑所有节点的语义信息。

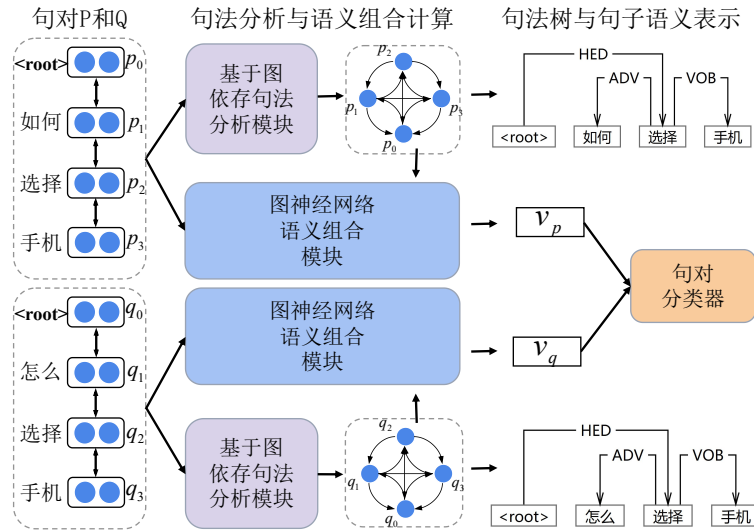


图 3: 本文提出的联合模型整体框架

3 基于图神经网络的依存分析和语义组合计算联合模型

我们采用依存句法树作为句法结构指导语义组合计算，提出了基于图的依存句法分析和语义组合计算的联合框架，模型架构如图3所示。模型接收句对 $P = \{p_1, \dots, p_N\}$ 和 $Q = \{q_1, \dots, q_M\}$ 。首先经依存句法分析分别得到带有概率的依存关系结构图，并从中得到依存树，然后经语义组合计算利用该图得到句子的语义表示，并送入句对分类器进行判断。在模型训练阶段，我们联合依存分析和复述识别任务两个目标共同学习模型参数。

3.1 依存句法分析

本文采用基于图的依存句法分析方法(Dozat and Manning, 2017)，该方法可以考虑全局信息进行依存分析决策，最近研究显示该方法在性能上超过了基于转移的依存分析方法(Ji et al., 2019)。下面，我们以句子 $P = \{p_0, p_1, \dots, p_N\}$ 为例，详细介绍依存句法分析模块。按照通常做法，我们在每个句子的开头加入根节点的标识“<root>”作为 p_0 。

首先将输入的单词序列转化为数值向量表示，我们采用预训练词向量、随机初始化词向量和词性标签向量三部分构成输入词向量。我们用 $e(p_i) \in \mathbb{R}^d$ 表示预训练词向量， $e'(p_i) \in \mathbb{R}^d$ 表示随机初始化词向量， $e(pos_i) \in \mathbb{R}^{d_{pos}}$ 表示词性标签向量， d_{pos} 为词性的嵌入维度，三部分的表示在训练中被更新。最终，每个单词的表示由公式1计算得出，其中 \oplus 为拼接操作。

$$x_i = (e(p_i) + e'(p_i)) \oplus e(pos_i) \quad (1)$$

为了捕捉句子长距离的上下文信息，我们采用深层双向LSTM(BiLSTM)学习句子中的词表示。其中，第 i 时刻(对应第 i 个单词)的隐藏状态表示如公式2所示。

$$h_i = BiLSTM(x_i, \overleftarrow{h}_{i+1}, \overrightarrow{h}_{i-1}, \theta) \quad (2)$$

其中， \overleftarrow{h}_i 和 \overrightarrow{h}_i 是在时刻 i 前向和逆向LSTM的隐藏表示； θ 为BiLSTM中的参数。

本文使用图 $G = (V, E)$ 表示句子 P 的依存关系图，其中 $V = \{p_0, p_1, \dots, p_N\}$ 是句子中单词节点集合， E 是依存关系边集合。序列 P 中每个词与图上的节点对应，使用 $p_j \rightarrow p_i$ 表示核心词(head) p_j 与依存词(dep) p_i 之间存在依存关系。由于句子中任意两个单词之间存在两种依存关系 $p_j \rightarrow p_i$ 和 $p_i \rightarrow p_j$ ，需要为每个单词计算其作为核心词或依存词的向量表示。为此，我们为每个单词设置两个向量表示，一个是单词作为依存词的表示，另一个是单词作为核心词的表示。对于这两种表示的计算，我们分别采用多层感知器对BiLSTM的输出 h_i 进行计算，如公式3和4所示(Dozat and Manning, 2017)。在此基础上，可以为所有单词对中的两种依存关系计算得分，具体的我们采用双仿射注意力机制进行计算，计算过程如公式5所示。其中， s_{ij} 表

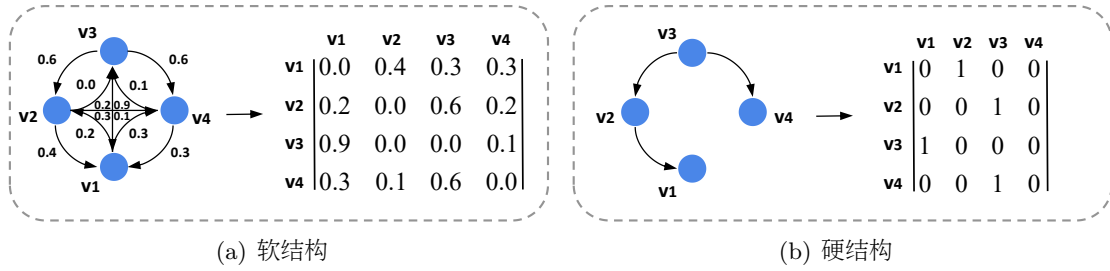


图 4: 基于图神经网络使用结构信息的两种方式

示 $p_j \rightarrow p_i$ 的得分，得分越大表示构成 $p_j \rightarrow p_i$ 的可能性越大。

$$r_i^{dep} = MLP^{(dep)}(h_i) \quad (3)$$

$$r_j^{head} = MLP^{(head)}(h_j) \quad (4)$$

$$s_{ij} = r_i^{dep} U r_j^{head} + r_j^{head} u \quad (5)$$

其中， U 表示权重矩阵， u 表示偏置项。

$s_i = [s_{i0}, \dots, s_{ij}, \dots, s_{iN}] (j \in \{0, 1, \dots, N\})$ ， s_{ij} 是 $p_j \rightarrow p_i$ 依存关系的得分，其中 s_{i0} 用于衡量第 i 个单词成为根 $ROOT$ 的可能性。随后采用公式6进行归一化操作得到概率分布 α_i ，由 $\alpha_i (i \in \{0, 1, \dots, N\})$ 构成依存关系概率矩阵 α 。最后采用最大生成树算法解码获得句子的依存结构。在训练阶段，我们使用交叉熵作为损失函数，如公式7所示。

$$\alpha_i = softmax(s_i) \quad (6)$$

$$\mathcal{L}_0 = - \sum_{k=1}^{N_s} \sum_{i=1}^{N_k} \beta_i^k \log(\alpha_i^k) \quad (7)$$

目标函数 \mathcal{L}_0 表示交叉熵损失， N_s 表示一个批次中句对个数， N_k 表示第 k 个 P 句的单词个数， β_i^k 是第 k 个 P 句中第 i 个单词真实核心词的独热码表示。参照(Dozat and Manning, 2017)，我们设计预测依存关系类型的预测模型，此部分构成的损失函数为 \mathcal{L}_1 。我们将预测结构的损失 \mathcal{L}_0 与预测依存关系类型的损失 \mathcal{L}_1 相加构成 \mathcal{L}_p 。同理，对于句子 Q 我们可以得到相应的目标函数 \mathcal{L}_q 。最后，我们将 \mathcal{L}_p 与 \mathcal{L}_q 相加作为依存分析模型的损失函数 \mathcal{L}_{dep} 。

3.2 语义组合计算

我们提出一种基于图神经网络的语义组合计算方法，通过利用上一节的依存分析提供的依存关系的概率矩阵 α 进行语义组合计算，以支持批处理大幅提升计算速度。根据句法分析， α_{ij} 表示单词 p_j 是 p_i 核心词的概率，我们将依存句法分析学习到的权重 α_{ij} 视为依存关系 $p_j \rightarrow p_i$ 的语义相关性权重，同时将 h_i 视为图上节点 p_i 的语义表示，然后在此图基础上进行语义组合计算。

本文采用图信息传递机制(Veličković et al., 2018; Huang et al., 2019)建模图中每个节点的语义信息，首先节点 p_i 从邻节点收集语义信息，我们设计了两种收集的方式。第一种收集方式利用依存关系概率矩阵 α 直接作为权重结合邻节点的语义表示，计算公式如8所示，我们称这种方式为软结构信息，示意图如图4(a)所示。第二种收集方式依据依存结构结合具有依存关系节点的语义信息(Huang et al., 2019; Yao et al., 2018)，首先修改依存关系概率矩阵 α ，对于单词 p_i 设置概率最大的核心词的概率为1，其他单词的概率设置为0，具体修改方式如公式9所示，然后再按公式8进行语义信息的收集，我们称这种方式为硬结构信息，示意图如图4(b)所示。得到邻节点语义信息 M_i 后，根据公式10更新当前节点的语义表示。

$$M_i = \sum_{j=1}^N \alpha_{ij} h_j \quad (8)$$

$$\alpha_{ik} = \begin{cases} 1 & ,k = \operatorname{argmax}\alpha_i \\ 0 & ,\text{else} \end{cases} \quad (9)$$

$$h'_i = (1 - \eta_{p_i})\operatorname{LeakyReLU}(M_i) + \eta_{p_i}h_i \quad (10)$$

其中, $M_i \in \mathbb{R}^d$ 是节点 p_i 从邻节点获得的语义信息, $h_i \in \mathbb{R}^d$ 表示节点 p_i 原始的语义表示, $\eta_{p_i} \in \mathbb{R}$ 是节点 p_i 的语义更新权重, 控制应保留 p_i 多少原来的语义信息, $1 - \eta_{p_i}$ 用于控制节点 p_i 接收到多少邻节点的语义信息。最后, 使用平均池化获得句子的语义表示。句子语义表示定义为:

$$v_p = \frac{1}{|N_p|} \sum_{i \in N_p} h'_i \quad (11)$$

其中, N_p 是句子 P 中单词节点下标的集合, $|N_p|$ 是句子 P 中单词的个数。 v_p 即为句子 P 的语义表示, 同理, 对于句子 Q , 我们可以获得其语义表示 v_q 。

为了检验本文基于图神经网络的语义组合计算方法能更好的学习句子的语义表示, 我们联合了复述识别任务。给定句对 P 和 Q , 预测两个句子是否具有相同的语义。首先基于语义组合计算模块, 为句对中的每个句子生成语义表示 v_p 和 v_q 。然后, 使用这两个句子的语义表示(v_p 和 v_q)构造特征向量 d (Mou et al., 2016), 如公式12所示。然后将此特征向量 d 送入句对分类器。

$$d = v_p \oplus v_q \oplus (v_p - v_q) \oplus (v_p \odot v_q) \quad (12)$$

其中, \odot 表示按元素乘积操作, \oplus 表示向量拼接操作, $d \in \mathbb{R}^{4d}$ 是构造的特征向量, 句对分类器我们采用多层感知机的方式, 如公式13所示。

$$\hat{y} = \operatorname{softmax}(MLP^{(clf)}(d)) \quad (13)$$

在训练阶段我们使用交叉熵作为损失函数, 定义为:

$$\mathcal{L}_{pair} = - \sum_{i=1}^{N_s} g_i \log(\hat{y}_i) \quad (14)$$

其中, N_s 为一个批次中句对的个数, g_i 表示第 i 个句对是否为复述, 如果为复述关系, $g_i = [1, 0]$, 如果为非复述关系, 则 $g_i = [0, 1]$, \hat{y}_i 是第 i 个句对各类别的估计概率, 如公式13所示。

3.3 联合学习

本文提出的联合模型涉及到两个任务, 依存句法分析和语义组合计算, 我们采用复述识别验证语义组合计算。由此, 模型需要同时学习和优化多个学习目标。在传统的联合学习中, 通常对各个任务的损失进行线性加权求和, 如公式15, 该方法权重较难设定。为了解决多目标联合学习问题, 我们采用Kendall et al. (2018) 设计的自学习多目标权重方法。该方法根据噪声方差作为模型收敛程度的评估, 进行比重调整。其目标函数设计如公式16。

$$\mathcal{L} = (1 - w)\mathcal{L}_{pair} + w\mathcal{L}_{dep} \quad (15)$$

$$\mathcal{L} = \frac{1}{2\sigma_1^2}\mathcal{L}_{pair} + \log\sigma_1^2 + \frac{1}{2\sigma_2^2}w\mathcal{L}_{dep} + \log\sigma_2^2 \quad (16)$$

其中 $\sigma_1, \sigma_2 \in \mathbb{R}$ 为学习的参数, 跟随训练过程被更新, \mathcal{L}_{dep} 为依存分析的损失函数, \mathcal{L}_{pair} 为复述识别的损失函数。

4 实验

4.1 数据集介绍和超参数设置

本文使用公开汉语复述识别数据集LCQMC (Liu et al., 2018)作为实验数据。我们采用高精度的哈工大语言技术平台ltp3.4.0¹(Che et al., 2010)获取分词、词性和依存句法标注,我们将依存句法标注视为ground truth。表1给出了LCQMC数据集的统计信息。

数据集	划分	句对数	正例数	负例数	词数
LCQMC	训练集	238,766	138,574	100,192	3279k
	开发集	8,802	4,402	4,400	138k
	测试集	12,500	6,250	6,250	152k

表 1: 实验数据集

实验中采用预训练的Word2Vec 词向量(Mikolov et al., 2013), 预训练词向量为200维。词性标签向量设置100维, 设置所有LSTM结构的隐藏层为400维, 层数为3。对与 $MLP^{(dep)}$ 和 $MLP^{(head)}$ 设置层数都为1层隐藏层维度分别为100和500, 采用leakyrelu激活函数, α 设置为0.1。对于 $MLP^{(cf)}$ 设置层数为2, 隐藏层维度为800和400, 采用相同的激活函数。我们采用Adam (Kingma and Ba,)优化算法, 设置初始学习率大小为 $2e-3$, 为 β_1 为0.9, β_2 为0.9。在每一轮迭代中, 学习率以0.95的频率衰减。训练batch的大小为128。为了防止过拟合, 我们使用了dropout。设置词向量输入层的dropout率为0.33, leakyrelu层输出层的dropout率为0.33。与已有工作一致, 我们采用无标记依存正确率UAS和带标记依存正确率LAS作为依存分析评价指标, 采用Accuracy和融合Precision和Recall的综合指标F1 值作为复述识别的评价指标。

4.2 基于自学习多目标权重的实验结果

如果按照公式15计算损失函数, 为了找到合理的 w 需要多次实验, 实验结果如表2所示。表2显示了不同权重 w 对依存分析和复述识别任务性能的影响结果。当 w 较小时, 复述识别性能较好, 但是依存分析精度较低; 当 w 较大时, 依存分析精度较好但是复述识别性能较低。当 w 设置为0.9时, 依存分析的结果达到最好, 带标记正确率达到94.37%, 但复述识别的Accuracy只有73.37%。当 w 设置为0.5时, 能共同得到较好的性能, 复述识别Accuracy为76.31%, 依存分析LAS为93.99%。

如果按照公式16, 采用Srivastava et al. (2014)设计的多目标损失函数, 复述识别Accuracy达到76.77%, 依存分析LAS 92.70 %, 与公式15中 $w = 0.5$ 时的最好结果相比, 其复述识别的准确率提高0.46个点, 显示该方法优于线性加权的损失函数。随后实验中我们采用Kendall et al. (2018) 设计的多目标函数方法。

4.3 语义组合次数实验结果

3.2节介绍了每个节点结合邻近节点语义信息更新自身语义表示的组合计算方法, 使得每个节点包含了直接核心词的语义信息。如果在此基础上再进行一次语义组合计算, 将使每个节点获得间接核心词的语义信息。为了分析语义组合次数的影响, 我们分别进行了基于0次、1次、2次和3次语义组合计算的评测, 实验结果如表3所示。 $n=0$ 表示没有利用结构信息, $n=1,2,3$ 表示以不同语义组合计算次数利用结构信息。与 $n=0$ 相比, $n=1$ 的模型在测试集上, 复述识别在 F_1 和Accuracy分别提高了1.96和1.97个点, 说明句法结构指导语义组合计算上的有效性。

与 $n=2,3$ 相比, $n=1$ 的模型在复述识别任务上均优于 $n=2,3$ 的模型。实验结果表明继续增加组合次数并没有提升效果, 同时, 随着组合次数的增加, 模型的复杂度也会增加, 随后实验中我们选择一次语义组合计算。另外我们注意到联合模型并未给依存分析带来性能上的提升, 一方面由于本文的重点放在语义组合计算上, 还没有找到同时提升依存分析精度的有效联合方法; 另一方面本文使用的依存结构标注并非人工标注, 我们分析存在一定错误难以给出依存分析模型的正确评测结果。

¹<http://ltp.ai/download.html>

目标函数权重 w		复述识别	依存句法
复述识别	依存句法	Acc[%]	LAS[%]
0.975	0.025	74.97	86.37
0.95	0.05	75.68	89.09
0.9	0.1	75.61	90.75
0.85	0.15	76.68	91.61
0.8	0.2	76.47	92.33
0.7	0.3	76.75	93.31
0.5	0.5	76.31	93.99
0.1	0.9	73.37	94.37
Kendall et al. (2018)		76.77	92.70

表 2: 不同 w 下, 联合模型在开发集中两个任务上的性能

组合次数	开发集				测试集			
	依存分析		复述识别		依存分析		复述识别	
	UAS	LAS	F_1	Acc	UAS	LAS	F_1	Acc
n=0	93.77	92.67	73.93	74.07	95.36	94.25	79.88	77.57
n=1	93.92	92.70	77.23	76.77	95.32	94.19	81.84	79.54
n=2	93.82	92.60	76.94	76.02	95.27	94.16	80.76	78.24
n=3	93.77	92.57	76.74	75.47	95.23	94.08	79.92	77.00

表 3: 语义组合次数在不同任务上的性能, n=0表示没有利用句法结构信息

4.4 模型对比实验

我们与基于序列化和树结构的5种语义组合计算方法进行比较, 对比模型分为以下几类:

Baseline: 上一节中n=0的模型, 即包含序列信息无结构信息。

MeanVector: 将词表示的平均池化作为句子的语义表示(Blacoe and Lapata, 2012), 其中词表示的计算方法如公式1, 该方式无序列信息也无句法结构信息。

CNN: 基于卷积神经网络的语义组合计算方法Kim (2014)和Liu et al. (2018), 该方式包含序列信息无结构信息。

BiLSTM: 使用前向LSTM和后向LSTM最后时刻的隐状态向量拼接作为句子表示(Mueller and Thyagarajan, 2016; Tomar et al., 2017; Liu et al., 2018), 该方式包含序列信息无结构信息。

TreeLSTM: 使用Tai et al. (2015)提出的Child-Sum Tree-LSTM, 利用依存结构树进行语义组合计算, 将根节点获得的隐状态向量视为句子的表示, 该方式包含结构信息。

在复述识别任务上, 我们的模型与5种模型在测试集上的评测结果如表4所示。

从表4的结果可以看出, 在无结构信息的4种方法中, 我们设计的Baseline取得了最好的结

是否利用结构信息	方法	F_1	Acc
否	Baseline	79.88	77.57
否	Mean vectors	78.68	75.08
否	CNN	75.70	72.80
否	BiLSTM	78.92	76.10
是	Tree-LSTM	82.02	80.22
是	Our	81.84	79.54

表 4: 在复述识别上和已有序列化和树结构语义组合方式的比较结果

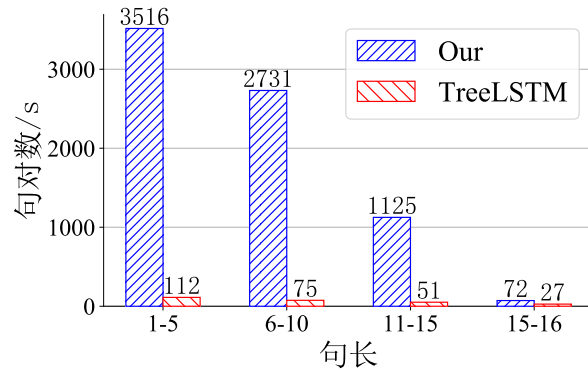


图 5: 句长对模型预测速度的影响

系统	Dev		Test	
	F_1	Acc	F_1	Acc
(1)无结构信息	73.93	74.07	79.88	77.57
(2)硬结构信息	77.01	76.51	81.52	79.15
(3)软结构信息	77.23	76.78	81.84	79.54

表 5: 模型融入依存结构信息有效性分析结果

果, 在 F_1 和Accuracy达到79.88%和77.57%。我们分析原因是在我们的模型中使用了图神经网络, 当处理较长句子时, 每个节点能从较远节点收集语义信息更新自身表示, 能捕捉较长句子的上下文信息。与未利用结构信息的模型中最好精度的Baseline相比, 我们的模型在Baseline的基础上加入依存结构优化目标, 在 F_1 和Accuracy上分别提高了1.96和1.97个点。实验结果表明利用句法结构信息进行语义组合计算的有效性。与利用结构信息的Tree-LSTM相比, 我们模型在 F_1 和Accuracy略低于Tree-LSTM 0.18和0.68个点。我们分析原因是Tree-LSTM直接使用了我们视为ground truth的依存标签, 而我们的模型使用的是依存标签训练之后依存句法分析模块产生的依存结构, 其依存分析精度没有ground truth高。

已有的基于结构的Tree-LSTM每次只能处理一个句对, 本文采用基于图的依存分析和图网络语义组合方法, 可以实现对多个句对的批处理, 从而解决已有模型预测速度慢的问题。我们在不同长度的句子上对模型的预测速度进行评测, 并与Tree-LSTM进行对比, 评测结果如图5所示。句长在1-5和6-10中我们模型预测速度是Tree-LSTM的30倍; 在句长为11-15中, 速度是Tree-LSTM的20倍。这些结果显示本文提出的模型在预测速度上较Tree-LSTM有显著优势。

以上分析结果显示, 本文提出的基于依存句法分析和复述识别的联合模型, 采用基于图神经网络的语义组合方法, 可以有效利用句法结构信息改进语义组合计算, 提高复述识别系统的精度和计算速度。

4.5 结构信息有效性分析

我们分析了模型中结构信息对最终复述识别精度的影响, 实验结果展示在表5中。从表5的实验结果来看, 基于图神经网络引入依存结构信息, 有效改进了复述识别的性能。模型(1)没有使用结构信息, 仅使用了复述识别的目标函数进行优化, 未考虑句子的句法结构, 复述识别的Accuracy达到77.57%。模型(2)引入了句法目标训练模型参数, 采用了本文提出的硬结构信息, 复述识别Accuracy达到79.15%, 对比没有结构信息提高了1.58个百分点, 这表明引入句法结构对语义组合的有效性。模型(3)采用了软结构信息, Accuracy达到79.54%, 进一步改进了复述识别的性能, 同时, 实验表明本文提出的软结构依存信息在性能上优于硬结构的方法。最终, 实验结果表明, 本文提出的基于句法结构进行语义组合计算, 可以有效学习句子的语义表示, 提高了复述识别系统的精度。

ID	句子1(P)	句子2(Q)	Bleu	真实标签	Our	Baseline
A	网站排名推广, 主要有哪些推广方式, 效果好点的.	目前网上有哪些推广方式	0.19	T	T	F
B	我想知道女生各种发型的名字, 加上配图	女生各种发型名称图片	0.24	T	T	F
C	亲爱的韩语怎么说	亲爱的韩语怎么写?	0.71	F	F	T
D	小薏米和大薏米有什么区别	薏仁粉和薏米粉有什么区别	0.61	F	F	T
E	如何编织小狗狗的衣服要方法及图解	如何给小狗做衣服图片	0.24	T	F	F
F	小学二年级语文	小学二年级语文题	0.80	F	T	T

表 6: 一些复杂的例子在本文模型和Baseline上的表现, *T*表示是复述关系, *F*表示非复述关系。

4.6 实例分析

我们在LCQMC的测试集中挑选了一些句对进行进一步分析。使用1-gram计算句子*P*与句子*Q*的Bleu值, 对于复述识别来说, Bleu很高的非复述句对和Bleu很低的复述句对, 都是很难的任务, 基于浅层信息的方法很难正确识别, 需要深层语义理解才可解决。我们特地选择这样的句对评测我们模型的效果, 分析结果如表6所示。

示例A-B为Bleu较低的复述句对, 因此, 容易识别为非复述关系。但是本文模型能够正确识别为复述关系, 而Baseline错误的识别为非复述关系。这一对比结果表明本文利用句法结构进行语义组合计算的方法可以捕捉句对之间深层的语义相关性, 实现正确判断。

示例C-D为Bleu较高的非复述句对, 因此, 容易识别为复述关系。但是本文模型能够正确识别为非复述关系, 而Baseline错误的识别为复述关系。这一对比结果表明句法结构更易于解决涉及结构复杂表达的语义理解。

示例E-F是Baseline和本文模型都产生错误的情况。E为Bleu较低的复述句对。我们分析预测错误的原因是句子的表达口语化, 句法分析很难进行正确分析。F为Bleu较高的非复述句对, 其中含有相似的词语“语文”和“语文题”, 我们分析预测错误的原因是词的语义表示不能有效的区分二者, 这使模型错误的认为它们是复述的关系。对于更复杂的情况, 句子的语义表示仍然面临很多的问题, 例如歧义性以及口语表达。模型可能需要更多的推理信息来区分这些关系并做出正确的决定, 例如结合外部知识用于帮助模型更好地理解词汇和短语语义。

5 总结与展望

本文提出一种依存句法分析和语义组合计算的联合框架, 设计了基于图的依存句法分析模型和基于图神经网络语义组合计算模型, 利用依存分析给出的带有概率的依存关系结构图, 实现软结构的语义组合计算方法。一方面图模型中的并行计算能够支持训练和预测阶段的批处理, 极大提高计算速度; 另一方面两个任务的联合学习可使语义表示同时学习句法结构和语义的上下文信息, 提高复述识别精度。

今后, 我们考虑结合预训练模型, 如ELMO, BERT, 以改进模型性能。同时, 探索联合模型中提升依存分析精度的方法, 从而进一步提升语义组合计算的精度。

参考文献

- William Blacoe and Mirella Lapata. 2012. A comparison of vector-based representations for semantic composition. In Jun'ichi Tsujii, James Henderson, and Marius Pasca, editors, *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 546–556. ACL.
- Samuel R. Bowman, Jon Gauthier, Abhinav Rastogi, Raghav Gupta, Christopher D. Manning, and Christopher Potts. 2016. A fast unified model for parsing and sentence understanding. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1466–1477, Berlin, Germany, August. Association for Computational Linguistics.
- Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved statistical machine translation using paraphrases. In Robert C. Moore, Jeff A. Bilmes, Jennifer Chu-Carroll, and Mark Sanderson, editors, *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 4-9, 2006, New York, New York, USA*. The Association for Computational Linguistics.

- Wanxiang Che, Zhenghua Li, and Ting Liu. 2010. LTP: A Chinese language technology platform. In *Coling 2010: Demonstrations*, pages 13–16, Beijing, China, August. Coling 2010 Organizing Committee.
- Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668, Vancouver, Canada, July. Association for Computational Linguistics.
- Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.
- Miao Fan, Wutao Lin, Yue Feng, Mingming Sun, and Ping Li. 2018. A globalization-semantic matching neural network for paraphrase identification. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM '18*, page 2067–2075, New York, NY, USA. Association for Computing Machinery.
- Lianzhe Huang, Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2019. Text level graph neural network for text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3444–3450, Hong Kong, China, November. Association for Computational Linguistics.
- Tao Ji, Yuanbin Wu, and Man Lan. 2019. Graph-based dependency parsing with graph neural networks. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Papers*, pages 2475–2485. Association for Computational Linguistics.
- Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 7482–7491. IEEE Computer Society.
- Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October. Association for Computational Linguistics.
- Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization.
- Jiwei Li, Thang Luong, Dan Jurafsky, and Eduard Hovy. 2015. When are tree structures necessary for deep learning of representations? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2304–2314, Lisbon, Portugal, September. Association for Computational Linguistics.
- Xin Liu, Qingcai Chen, Chong Deng, Huajun Zeng, Jing Chen, Dongfang Li, and Buzhou Tang. 2018. LCQMC: a large-scale Chinese question matching corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1952–1962, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.
- Tomas Mikolov, Greg Corrado, Chen Kai, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations (ICLR 2013)*.
- Lili Mou, Rui Men, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin. 2016. Natural language inference by tree-based convolution and heuristic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 130–136, Berlin, Germany, August. Association for Computational Linguistics.
- Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In Dale Schuurmans and Michael P. Wellman, editors, *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 2786–2792. AAAI Press.
- Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*.

- Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958.
- Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566, Beijing, China, July. Association for Computational Linguistics.
- Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective LSTMs for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3298–3307, Osaka, Japan, December. The COLING 2016 Organizing Committee.
- Gaurav Singh Tomar, Thyago Duque, Oscar Täckström, Jakob Uszkoreit, and Dipanjan Das. 2017. Neural paraphrase identification of questions with noisy pretraining. In Manaal Faruqui, Hinrich Schütze, Isabel Trancoso, and Yadollah Yaghoobzadeh, editors, *Proceedings of the First Workshop on Subword and Character Level Models in NLP, Copenhagen, Denmark, September 7, 2017*, pages 142–147. Association for Computational Linguistics.
- Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations*. accepted as poster.
- Liang Yao, Chengsheng Mao, and Yuan Luo. 2018. Graph convolutional networks for text classification. *CoRR*, abs/1809.05679.