# 融合多粒度特征的低资源语言词性标记和依存分析联合模型

陆杉[1,2]，毛存礼[*1,2]，余正涛[1,2]，高盛祥[1,2]，黄于欣[1,2]，王振晗[1,2]
1.昆明理工大学，信息工程与自动化学院，昆明，650500
2.昆明理工大学，云南省人工智能重点实验室，昆明，650500
lushan88d@163.com, maocunli@163.com, ztyu@hotmail.com
gaoshengxiang.yn@foxmail.com, huangyuxin2004@163.com, 243836042@qq.com

## 摘要

研究低资源语言的词性标记和依存分析对推动低资源自然语言处理任务有着重要的作用。针对低资源语言词嵌入表示，已有工作并没有充分利用字符、子词层面信息编码，导致模型无法利用不同粒度的特征，对此，提出融合多粒度特征的词嵌入表示，利用不同的语言模型分别获得字符、子词以及词语层面的语义信息，将三种粒度的词嵌入进行拼接，达到丰富语义信息的目的，缓解由于标注数据稀缺导致的依存分析模型性能不佳的问题。进一步将词性标记和依存分析模型进行联合训练，使模型之间能相互共享知识，降低词性标记错误在依存分析任务上的线性传递。以泰语、越南语为研究对象，在宾州树库数据集上，提出方法相比于基线模型的UAS、LAS、POS均有明显提升。

**关键词：** 低资源语言；词性标记；依存分析；多粒度特征；联合模型

# A Joint Model with Multi-Granularity Features of Low-resource Language POS Tagging and Dependency Parsing

**Shan Lu[1,2], Cunli Mao[*1,2], Zhengtao Yu[1,2], Shengxiang Gao[1,2], Yuxin Huang[1,2], Zhenhan Wang[1,2]**
1. Faculty of Information Engineering and Automation, Kunming University of Science and Technology Kunming 650500, China
2. Yunnan Key Laboratory of Artificial Intelligence, Kunming University of Science and Technology Kunming 650500, China
lushan88d@163.com, maocunli@163.com, ztyu@hotmail.com
gaoshengxiang.yn@foxmail.com, huangyuxin2004@163.com, 243836042@qq.com

## Abstract

The study of part-of-speech tags and dependency parsing of low-resource languages plays an important role in promoting low-resource natural language processing tasks. For low-resource language word embedding representation, the existing work does not make full use of character and sub-word level information encoding, resulting in models that cannot use features of different granularities. For this reason, a word embedding representation that integrates multi-granularity features is proposed, and different language models are used separately to obtained on different semantic information at the character, sub-word and word level. And three granular word embeddings are combined to achieve the purpose of enriching semantic information and alleviate the problem of poor performance of the dependency parsing model caused by the scarcity of annotation data. The part-of-speech tagging and dependency parsing model are further jointly trained, so that the models can share knowledge with each other and reduce the

*毛存礼(通信作者):maocunli@163.com
国家自然科学基金重点项目(61732005)；国家自然科学基金(61866019，61761026，61972186)；云南省应用基础研究计划重点项目(2019FA023)；云南省中青年学术和技术带头人后备人才项目(2019HB006)

linear transmission of part-of-speech tagging errors on the dependency parsing task. Taking Thai and Vietnamese as the research objects, on the Penn Treebank data set, the proposed method is significantly improved compared to the baseline model UAS, LAS, and POS.

**Keywords:** Low-resource Language , POS Tagging , Dependency Parsing , Multi-granularity Features , Joint Train

## 1 引言

泰语、越南语均属于低资源语言，其相关依存分析研究较少并且效果不佳。大多数传统的依存分析模型都靠人工定义核心的特征工程 (Nivre, 2003; Mcdonald, 2006)，但是这种方法受特征选取的影响较大，随着深度神经网络技术为自然语言处理研究带来崭新建模方式和性能上的巨大提升，基于神经网络的依存分析方法成为研究热点 (Chen and Manning, 2014; Dyer et al., 2015; Kiperwasser and Goldberg, 2016)。

目前，基于神经网络的依存分析主流的方法为基于转移的依存分析 (Dyer et al., 2015)和基于图的依存分析 (Kiperwasser and Goldberg, 2016)。基于图的依存分析的目的是寻找一棵最大生成树，得到句子整体的依存结构全局最优解，该方法对长距离依存分析准确率较高，可处理非投射现象，但模型解码时需进行全局搜索，算法复杂度较高，耗时较长。而基于转移的依存分析将句子的解码过程建模为一个有限自动机问题，使模型可以达到线性时间复杂度，但其采用的是局部搜索策略，容易出现错误传递现象，且准确率要低于基于图的依存分析方法。

无论是基于图的方法还是基于转移的方法，编码层都只使用了简单的词向量表示，如图1所示，泰语句子在进行依存分析的过程中仅仅利用了词语的语义，然而，泰语是由字符、子词以及词语三种粒度组成，将三种不同粒度的表征结合能从各个层面更好的表征其语义信息，另外，基于深度学习的方式训练模型在一定程度上依赖于训练数据的规模，所以，针对标注数据充足的语言时往往都能取得较好的效果，但针对低资源语言，如泰语、越南语，模型获得的效果就不太理想，且现有方法处理泰语、越南语依存分析时，使用的词性标记信息都是和依存分析任务分开处理得到的，词性标记和依存分析作为独立的任务单独训练会导致其任务之间特征信息传递不连贯，增加词性标记错误在依存分析任务上的传递。

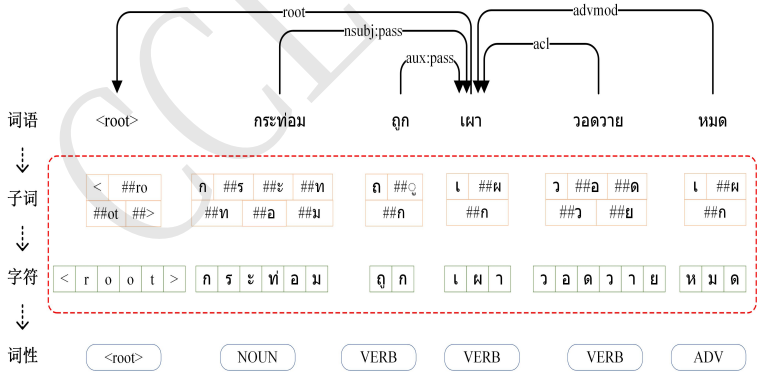


图 1: 泰语依存分析结果示例(汉语释义:小屋被烧毁了。)

针对上述问题，本文提出融合多粒度特征的词性标记和依存分析联合模型。与中文、英文等拥有丰富标注数据的语言不同，泰语、越南语的公开依存分析标注数据分别仅有1000条和3000条。为缓解泰语、越南语标注数据资源不足问题，通过从维基百科获取的大规模单语语料使用Word2Vec (Mikolov et al., 2013)训练词向量来表征词级嵌入，获得词之间丰富的相似性特征信息；利用Jacob Devlin等人 (2019)提出的一种预训练语言模型—BERT并结合层注意力机制来表征词语的子词级嵌入，使词语的子词表征能包含丰富的上下文信息，并充分吸收预训练

模型中的词性、句法等信息 (Jawahar et al., 2019)；通过BiLSTM来编码表征词语的字符级嵌入，使字符级表征拥有丰富的词法信息 (Vania et al., 2018)；并把它们拼接作为联合嵌入使最终嵌入拥有更加丰富的语义、上下文和句法等信息，有效缓解了由于训练数据不足导致的模型性能不佳的问题。最后，通过联合训练的方式，使词性标记和依存分析组件相互共享知识，缓解依存分析和词性标记任务之间错误传递和不连贯性问题。

本文主要做出以下贡献：

（1）利用多粒度特征联合嵌入的方式，在各个粒度嵌入上使用相应方法，使词嵌入拥有丰富的上下文语义信息，词法、句法信息，有效缓解了泰语、越南语标注数据稀缺的问题。

（2）通过联合训练的方式，使词性标记和依存分析模型之间能相互共享知识，缓解了单独训练导致的任务之间错误传递问题，提高了模型整体性能。

（3）在宾州树库[-1]泰语和越南语数据集上，本文提出的方法取得了明显的效果，相较于基线模型，在POS，UAS，LAS三种评价指标上都得到了明显提升。

本文组织结构如下：第1章介绍了词性标记、依存分析的相关研究工作；第2章对本文提出的融合多粒度特征的词性标记和依存分析联合模型进行了详细说明；第3章对本文实验数据、实验参数、实验评价标准进行介绍，并对实验结果进行分析；第4章对本文进行的研究进行了总结。

## 2 相关研究

词性标记和依存分析是自然语言处理任务中重要的基础工作。词性标记是将语料库内单词的词性按其含义和上下文内容进行标记的文本数据处理技术。Toutanova (2003)提出使用隐马尔可夫模型来做词性标记，其词性标记模型取得了很好的效果。Tsuboi等人 (2014)提出一种使用神经网络的词性标记方法，在英语数据集上词性标记结果明显改善。Huang等人 (2015)研究表明基于BiLSTM和CRF的词性标记模型在增强鲁棒性的同时还能提高词性标记的准确率。Kann等人 (2018)提出一种使用词的字符特征作为监督信号提升低资源语言词性标注效果的模型。

依存分析的目的是确定句子的句法结构或者句子中词汇之间的依存关系。传统的依存句法分析特征向量稀疏，特征向量泛化能力差，且计算消耗大 (Nivre, 2003; Mcdonald, 2006)，针对此问题Chen和Manning (2014)提出使用神经网络的方法做依存分析，大大提高了依存分析的准确率和速度。Kiperwasser (2016)提出使用BiLSTM来改进依存分析效果，通过BiLSTM编码过后的句子会考虑词的上下文信息，这项研究使依存分析的效果再次得到提升。同年，Dozat和Manning (2016)对Kiperwasser等人提出的方法加以改进，提出使用双仿射注意力机制代替传统机制，再使用双仿射依存标签分类器，使依存分析准确率达到新的高度。而后，Woraratpanya (2019)提出融合字符信息的泰语依存分析方法，在其实验的所有基线模型中，融合了字符信息的依存分析模型效果均要好于没有融合字符的。

词性标记对依存分析起着重要作用，而依存分析同样也对词性标记有着帮助，所以，越来越多研究者把词性标记和依存分析通过联合训练的方法一起训练。Hatori (2011)提出一种词性标记和依存分析的增量联合模型，在中文数据集上达到了当时最佳的效果。Dat等人 (2018)提出一种融入字符信息的词性标记和依存分析联合模型，其效果在各项评价指标上均得到提升。Dat等人 (2018)还提出一种针对于越南语的神经联合模型，根据越南语语言特点，利用越南语的音节信息对越南语进行分词、词性标记以及依存分析处理，在越南语数据及上取得了较好的效果。Yan等人 (2020)认为依存分析是在单词级别进行的任务，故提出一种基于图的中文分词和依存分析联合模型，其效果达到了中文依存分析最佳。

虽然，词性标记和依存分析联合训练已成为依存分析任务的主流方法，但是，现有的词性标记和依存分析联合模型的良好效果大都基于大规模的标注数据或针对某种语言的语言特点进行相关特征融合。模型本身并不适用于低资源语言，以至于在低资源语言上，模型效果不佳，基于此，本文提出了融合多粒度特征的词性标记和依存分析联合模型。

## 3 多粒度特征融合的词性标记和依存分析联合模型

在本节中，图2为提出方法的模型框架，模型从整体上可以被看作是由三个部分组合而成：词向量表示部分、词性标记部分以及依存分析部分。

[-1]https://universaldependencies.org/

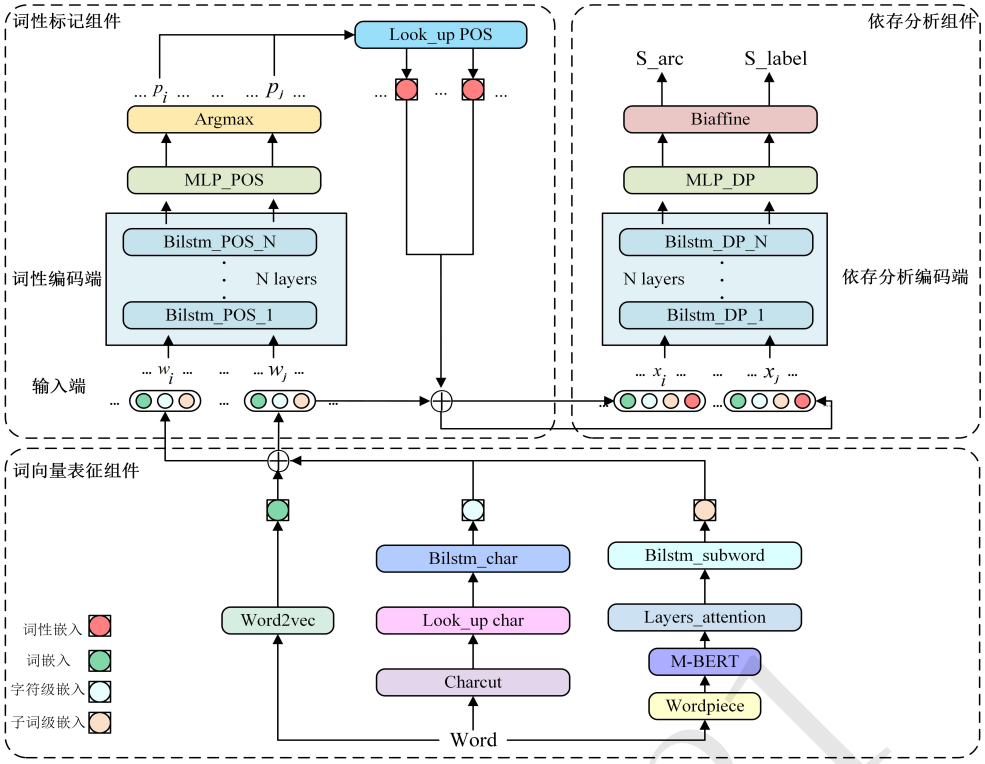


图 2: 融合多粒度特征的词性标记和依存分析联合模型框架图

（1）**词向量表示**：将词向量表示用三部分构成，分别为词级向量、字符级词向量和子词级词向量，使其包含不同粒度的丰富上下文语义信息和部分词法、句法信息。

（2）**词性标记**：对于词性标记任务，使用BiLSTM网络来学习表示词语的潜在特征向量，再将这些特征向量送入多层感知器（MLP）进行降维后通过Argmax预测得到词性标签。

（3）**依存分析**：依存分析部分使用词的联合嵌入向量拼接词性标记组件预测得到的词性标签向量，通过另一个BiLSTM学习另一组潜在特征表示。这些潜在特征表示被送入到MLP进行降维后经过双仿射注意力机制 (Dozat and Manning, 2016)预测得到最终的依存分析结果。

### 3.1 多粒度特征融合的词向量表示

给定一句输入含有$n$个单词的泰语句子$\mathrm{Sen} = \mathrm{w}_1, \mathrm{w}_2, ..., \mathrm{w}_\mathrm{n}$，我们用$\mathbf{w}_j$来表示句子Sen中第$j$个单词$w_j$的向量表示。因泰语、越南语标注数据稀缺，模型训练过程中能学习到的知识相对较少，所以我们通过拼接词嵌入$\mathbf{e}^w{}_j$、字符级词嵌入$\mathbf{e}^c{}_j$和子词级词嵌入$\mathbf{e}^s{}_j$来表征$\mathbf{w}_j$，使联合词嵌入$\mathbf{w}_j$拥有丰富的上下文语义信息和词语之间的相似性特征，同时也能学习得到部分词法信息，如公式(1)所示：

$$\mathbf{w}_j = \mathbf{e}^w{}_j \oplus \mathbf{e}^c{}_j \oplus \mathbf{e}^s{}_j \tag{1}$$

其中，任意一个词w的词嵌入表示为$\mathbf{e}_w$，由k个子词组成,其表示为$\mathrm{w} = \mathrm{sub}_1, \mathrm{sub}_2, ..., \mathrm{sub}_\mathrm{k}$，由n个字符组成，其表示为$\mathrm{w} = \mathrm{ch}_1, \mathrm{ch}_2, ..., \mathrm{ch}_\mathrm{n}$。对词w中第j个子词$\mathrm{sub}_\mathrm{j}$的向量用$\mathbf{s}_j$来表示，第j个字符$\mathrm{ch}_\mathrm{j}$的向量用$\mathbf{c}_j$来表示。向量$\mathbf{c}_j$是由随机初始化得到。而向量$\mathbf{s}_j$是由多语言BERT[0]预训练模型得到的12层输出再通过层注意力机制得到 (Kondratyuk and Straka, 2019)，其中，预训练语言模型可使$\mathbf{s}_j$拥有丰富的上下文语义信息，再结合层注意力机制使其重点获取到对后续任务更加有帮助的上下文语义信息，以达到缓解泰语、越南语标注数据稀缺的问题，如公式（2）所示:

$$\mathbf{s}_j = \sum_{i=1}^{12} \mathrm{Bert}_{\mathrm{ij}} \times softmax(u_i) \tag{2}$$

[0]https://github.com/google-research/bert

上述公式中，$\mathrm{Bert_{ij}}$表示BERT第$i$层的第$j$个子词的输出，$u_i$是可训练的权重。

接着，分别使用$\mathrm{BiLSTM_{sub}}$和$\mathrm{BiLSTM_{char}}$来进一步学习得到子词级向量$\mathbf{e}^s{}_j$的表示和字符级向量$\mathbf{e}^c{}_j$的表示，其中$\mathrm{BiLSTM_{sub}}$的输入为k个子词向量表示$\mathbf{s}_{1:k}$，输出为LSTM正向和反向最后时刻隐状态的拼接，如公式（3）所示：

$$\mathbf{e}^s{}_j = \mathrm{BiLSTM_{sub}}(\mathbf{s}_{1:k}) \tag{3}$$

$\mathrm{BiLSTM_{char}}$的输入为n个字符向量表示$\mathbf{c}_{1:n}$，输出同样为LSTM正向和反向最后时刻隐状态的拼接，通过模型训练，可使$\mathbf{e}^c{}_j$学习得到词法信息，帮助后续词性标记和依存分析提升效果，如公式（4）所示：

$$\mathbf{e}^c{}_j = \mathrm{BiLSTM_{sub}}(\mathbf{c}_{1:n}) \tag{4}$$

### 3.2 词性标记

在词性标记部分，本文使用的依存分析数据中，所有的数据都具备词性标记的属性，我们将输入句子Sen含有$n$个单词的词性标记标签向量序列表示为$\mathbf{e}^{pos}_{1:n}$，其中$\mathbf{e}^{pos}_{1:n} = e^{pos}_1, e^{pos}_2, ..., e^{pos}_n$。我们将向量序列$\mathbf{e}^{pos}_{1:n}$输入到$\mathrm{BiLSTM_{pos}}$产生输入句子中的第i个词的词性潜在特征向量$\mathbf{r}^{pos}_i$，如公式（5）表示：

$$\mathbf{r}^{pos}_i = \mathrm{BiLSTM_{pos}}(\mathbf{e}^{pos}_{1:n}, i) \tag{5}$$

我们使用$\mathrm{MLP_{pos}}$和Argmax组合的分类器对$\mathbf{h}^{pos}_i$进行预测，先将$\mathbf{h}^{pos}_i$输入到$\mathrm{MLP_{pos}}$得到维度大小为词性标签种类数的输出，再使用Argmax预测出词性标签。

词性预测部分的损失$\mathrm{Loss_{pos}}$采用交叉熵损失函数来计算。

此部分获得的词性标记的结果定义为$\mathrm{p_1, p_2, ..., p_n}$，将这些结果进行向量化表示，所获得的词性标记部分特征信息继续传递给依存分析部分，如上图2所示。

### 3.3 基于双仿射注意力机制的依存分析

通过联合模型的词性标记部分获得了词性标记的结果，通过词性标记结果得到我们所需的对应词性向量，我们使用$\mathbf{e}^p_i$表示第j个词的词性向量表示，拼接词的词性向量和上述公式（1）中得到的词的联合嵌入$\mathbf{w}_j$作为输入送入依存分析部分，其中第j个单词的输入表示为$\mathbf{x}_j$，如公式（6）：

$$\mathbf{x}_j = \mathbf{e}^p_i \oplus \mathbf{w}_j \tag{6}$$

同样的，我们通过另一个$\mathrm{BiLSTM_{dep}}$得到输入句子中的第i个词的潜在特征向量$\mathbf{r}^{dep}_i$表示，如公式（7）所示。

$$\mathbf{r}^{dep}_i = \mathrm{BiLSTM_{dep}}(\mathbf{x}_{1:n}, i) \tag{7}$$

对于潜在特征向量$\mathbf{r}^{dep}_i$，我们分别通过两个$\mathrm{MLP}^{arc}_d$，$\mathrm{MLP}^{arc}_h$得到第$i$个词作为依赖的表征$\mathbf{d}^{arc}_i$和头的表征$\mathbf{h}^{arc}_i$。再通过双仿射注意力机制 (Dozat and Manning, 2016)计算得到词$w_i$到$w_j$的弧的概率得分$s^{arc}_{ij}$，如公式（8）所示：

$$\mathbf{s}^{arc}_{ij} = \mathrm{Biaffine}^{arc}(\mathbf{h}^{arc}_i, \mathbf{d}^{arc}_i) = \mathbf{h}^{arc}_j U^{arc} \mathbf{d}^{arc}_i + \mathbf{h}^{arc} u^{arc} \tag{8}$$

其中$U^{arc}$为随机初始化的权重矩阵，维度为（$N_d$，$N_d$），$N_d$为词的联合嵌入维度大小，$u^{arc}$为偏执矩阵。在获得第j个词到所有头节点的分数后，我们选择最大分数节点作为头节点$y^{arc}_i$，如公式（9）（10）所示。

$$\mathbf{s}^{arc}_i = [\mathbf{s}^{arc}_{i1}, \mathbf{s}^{arc}_{i2}, ..., \mathbf{s}^{arc}_{in}] \tag{9}$$

$$y^{arc}_i = \mathrm{Argmax}(\mathbf{s}^{arc}_i) \tag{10}$$

依存弧的损失$\mathrm{Loss_{arc}}$使用交叉熵损失计算。

在获得最佳预测的未标记依存树后，对于任意的两个单词之间的依存关系，我们使用固定类别的仿射分类器进行预测。首先，分别通过两个$\mathrm{MLP}^{label}_d$，$\mathrm{MLP}^{label}_h$，得到第i个词的

表征$\mathbf{d}_i^{label}$和$\mathbf{h}_i^{label}$。给定依存弧$(i,j)$，弧$(i,j)$的依存关系的概率得分$\mathbf{s}_{ij}^{label}$表示如公式（11）所示：

$$\mathbf{s}_{ij}^{label} = \text{Biaffine}^{\text{label}}(\mathbf{h}_i^{label}, \mathbf{d}_i^{label}) = \mathbf{h}_j^{label} U^{label} \mathbf{d}_i^{label} + (\mathbf{h}_j^{label} \oplus d_i^{label}) \mathbf{V}^{label} + b \quad (11)$$

其中，$U^{lablel}$为三维矩阵，维度为（$N^{label}$，$N_d$，$N_d$），$N^{label}$是依存关系种类数。$V^{label}$是维度为$(N^{label}，2N_d)$的二维矩阵，$b$为随机初始化的偏执向量。最终弧$(i，j)$的依存关系预测如公式（12）所示：

$$y_{ij}^{label} = \text{Argmax}(s_{ij}^{label}) \quad (12)$$

依存关系预测的损失同样使用交叉熵损失函数来计算。

### 3.4 联合模型损失

最终，我们将联合模型的训练目标损失函数表示为$\text{Loss}_{\text{all}}$，联合函数的损失是由词性标记损失，依存分析中的依存弧损失和依存关系损失共同表示，如公式（13）所示，其中$\lambda_1$，$\lambda_2$，$\lambda_3$ 为超参数：

$$\text{Loss}_{\text{all}} = \lambda_1 \text{Loss}_{\text{pos}} + \lambda_2 \text{Loss}_{\text{arc}} + \lambda_3 \text{Loss}_{\text{rel}} \quad (13)$$

## 4 实验评测与结果分析

### 4.1 实验数据

目前，泰语、越南语公开的语料数据集极少，可用的数据资源极其稀缺，实验中使用的数据集为宾州树库公开泰文依存分析数据集Thai-PUD和越南语依存分析数据集Vietnamese-VTB，其采用CoNLL-U格式，泰语包含1000个句子，越南语包含3000个句子。其中泰语数据中一共包含22,322个词语，越南语数据由43,754个词语组成。通过分析数据集可以得到，泰语、越南语数据集中依存关系类型分别有43种、29种。如图3、4所示，泰语句子中词语词性类型共包含有15种，其中词语词性为名词，动词，副词的数量最多。而越南语句子中词语词性类型共包含有14种，其中词语词性为名词，动词，标点的数量最多

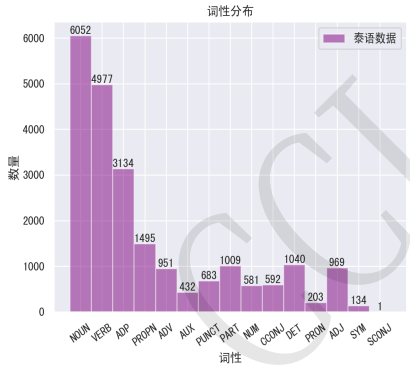


图 3: 泰语数据词性分布情况

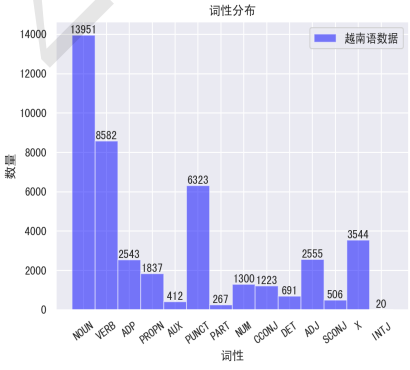


图 4: 越南语数据词性分布情况

实验所用数据集中泰语、越南语句子以复杂句和长句和简单句组成 (Singkul and Woraratpanya, 2019)，具体分布如表1、2所示。其中，词语数在8个以内的句子为简单句，词语数在8-16个之间的句子为长句，大于16个词语的句子称为复杂句。

本文实验中泰语实验所用训练集，验证集，测试集由宾州树库的1000句泰语依存分析数据按8:1:1分割所得，800句用作训练集，100句用作验证集，100句用作测试集。越南语实验使用宾州树库划分好的数据集，其中训练集1400句，验证集800句，测试集800句。

### 4.2 实验参数设置

本文使用的泰语词向量是通过维基百科[1]爬取的1,000,000句泰语单语语料经过分词[2]后使用Word2vec生成的100维静态词向量，越南语词向量是通过维基百科爬取的1,000,000句越南语

[1]https://th.wikipedia.org/wiki/
[2]http://www.sansarn.com/lexto/

| 句子种类 | 句子数量 | 词数量 |
|---|---|---|
| 简单句 | 25 | 136 |
| 长句 | 207 | 2,509 |
| 复杂句 | 768 | 19,677 |

表 1: 泰语数据统计

| 句子种类 | 句子数量 | 词数量 |
|---|---|---|
| 简单句 | 171 | 737 |
| 长句 | 1501 | 16,333 |
| 复杂句 | 1328 | 26,684 |

表 2: 越南语数据统计

单语语料经过Vncorenlp[3]分词后使用Word2vec生成的100维静态词向量。字符初始向量表示和词性标签向量表示是由随机初始化得到。本文模型参数的具体细节如下。

| 参数 | 大小 | 参数 | 大小 |
|---|---|---|---|
| $\text{MLP}_{\text{Arc}}$ size | 400 | BiLSTM hidden size | 500 |
| $\text{MLP}_{\text{Lab}}$ size | 200 | Arc MLP dropout | 0.25 |
| cEmbedding | 100 | Lab MLP dropout | 0.25 |
| wEmbedding | 100 | $\beta_1$,$\beta_2$ | 0.9,0.99 |
| sEmbedding | 100 | $\lambda_1$,$\lambda_2$,$\lambda_3$ | 0.2,0.6,0.2 |
| pEmbedding | 100 | LSTM dropout | 0.33 |
| LSTM depth | 3 | Emb dropout | 0.1 |
| MLP depth | 1 | Epoch | 30 |

表 3: 实验超参数设置表

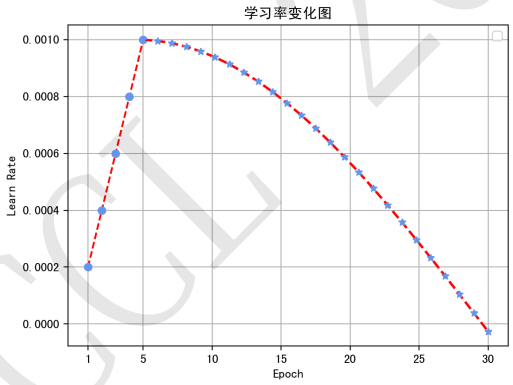


图 5: 学习率变化图

模型优化器选用Adam，其中$\beta_1$设置为0.9，$\beta_2$设置为0.99，如表3所示，训练轮次为30轮。模型学习率初始设置为$2e^{-4}$，经过五轮预热，每轮增加$2e^{-4}$的学习率，直至升至$1e^{-3}$，再使用余弦退火衰减进行调整，余弦退火衰减T设置为25，学习率变化如图5所示。模型使用的LSTM层数为三层，词向量维度设置为100维，子词向量维度设置为100维，词性向量维度设置为100维，LSTM维度设置为500维，依存分析弧预测部分MLP隐藏层维度设置为500，依存分析弧关系预测部分MLP隐藏层维度设置为200，弧预测和弧关系预测部分MLP层数均为1。损失函数超参$\lambda_1$,$\lambda_2$,$\lambda_3$分别设置为0.2，0.6，0.2。为了防止过拟合问题，我们使用了Dropout正则化 (Srivastava et al., 2014)技术，词性预测模型的词语向量输入层、依存分析模型的词语向量输入层的Dropout概率均设置为0.1，词性预测模型的BiLSTM、依存分析模型的BiLSTM中Dropout概率均设置为0.33，弧预测和弧关系预测部分MLP的Dropout概率均设置为0.25。文中模型中所使用的激活函数均为LeakyReLU激活函数。

[3]https://github.com/dnanhkhoa/python-vncorenlp

### 4.3 实验评价指标

目前依存分析任务的评价指标主要是无标签依存关系准确率（UAS）和带标签依存关系准确率（LAS），词性标注任务的评价指标通常是词性准确率（POS）。本文是基于词性标注和依存分析的联合模型，所以评价本文实验的评价标准选取UAS、LAS、POS三种评价指标来评测模型的性能，具体公式如（14）、（15）、（16）所示：

$$UAS = \frac{\text{依存弧正确的词数}}{\text{所有词数}} \times 100\% \tag{14}$$

$$LAS = \frac{\text{依存弧正确并且依存关系正确的词数}}{\text{所有词数}} \times 100\% \tag{15}$$

$$POS = \frac{\text{词性正确的词数}}{\text{所有词数}} \times 100\% \tag{16}$$

### 4.4 实验结果分析

为了体现本文所提出的方法有效性，本文设计了三组对比实验：

**实验一:不同模型方法的实验结果对比**

为了验证本文方法的有效性，将本文方法与其他相关模型进行对比。记录每组实验的UAS、LAS、POS，实验结果如表4所示。

本文选取对比的基线模型如下：

（1）**BIST-graph** (Kiperwasser and Goldberg, 2016)：由Kiperwasser等人在2016年提出的一种使用BiLSTM特征表示的基于图的依存分析模型。

（2）**BIST-transition** (Dyer et al., 2015)：由Dyer等人在2016年提出的一种使用Stack-LSTM的基于转移的依存分析模型。

（3）**Deep Biaffine Attention** (Dozat and Manning, 2016):由Manning等人在2016年提出的一种双仿射注意力机制依存分析模型。

（4）**UDPipe** (Zeman et al., 2018)：由Milan Straka等人2018年提出的一种词性标记、依存分析的多任务模型。

（5）**UDify** (Kondratyuk and Straka, 2019)：由Dan等人2019年提出的一种基于Bert实现的词性标记、依存分析的多任务模型。

（6）**JPTDP2.0** (Nguyen and Verspoor, 2018)：由Dat等人2018年提出的一种联合词性的神经网络依存分析模型。

| Treebank | Model | UAS（%） | LAS（%） | POS（%） |
|---|---|---|---|---|
| Thai-PUD | BIST-graph | 83.13 | 75.64 | / |
| | BIST-transition | 80.78 | 73.82 | / |
| | Deep Biaffine Attention | 86.32 | 78.47 | / |
| | JPTDP2.0 | 82.73 | 74.61 | 92.71 |
| | **本文方法** | **86.84** | **78.87** | **95.19** |
| Vietnamese-VTB | BIST-graph | 74.77 | 70.96 | / |
| | BIST-transition | 73.02 | 68.54 | / |
| | Deep Biaffine Attention | 75.83 | 72.16 | / |
| | UDPipe | 70.38 | 62.56 | 89.68 |
| | UDify | 74.11 | 66.00 | 91.29 |
| | JPTDP2.0 | 67.72 | 58.27 | 87.63 |
| | **本文方法** | **76.62** | **73.84** | **93.30** |

表 4: 不同模型实验结果

实验结果表明，本文提出的融合多粒度特征的词性标记和依存分析联合模型，在泰语数据集上，UAS、LAS和POS分别较基线模型JPTDP2.0提升了4.11%、4.26%、2.32%，在越南语数据集上，各项评价指标也较其它基线模型有明显提升。通过结果可知，针对泰语、越南语这种低资源语言，融合多粒度特征后词性标记任务和依存分析任务的词向量表示都拥有了更加丰富的语义信息，弥补了因资源稀缺导致的模型吸收语义知识不足的问题，且联合训练大大缓解了词性标记和依存分析任务之间的错误传递，共享了信息，对依存分析和词性标注效果都有明显的提升。

**实验二：不同BERT微调策略的实验结果对比**

为了验证本文方法的有效性并研究在多语言Bert模型的12层输出上使用不同策略作为子词向量表征对实验结果的影响，本文选用泰语数据在子词向量表征分别选取Bert输出的1-4层求和、4-8层求和、8-12层求和、4-12层求、单独使用第12层和对12层输出使用层注意力机制的结果进行对比，记录每组实验的UAS、LAS和POS，实验结果如表5所示。

| Layers | UAS（%） | LAS（%） | POS（%） |
|---|---|---|---|
| Sum 1-4 | 85.73 | 77.65 | 94.98 |
| Sum 4-8 | 86.03 | 78.41 | 95.08 |
| Sum 8-12 | 86.12 | 78.46 | 95.17 |
| Sum 12 | 86.15 | 78.34 | 95.12 |
| Sum 4-12 | 86.31 | 78.71 | 95.23 |
| **Layers attention** | **86.84** | **78.87** | **95.19** |

表 5: 不同策略的Bert使用情况对实验结果的影响

实验结果表明，采用不同策略使用Bert的12层输出对模型性能有着较大的影响。当使用Bert输出的1-4层，4-8层时模型在三种评价指标上均低于使用8-12层，4-12层和单独使用12层，可知Bert的12层输出中不同层数的向量对依存分析和词性标记有着不同的影响。其中，使用4-12层的求和做为子词向量表征时词性标记的结果最高，达到95.23%，可知后8层的输出对词性标记有着重要影响。而对Bert的12层输出使用层注意力机制取最后加权结果做为子词向量表征时UAS和LAS达到最高的86.84%和78.87%，可知让模型在训练过程中自主学习对Bert各层输出的权重能使模型达到较好的效果。

**实验三：消融实验**

为了验证不同粒度的联合嵌入的效果，本文使用泰语数据设计了使用词本身嵌入，使用字符和词的联合嵌入，使用子词和词的联合嵌入，使用词、子词和字符的联合嵌入四种不同实验进行对比，记录每组实验的UAS、LAS和POS值，实验结果如表6所示。

| | UAS（%） | LAS（%） | POS（%） |
|---|---|---|---|
| 词 | 85.37 | 77.23 | 94.21 |
| 字符+词 | 85.75 | 77.92 | 94.85 |
| 子词+词 | 85.93 | 78.33 | 94.82 |
| **子词+字符+词** | **86.84** | **78.87** | **95.19** |

表 6: 不同粒度联合嵌入对实验结果的影响

本组实验证明，子词、字符与词的联合嵌入作为实验输入相比于字符和词的联合嵌入、子词和词的联合嵌入和仅使用词嵌入本身在各项评价指标上都有更好的效果。字符为词语的最小粒度，其词表很小并不能充分包含上下文信息，且字符切分包含了大量冗余信息。子词切分的粒度在词语与字符之间，其语义表示相比于字符更加充分，相比于词语更加细腻，所以其与词的联合嵌入效果比字符与词的联合嵌入效果更好。而把三种不同粒度的表示做为联合嵌入，更

能使词表征获得各个层面上的语义信息，所以使用子词、字符加上词的联合嵌入在各项评价指标上都获得了最好的结果。

### 4.5 不同句子类型结果分析

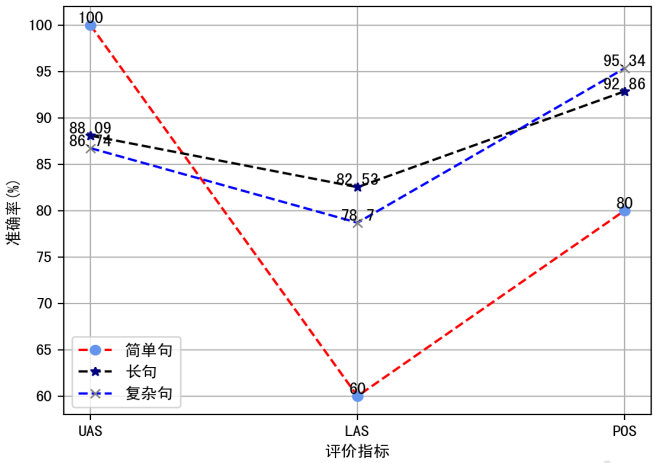


图 6: 泰语不同句子类型结果分析

本文实验中把1000句泰语数据集按照8:1:1随机切分为训练集、验证集和测试集合。其中测试集中包含1句简单句，10句长句和89句复杂句。观察测试集结果，如图6所示，因简单句只有1句,数据样本太少，故不做详细分析。长句有10句，包含126个词语，其中UAS、LAS和POS分别为88.09%、82.53%、92.86%。复杂句一共包含2301个词语，其中UAS、LAS和POS分别为86.74%、78.70%、95.34%。通过上述结果可以得知，模型在对复杂句进行词性预测时，因上下文更加充分，其效果要好于其它类型句子。而对长句预测的UAS和LAS要明显高于平均值，可知模型对复杂句的句法解析效果不如对长句的解析效果。

## 5 总结

针对于泰语、越南语因标注数据稀缺导致的词性标记和依存分析效果不佳问题，本文提出一种针对低资源语言的融合多粒度特征的词性标记和依存分析联合模型。通过不同方法得到字符级、子词级和词级表征，并把它们进行联合嵌入使得编码端能拥有不同层面丰富的形态特征信息、上下文信息和相似性特征信息，有效缓解了标注数据稀缺导致的模型效果不佳问题。再结合联合模型，使词性标记和依存分析任务之间相互共享知识，有效减少单独训练各任务出现的错误线性传递问题。我们的模型有效提升词性标记以及依存分析任务的效果。今后的研究中，我们会将分词任务一同融入到所提出的模型框架中来进行联合训练，探究低资源语言中分词、词性标记、依存分析组件之间能否更加有效的共享知识，达到提升依存分析效果的目的。

## 参考文献

Danqi Chen and Christopher D Manning. 2014. A fast and accurate dependency parser using neural networks. pages 740–750.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A Smith. 2015. Transition-based dependency parsing with stack long short-term memory. *arXiv: Computation and Language*.

Jun Hatori, Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. 2011. Incremental joint pos tagging and dependency parsing in chinese. In *Proceedings of 5th international joint conference on natural language processing*, pages 1216–1224.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does bert learn about the structure of language? In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*.

Katharina Kann, Johannes Bjerva, Isabelle Augenstein, Barbara Plank, and Anders Søgaard. 2018. Character-level supervision for low-resource pos tagging. In *Proceedings of the Workshop on Deep Learning Approaches for Low-Resource NLP*, pages 1–11.

Eliyahu Kiperwasser and Yoav Goldberg. 2016. Simple and accurate dependency parsing using bidirectional lstm feature representations. *Transactions of the Association for Computational Linguistics*, 4(1):313–327.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing universal dependencies universally. *arXiv preprint arXiv:1904.02099*.

Ryan Mcdonald. 2006. *Discriminative Learning and Spanning Tree Algorithms for Dependency Parsing*. Ph.D. thesis, USA. AAI3225503.

Tomas Mikolov, Kai Chen, Greg S Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Dat Quoc Nguyen and Karin Verspoor. 2018. An improved neural network model for joint pos tagging and dependency parsing. *arXiv preprint arXiv:1807.03955*.

Dat Quoc Nguyen. 2018. A neural joint model for vietnamese word segmentation, pos tagging and dependency parsing. *arXiv preprint arXiv:1812.11459*.

Joakim Nivre. 2003. An efficient algorithm for projective dependency parsing. In *Proceedings of the Eighth International Conference on Parsing Technologies*, pages 149–160, Nancy, France, April.

Sattaya Singkul and Kuntpong Woraratpanya. 2019. Thai dependency parsing with character embedding. In *2019 11th International Conference on Information Technology and Electrical Engineering (ICITEE)*, pages 1–5. IEEE.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 252–259.

Yuta Tsuboi. 2014. Neural networks leverage corpus-wide information for part-of-speech tagging. pages 938–950.

Clara Vania, Andreas Grivas, and Adam Lopez. 2018. What do character-level models learn about morphology? the case of dependency parsing. *arXiv preprint arXiv:1808.09180*.

Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2020. A graph-based model for joint chinese word segmentation and dependency parsing. *Transactions of the Association for Computational Linguistics*, 8:78–92.

Daniel Zeman, Jan Hajic, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. Conll 2018 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual parsing from raw text to universal dependencies*, pages 1–21.