联合依存分析的汉语语义组合模型

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

在语义组合方法中，结构化方法强调以结构信息指导词义表示的组合方式。现有结构化语义组合方法使用外部分析器获取句法结构信息，导致句法分析与语义组合相互割裂，句法分析的精度严重制约语义组合模型的性能，且训练数据领域不一致等问题会进一步加剧性能的下降。对此，本文提出联合依存分析的语义组合模型，将依存分析与语义组合进行联合，一方面在训练语义组合模型时对依存分析模型进行微调，使其能够更适应语义组合模型使用的训练数据的领域特点；另一方面，在语义组合部分加入依存分析的中间信息表示，获取更丰富的结构信息和语义信息，以此来降低语义组合模型对依存分析错误结果的敏感度，提升模型的鲁棒性。我们以汉语为具体研究对象，将语义组合模型用于复述识别任务，并在CTB5汉语依存分析数据和LCQMC汉语复述识别数据上验证本文提出的模型。实验结果显示，本文所提方法在复述识别任务上的预测正确率和F1值上分别达到76.81%和78.03%；我们进一步设计实验对联合学习和中间信息利用的有效性进行验证，并与相关代表性工作进行了对比分析。

关键词： 句法分析；语义组合；联合学习；图注意力网络

Chinese Semantic Composition Model with Dependency Parsing

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

In the semantic composition methods, the structural methods emphasize the combination mode of words' meaning representation guided by structural information. Existing structural semantic composition methods use external parser to obtain syntactic structure information, resulting in the separation of syntactic parsing and semantic composition. The accuracy of syntactic analysis will severely restrict the performance of semantic composition models, and the inconsistent training data fields will further aggravate the performance degradation. To solve this problem, this paper proposes a semantic composition model combined with dependency parsing. On the one hand, the dependency model is fine-tuned when training the semantic composition model, so that it can be more suitable for the domain characteristics of the training data used

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

by the semantic composition model. On the other hand, we add the intermediate information representation of dependency to the semantic composition part to obtain more abundant structural information and semantic information, so as to reduce the sensitivity of semantic composition model to erroneous results of dependency parsing and improve the robustness of the model. We take Chinese as the specific research object, apply semantic combination model to retelling recognition task, and verify the model proposed in this paper on CTB5 Chinese dependency parsing data and LCQMC Chinese retelling recognition data. The experimental results show that the prediction accuracy and F1 value of the method proposed in this paper reach 76.81% and 78.03% respectively in retelling recognition tasks. We further designed experiments to verify the effectiveness of joint learning and intermediate information utilization, and made comparative analysis with relevant representative work.

Keywords: Syntactic analysis , Semantic combination , Joint learning , Graphical attention network

1 引言

语义组合以一定的方式将句子中的词义表示进行计算合并，从而得到句子的语义表示。作为语义组合的重要组成部分，组合方式的选择对最终得到的语义表示的性能有着重要影响。目前主流的组合方式是序列化的语义组合方法，仅对句子进行序列化处理，忽视了句子的语法结构，导致获取的句子表示难以准确地反应句子的语义。汉语由于词序更加灵活，且缺乏表层变化信息等特点，因此在计算句子语义时需要句法结构信息的指导。依存句法信息由于与词义和语义关联更为密切，因此在汉语的语义表示研究领域，依存句法分析和语义组合的结合是未来主要的研究方向之一。

目前一些研究者尝试利用依存句法信息作为指导，构建树结构的语义组合方法，通过依存分析器预测句子的依存结构，然后根据依存树结构进行语义组合，在句子匹配等任务上取得了一定的成就(Mou et al., 2016)。但这类方法仍存在如下问题：（1）依存分析和语义组合相互割裂。现有方法直接使用外部依存分析器获得的依存句法信息，没有针对语义组合任务进一步优化依存分析模型，从而限制了最终获取的语义表示的精度。（2）数据领域不一致。依存分析与语义组合的训练数据可能来自不同领域，将会导致依存分析模型在应用于语义组合数据时精度降低，进而影响语义组合模型的性能。（3）信息利用不充分。使用外部依存分析器获取依存句法信息，仅能利用预测得到的依存句法树，而在依存分析过程中产生的结构信息和语义信息则未加利用，浪费了大量的中间信息。

针对上述问题，本文提出联合依存分析的语义组合模型。以依存句法树作为图注意力计算中的图，对每个节点的语义根据其孩子节点进行组合计算；然后提出依存分析中间信息的利用方法，将依存关系中作为头节点的语义信息引入语义组合模型，以降低依存分析的预测错误对语义组合模型带来的影响，提升语义组合模型的鲁棒性；最后通过依存分析与语义组合的联合学习，对依存分析模型进行领域自适应，提升依存分析模型的鲁棒性。我们将语义组合模型用于复述识别任务，在汉语复述识别数据集LCQMC上的预测正确率达到76.81%，F1值达到78.83%。

2 相关工作

目前语义组合方法主要可以分为两类：一种是将句子视为序列结构进行组合，将句子中各个词的信息进行加权整合，从而得到能够有效表达句子语义的表示；另一种则是利用句法结构作为语义组合的指导，根据句子中的结构关系对词义表示的组合顺序和方式加以限制，得到能更准确表达句子语义的表示。

对于如何通过组合词汇语义得到句子语义，一种朴素的思想是将词义表示相加得到句子语义表示，一般这种方式被称为加法组合（Additive Compositionality）(Mikolov et

al., 2013)。Hu et al. (2015)借鉴图像处理的技术，提出基于多层卷积操作的语义组合方法。Sutskever et al. (2014)和Cho et al. (2014)在他们提出的seq2seq（sequence to sequence）模型中，将RNN模型在最后一步的输出作为整个句子的语义表示,利用RNN类模型能够充分利用长距离信息的特点（以LSTM和GRU等变种为主），将句子信息进行有效地融合。该方法一度随seq2seq模型一道成为机器翻译、语音识别等生成任务中常用的语义组合方法。Chen et al. (2017)考虑到加法组合的语义组合方法会受到句子长度的影响，因此提出通过平均池化和最大池化相结合的方法，将句子长度的影响消去。该方法以其简单高效和对句子长度不敏感的特性，成为目前句子匹配任务中主流的语义组合方法。

鉴于目前句子结构主要被定义为树结构，因此结构化语义组合也以递归神经网络为基本模型。Zhu et al. (2015)对LSTM单元进行修改，提出针对二叉树句法结构的S-LSTM，利用LSTM能够进行长距离信息传递的特性，将各个词的信息经转化为二叉结构的短语句法树逐层传递至根节点，从而得到句子的语义表示。Mou et al. (2016)利用CNN能够轻松处理递归结构的特性，提出TBCNN（tree-based convolutional neural network），将依存句法树中每两层的语义信息加以融合，然后通过池化合并得到最终的句子语义表示。

序列化语义组合方法的优点是模型结构简单，能够快速得到句子表示，但由于忽视了句子内在的结构，对于序列差异较小的句子则难以区分。结构化语义组合方法能够更精确地表达句子的语义，但对依存分析的精度有较高的要求，同时模型更为复杂，时间消耗也更长。由于获取高质量句法分析标注难度较大，而使用外部句法分析器获取句法结构，可能会由于句法分析与语义组合数据领域不一致的问题，导致句法分析精度降低。本文针对现有结构化语义组合方法存在的问题，提出联合依存分析的汉语语义组合模型，将依存分析与语义组合计算进行联合，并提出依存分析中间信息的利用方法，从而得到更好的句子语义表示。

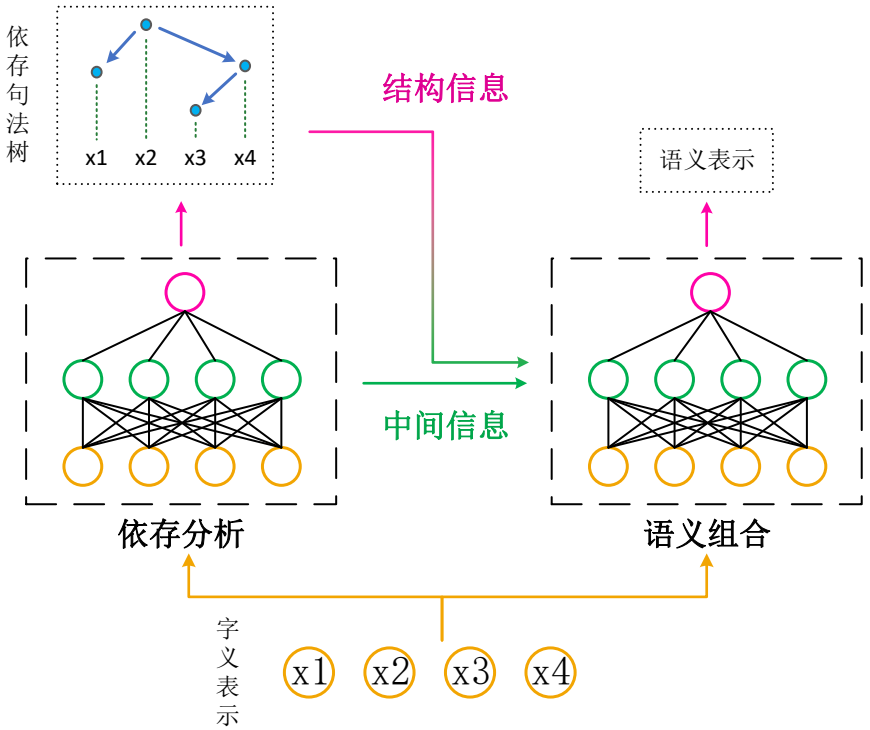

图 1: 联合依存分析的汉语语义组合计算模型

3 联合依存分析的汉语语义组合模型

针对现有结构化语义组合方法存在的问题，本文在Ma et al. (2018)的基础上，联合基于注意力的语义组合模型，提出联合依存分析的汉语语义组合模型。如图1所示，我们的模型主要包含依存分析和语义组合两大部分。依存分析部分对输入句子进行依存句法分析，并将分析过程中产生的中间信息和最终得到的依存句法树传递给语义组合部分；语义组合部分根据句子中每

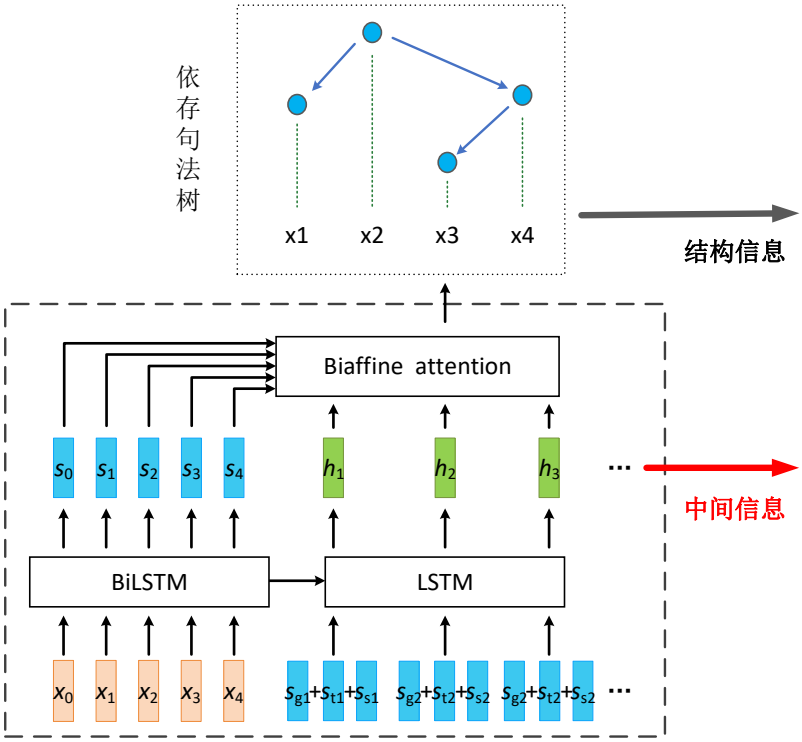

图 2: 联合模型中的依存分析部分

个字的字义表示，加入依存分析部分产生的中间信息，以依存句法树作为指导进行语义组合，从而得到句子的语义表示。

3.1 基于Stack-Pointer Networks的依存分析模型

Stack-Pointer Networks(StackPTR)是一个基于转移的依存分析模型。我们在StackPTR的基础上，针对汉语没有明显分词标记的特点，以每个词的最后一个字为根节点构建两层词内依存结构，以此进行字符级汉语依存分析。StackPTR模型大致框架如图2所示，其中每一步计算中得到的头节点表示h_i包含较为丰富的结构信息，因此我们将其作为依存分析的中间信息表示传递给语义组合部分。StackPTR中的具体细节请见Ma et al. (2018)。

3.2 基于注意力的语义组合模型

虑到每个字对句子语义的贡献程度不同，我们在语义计算部分提出基于注意力的语义组合模型，利用依存分析部分得到的结构信息作为指导，用注意力得分作为信息的权重，进行字义表示的组合。

如图3所示，模型主要分为字义编码层、字义组合层和句义输出层三个部分。字义编码层对每个字的语义表示进行编码；字义组合层以依存句法树作为语义计算的结构，将每个依存节点的信息传递给头节点；句义输出层将语义组合计算得到的每个字的结构化语义信息进行池化合并，得到表示句子语义的向量表示。

3.2.1 字义编码层

参考Hochreiter and Schmidhuber (1997)，我们使用双向LSTM进行字义表示的编码。依存分析模型的中间信息以字向量的形式，传递了丰富的结构信息和语义信息。我们将其作为额外的字义信息，对预训练字向量进行扩充。对于给定的句子$x = \{x_1, x_2, \ldots, x_n\}$，编码层首先将每个字的原始向量表示$x_i$与依存分析中间信息表示$h_i$进行拼接，得到字的向量表示$x_i'$。然后我们将输入双向LSTM，编码句子信息得到每个字在句子中的语义表示m_i。

此外，为了提升句子全局信息的利用，我们将双向LSTM两个方向的最后一步输出进行拼接，得到句子表示m_x，作为字义组合层中的额外信息输入。

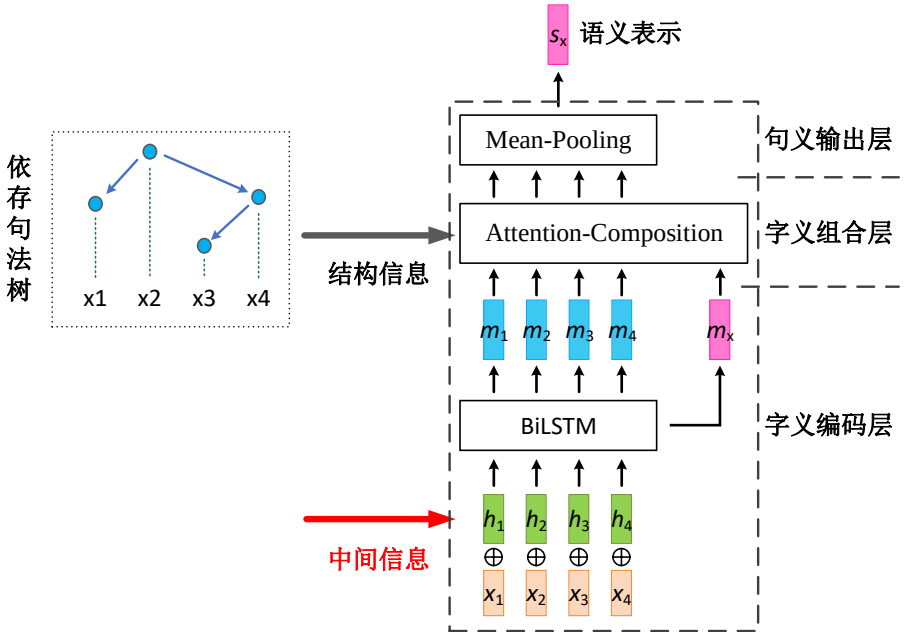

图 3: 基于注意力的语义组合模型

3.2.2 字义组合层

我们在语义组合计算层使用图注意力网络(Hochreiter and Schmidhuber, 1997)进行字义表示的组合计算，其中依存分析部分预测出的依存句法树作为指示节点相关性的有向图，对字义编码层输出的字义表示进行语义组合，其中每个节点在计算时仅考虑其依存节点，如图4所示。

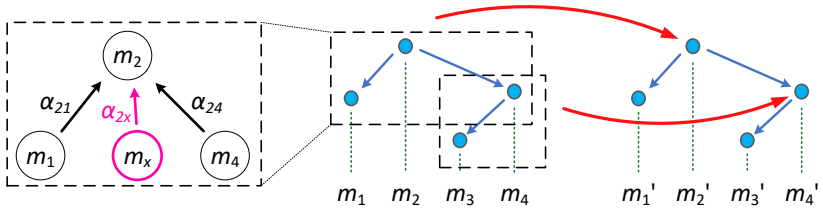

图 4: 语义组合计算层中的信息计算方式

我们将字义编码层中得到的句子表示m_x作为一个额外的节点，作为所有节点的依存节点参与图注意力网络的计算，使组合计算后每个字的表示都包含不同程度的句子全局信息。我们选择使用双线性变换作为注意力得分的计算机制，因此字义组合层的计算公式即为：

$$m_i^{'} = m_i + \sum_{j \in V(i)} m_i W m_j + m_i W m_x \times m_x \tag{1}$$

其中m_i和$m_i^{'}$分别表示第i个字在进行字义组合计算前后的语义表示；$j \in V(i)$表示节点i的所有依存节点，W为双线性变换的参数矩阵。

3.2.3 句义输出层

借鉴Mou et al. (2016)的工作，我们使用池化操作对字义组合计算后的字向量进行池化操作，获得最终的句子语义表示。为了使句子的语义表示不受句子长度的影响，同时尽可能保存更多的语义信息，我们选择使用平均池化对语义组合层输出的字向量表示进行合并。最终句子语义表示s_x的计算公式如下：

$$s_x = \frac{1}{n} \sum_{i=1}^{n} m_i^{'} \tag{2}$$

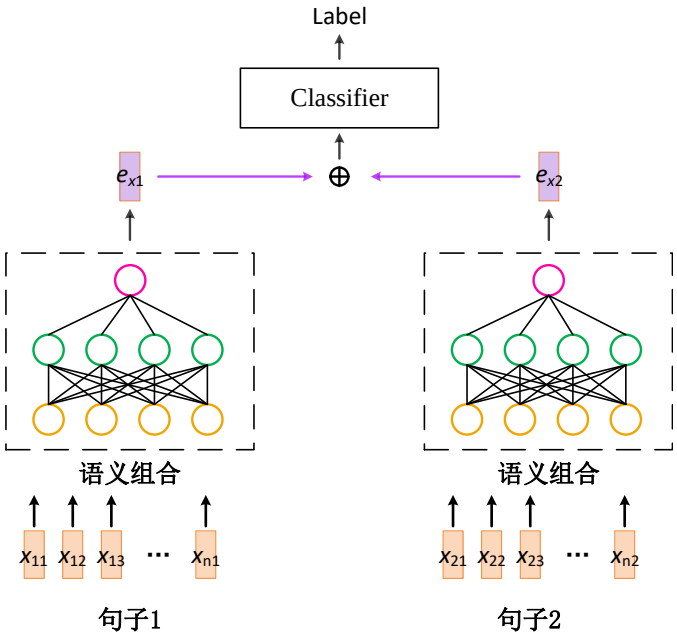

图 5: 复述识别模型

3.3 模型训练

3.3.1 复述识别任务

复述识别的主要目标是判断给定的两个句子是否表达相同（或相近）的语义。本文认为在所有相关任务中，复述识别任务与句子语义关联最为紧密，因此提出基于复述识别任务的语义组合计算的训练和评测方法。为了能够更直观地反应语义组合模型对复述识别性能的影响，我们未使用目前复述识别任务中最流行的交叉注意力机制(Veličković et al., 2017)和预训练语言模型(Wang et al., 2017)，而是借鉴早期常用的获得句子语义表示后进行关系预测的方式(Hu et al., 2015)，对每个句子独立进行语义组合计算，然后使用分类器进行复述的判别。

我们的复述识别模型如图5所示，使用本章提出的语义组合模型（依存分析部分未在图中显示）对两个输入句分别进行语义组合计算，得到它们的语义表示。然后将两个句子的语义表示拼接后输入一个分类器，判别它们是否互为复述。

3.3.2 联合模型训练方式

在进行联合模型的训练时，我们考虑两种训练方式：一种是将依存分析模型进行预训练，然后在训练语义组合模型时对依存分析模型的参数进行微调；另一种是直接将两个任务进行迭代训练。

预训练方式： 我们首先对联合模型中的依存分析部分的模型参数进行预训练，然后进行复述识别任务的训练，并在训练过程中对依存分析部分的模型参数进行微调。模型的目标函数通过最小化交叉熵损失定义：

$$\mathcal{L}(\theta) = \mathcal{L}_{par}(\theta_{com}, \theta_{dep}) = -logP_{par}(y|x, \theta_{dep}, \theta_{com}) \tag{3}$$

其中θ_{com}和θ_{dep}分别表示依存分析和语义组合模型的参数，表示复述关系标签。

迭代训练方式：在模型训练时，我们每次随机从两个任务中选择一个任务，并从对应的训练集中随机选取一批数据进行模型的训练。其中，在进行依存分析任务的训练时，仅对依存分析部分的模型参数进行学习；在进行复述识别任务的训练时，同时对联合模型中两个部分的参数进行学习。模型的目标函数通过最小化交叉熵损失定义：

$$\mathcal{L}(\theta) = \mathcal{L}_{dep}(\theta_{dep}) + \mathcal{L}_{par}(\theta_{com}, \theta_{dep}) = -logP_{dep}(A|x, \theta_{dep}) - logP_{par}(y|x, \theta_{dep}, \theta_{com}) \tag{4}$$

其中θ_{com}和θ_{dep}分别表示依存分析和语义组合模型的参数，A表示字节别依存句法树的边集，y表示复述关系标签。为了使模型训练过程更为稳定，我们首先对依存分析模型进行了100次训练。

4 实验

4.1 实验设置

本文使用的依存分析实验数据为宾州汉语树库CTB5，复述识别实验数据为语义相似度数据集LCQMC(Liu et al., 2018)。两个数据集的划分及统计数据如表1和2所示。

数据集	句子数	平均句长	词数	字数
训练集	18104	44.4	494k	805k
开发集	352	32.8	7k	12k
测试集	348	39.5	8k	14k

表 1: CTB5数据划分

数据集	句对数	平均句长	字数	正例占比
训练集	239k	10.9	5.2m	0.58
开发集	9k	12.5	0.2m	0.50
测试集	13k	9.7	0.2m	0.50

表 2: LCQMC数据详情

4.2 参数设置

我们使用word2vec工具在gigaword生语料上预训练字向量，字向量维度为100维；LSTM隐藏层维度为400，Dropout率为0.33。模型训练使用Adam（Adaptive Moment Estimation）优化算法，依存分析模型初始学习率设置为0.002，复述识别模型初始学习率设置为0.0001。对于预训练的模型训练方法，为了对依存分析部分的模型参数进行微调，我们对其设置了一个较小的学习率0.00001。

4.3 实验结果与分析

4.3.1 联合模型训练方式对比

我们分别对管道模型（pipeline）、预训练方法（pre-train）和迭代训练方法（alternate）进行实验，并在复述识别和一体化依存分析任务上进行了比较。其中，我们将管道模型中依存分析部分的参数学习率设置为0，用以模拟使用外部依存分析器提供结构信息的传统结构化方法。

复述识别任务：三个模型在复述识别任务上的对比结果如表3所示，其中预训练和迭代训练两种方式相较于管道模型均有明显的提升，且使用迭代训练方式的模型取得了最好的结果。

模型	类型	Acc(%)	F1(%)
Ours (pipeline)	结构化（管道）	72.64	75.74
Ours (pre-train)	结构化（联合）	74.01	76.86
Ours (alternate)		76.37	78.03

表 3: 联合模型训练方式在复述识别任务上的对比结果

一体化依存分析：三个模型在一体化依存分析任务上的对比结果如表4所示，其中两种训练方式得到的模型都较参数调整前的依存分析模型性能更低，且使用迭代训练方式的模型降低更为明显。

模型	分词(%)	词性标注(%)	依存分析(%)
Ours (pipeline)	98.25	95.13	85.44
Ours (pre-train)	97.85	94.35	83.04
Ours (alternate)	97.93	94.22	82.13

表 4: 联合模型训练方式在依存分析上的对比结果

总结分析：总的来说，虽然我们的联合模型在一体化依存分析任务上的精度有所降低，但在复述识别任务上的精度有所提升。我们根据表1和表2中两种数据平均句长的对比，以及进行数据分析后发现：我们所使用的依存分析数据（CTB5）为新闻领域的文本，句子较长且表达形式较为书面化；复述识别数据（LCQMC）为搜索引擎上收集的问句，句子较短且表达形式较为口语化。两种数据在领域和语言现象上存在较大的差异，因此我们做出如下推断：（1）使用CTB5上训练的依存分析模型，在对LCQMC中的句子进行的依存分析精度会有明显的降低，并因此导致语义组合模型在复述识别任务的精度较低；（2）我们的联合模型能够针对LCQMC的数据特点，对依存分析部分的参数进行适当地调整，虽然使其在CTB5上的一体化依存分析精度有所降低，但能够隐式地提升其在LCQMC数据上的依存分析精度，进而提升在复述识别任务上的精度。

4.3.2 依存信息利用的对比

我们对联合模型中的语义组合部分使用到的依存信息进行消融实验。分别对比了不使用依存结构信息（without-structure）和不使用依存中间信息（without- intermediate），实验结果如表5所示。

模型	ACC(%)	F1(%)
Ours (alternate)	76.37	78.03
Ours (without-structure)	75.70	76.78
Ours (without- intermediate)	75.86	77.07

表 5: 依存信息利用对比结果

从表中结果可以看出，去掉依存结构信息和去掉依存中间信息都会带来复述识别精度的明显下降。其中去除依存结构信息带来的性能降低较为明显，表明依存句法信息能够提升语义组合计算模型的性能；去除依存中间信息带来的性能降低表明，我们的语义组合模型能够有效利用依存分析过程中产生的语义表示，对汉字语义进行适当的补充，提升汉字表示包含。

4.3.3 语义组合方法对比

我们在本章所提的基于注意力的语义组合模型中，对字义组合层和句义输出层进行替换，实现了常见的语义组合计算方法，并与本章所提方法在复述识别任务上进行对比。对比的序列化方法包括平均池化（Mean）、基于CNN的方法（CNN）(Hu et al., 2015)和基于LSTM的方法（LSTM）(Sutskever et al., 2014; Cho et al., 2014)，结构化方法包括采用并列化处理的基于树卷积神经网络的方法（Tree-CNN）(Mou et al., 2016)。其中序列化方法将不使用依存分析模型提供信息，结构化方法将依存分析模型的参数学习率设置为0。对比结果如表6所示。

从对比结果可以看出，我们的模型较现有常见的语义组合计算方法，在复述识别任务上的预测准确率和F1值均有较明显的提升。其中较最好的LSTM方法分别提升了0.83%和1.68%。表明我们的方法能够有效地利用依存句法信息和依存中间信息，从而获取更为准确的语义表示，并反应在下游任务中。

此外，我们实现的结构化方法Tree-CNN和我们的管道模型在复述识别任务上性能接近，但较序列化方法的性能有较明显的下降。我们认为这是由于依存分析和复述识别任务的数据领域不一致的问题，对结构化语义组合方法带来的巨大影响，侧面反映了我们提出的联合模型能够有效降低数据领域不一致的问题。

模型	类型	Acc(%)	F1(%)
Baseline (Mean)	序列化	73.21	74.72
CNN		74.84	76.27
LSTM		75.53	76.31
Tree-CNN	结构化（管道）	72.93	75.46
Ours (pipeline)		72.64	75.74
Ours (alternate)	结构化（联合）	**76.37**	**78.03**

表 6: 语义组合计算方法对比结果

4.3.4 与现有复述识别模型的对比

我们与现在常用的复述识别模型(Wang et al., 2017; Devlin et al., 2019)在LCQMC数据集上的最好结果进行了比较，结果如表7所示。

模型	ACC(%)	F1(%)
BiMPM	83.4	**85.0**
Bert-large	**87.3**	-
Ours (alternate)	76.37	78.03

表 7: 复述识别性能对比

对比结果表明，我们的模型较现有复述识别方法有十分明显的差距。经过分析，我们认为这主要由以下原因导致：1）现在主流复述识别方法主要使用交叉注意力机制（Cross-Attention）进行字义表示的学习，即对两个句子中的字向量进行注意力机制的计算，这样能够对两个句子的相关信息进行利用，已有工作(Liu et al., 2018)表明交叉注意力机制对复述识别任务的性能能够带来显著提升。本章中复述识别任务的主要目的在于对语义组合计算模型的性能进行评价，因此未使用交叉注意力机制，仅对单句信息进行利用。2）以Bert为主的预训练语言模型能够显著提升字向量在具体句子中语义表示的精度，并且能够在大规模训练集中提取丰富的语言学知识，以此提升下游任务的性能。本章主要针对语义组合计算模型中的结构化信息利用和模型联合方法进行改进，仅使用word2vec预训练的静态字向量，且训练数据较小。今后可以尝试引入预训练语言模型，进一步验证我们所提方法的有效性。

5 总结

本文针对现有结构化语义组合计算方法的不足，提出联合依存分析的语义组合计算方法，在现有依存分析模型的基础上，使用图注意力网络根据依存句法树进行语义组合计算，并利用依存分析中间信息对语义组合计算的字义表示进行补充，提升模型的鲁棒性。今后我们将在现有模型中加入预训练语言模型，提升字义表示的性能，以此来进一步提升语义组合计算模型的性能。

参考文献

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1657–1668. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*

2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 1724–1734. ACL.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2015. Convolutional neural network architectures for matching natural language sentences. *CoRR*, abs/1503.03244.

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.

Xuezhe Ma, Zecong Hu, Jingzhou Liu, Nanyun Peng, Graham Neubig, and Eduard H. Hovy. 2018. Stack-pointer networks for dependency parsing. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1403–1414. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

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, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. The Association for Computer Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In Carles Sierra, editor, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 4144–4150. ijcai.org.

Xiao-Dan Zhu, Parinaz Sobhani, and Hongyu Guo. 2015. Long short-term memory over recursive structures. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1604–1612. JMLR.org.