

厦门大学学报(自然科学版)
Journal of Xiamen University(Natural Science)
ISSN 0438-0479,CN 35-1070/N

《厦门大学学报(自然科学版)》网络首发论文

题目：基于 tri-training 的跨领域依存句法分析
作者：李帅克，李英，李正华，张民
收稿日期：2021-10-27
网络首发日期：2022-05-10
引用格式：李帅克，李英，李正华，张民. 基于 tri-training 的跨领域依存句法分析[J/OL]. 厦门大学学报(自然科学版).
https://kns.cnki.net/kcms/detail/35.1070.N.20220510.1157.004.html

网络首发：在编辑部工作流程中，稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定，且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式（包括网络呈现版式）排版后的稿件，可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定；学术研究成果具有创新性、科学性和先进性，符合编辑部对刊文的录用要求，不存在学术不端行为及其他侵权行为；稿件内容应基本符合国家有关书刊编辑、出版的技术标准，正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性，录用定稿一经发布，不得修改论文题目、作者、机构名称和学术内容，只可基于编辑规范进行少量文字的修改。

出版确认：纸质期刊编辑部通过与《中国学术期刊（光盘版）》电子杂志社有限公司签约，在《中国学术期刊（网络版）》出版传播平台上创办与纸质期刊内容一致的网络版，以单篇或整期出版形式，在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊（网络版）》是国家新闻出版广电总局批准的网络连续型出版物（ISSN 2096-4188，CN 11-6037/Z），所以签约期刊的网络版上网络首发论文视为正式出版。

基于 tri-training 的跨领域依存句法分析

李帅克，李英，李正华✉，张民

（苏州大学计算机科学与技术学院，江苏 苏州 215006）
*通信作者 zhli13@suda.edu.cn

摘要 基于大规模领域内标注数据训练的句法分析模型在领域外数据上测试时，性能会急剧下降。导致该现象的原因之一是缺乏高质量的目标领域标注数据。由于人工标注数据耗时耗力，自动生成目标领域标注数据是一种有效的解决方法。其中，tri-training 作为一种典型的多模型决策协同训练方法，旨在利用多个模型的预测结果来保证自动标注数据的质量。本文针对跨领域依存句法分析任务，系统比较了三种常用的 tri-training 方法，在 NLPCC-2019 评测数据集上取得了目前最佳性能，并大幅度超过了目前最好结果。此外，还设计了详细的分析实验以深入理解跨领域模型性能下降的原因以及 tri-training 所起的作用。

关键词 自学习；领域移植；句法分析
中图分类号 TP391.1 文献标志码 A DOI: 10.6043/j.issn.0438-0479. 202110034

Tri-training based cross-domain dependency parsing

LI Shuaike, LI Ying, LI Zhenghua✉, ZHANG Min

(School of Computer Science & Technology, Soochow University, Suzhou 215006, China)

Abstract The performance degrades sharply when a parser is trained on a large-scale in-domain labeled data and tested on out-of-domain data. One of the reasons is the lack of high-quality target domain labeled data. Considering manually annotating data is time-consuming and labor-intensive, an effective solution is generating target domain labeled data automatically. Among them, tri-training, as a typical multi-model decision-making collaborative training method, is designed to ensure the quality of auto-labeled data by utilizing the prediction results of multiple models. This paper compares three typical tri-training methods systematically for cross-domain dependency parsing. Our model achieves the state-of-the-art results on NLPCC-2019 shared task datasets, which outperforms the best result by a large margin. In addition, we design detailed analysis to gain more insights on the reasons of performances degradation in cross-domain models and the effect of tri-training.

Key words tri-training; domain adaptation; dependency parsing

依存句法分析把输入句子$s=\{w_0,w_1,\cdots,w_n\}$解析为一棵依存句法树$t=\{(i\rightarrow j,l)|0\leq i\leq n,0<j\leq n,l\in L,\}$，其中$(i\rightarrow j,l)$是一条从核心词（head）$w_i$到修饰词（modifier）$w_j$的依存弧，弧的标签是$l$，$L$为标签集合。在独立同分布假设下，给定大量高质量标注数据，已经存在若干非常有效的有监督算法可以训练出高准确率的句法分析器[1-2]。然而在真实场景中，独立同分布假设往往是不成立的，此时训练数据和测试数据的潜在分布差异会导致训练数据上学习到的模型在测试数据上的性能显著下降。如何解决这一问题是领域移植（domain adaptation）研究的主要目标。本文聚焦于单源零样本跨领域依存句法分析，其中源领域数据为标注数据$S=\{(x_s,y_s)\}_{s=1}^{|S|}$（$x_s$和$y_s$分别为第 s 条数据及其标签），目标领域数据为无标注数据$T=\{x_t\}_{t=1}^{|T|}$，本研究希望利用这些已有数据资源训练一个可以很好地泛化到目标领域的模型。

半监督学习中的自学习方法（self-training[3]）是一种利用标注数据和无标注数据进行模型训练的代表性方法。其核心思想是利用在标注数据上训练得到的模型去自动分析无标注数据，然后选择高置

收稿日期：2021-10-27
基金项目：国家自然科学基金（61876116）
网络首发时间：2022-05-10 17:28:17 网络首发地址：https://kns.cnki.net/kcms/detail/35.1070.N.20220510.1157.004.html

信度的自动标注数据作为额外训练数据。训练流程如下：首先在标注数据上训练一个模型，然后使用这个模型去预测无标注数据，接着把置信度高的自动标注数据加入到有标注数据中，最后利用拼接后的数据重新训练一个模型。重复上述过程直到模型性能不再提升。三元训练（tri-training[4]）与 self-training 的训练方式类似，不过是使用多个模型投票进行自动标注数据的选择而非显式的置信度评估。

然而，直接把自学习方法应用在领域移植上效果往往并不理想。其主要原因是目标领域自动标注数据包含了大量的噪声，此时噪声带来的坏处超过了使用目标领域无标注数据带来的好处，使用该数据反而对模型产生了干扰。因此，如何筛选高质量的自动标注数据是自学习方法的主要挑战。为了解决此问题，前人做了一些探索。Mcclosky 等[5]把 self-training 成功应用到了成分句法分析上，其使用重排序策略获得高置信度的自动标注数据加入到训练集。Kawahara 等[6]通过一个额外的分类器来判别自动标注数据是否可靠成功地把 self-training 应用在了依存句法分析上。Yu 等[7]尝试使用基于句法解析树分值的置信度选择策略和基于 Delta 分值的置信度选择策略进行依存句法分析。他们的实验结果进一步证实了自动标注数据的质量在领域移植场景下依然至关重要，但是这些方法需要显式的置信度评估。Sogaard 等[8]把 tri-training 应用到了依存句法分析上，使用两个模型的预测一致性进行自动标注数据的置信度评估从而简化了训练流程。Li 等[9]运用 tri-training 的同时使用 BERT[10]作为模型的编码层，取得了 NLPCC-2019 句法分析评测开放赛道上的最佳结果。

Tri-training 作为一种简洁的生成自动标注数据的经典方法，目前仍缺乏将其应用于跨领域依存句法分析任务的系统性深入研究。为了解决这个问题，本研究使用 3 种简单的多模型决策协同训练算法（tri-training）来保证自动标注数据的质量进而提升跨领域依存句法分析性能。这 3 种算法分别从横向和纵向两个方面完成自动标注数据的筛选：同一时间轮次下三个模型预测一致的数据、同一时间轮次下当前模型之外的另两个模型预测一致的数据、当前轮次之前两轮的模型预测一致的数据。此外，本研究使用目标领域的无标注数据对 BERT 模型进行微调从而获得领域相关特征以进一步增强模型的编码能力。

为了验证 tri-training 可以提高自动标注数据的质量，进而提升跨领域依存句法分析性能，本研究在 NLPCC-2019 评测任务发布的依存句法分析领域移植数据（http://hlt.suda.edu.cn/index.php/Nlpcc-2019-shared-task）上进行了实验。为了进一步分析跨领域模型性能下降的原因以及 tri-training 和 BERT 为什么可以提升跨领域模型的性能，本文还设计了详细的分析实验。

1 方法

本章节会逐次介绍基础模型框架、多模型决策协同训练策略和预训练语言模型的微调过程。首先使用双仿射句法分析器[1]作为基础模型，然后在此模型的基础上使用多模型决策协同训练策略生成高质量的目标领域自动标注数据，从而提升跨领域模型性能。最后使用目标领域的无标注数据对 BERT 模型进行微调，并将其作为模型的特征进一步提升模型性能。

1.1 双仿射句法分析器

本研究将简明高效的双仿射句法分析器作为基础模型。模型框架如图 1 所示，该句法分析器包含 4 个模块：输入模块、双向长短期记忆网络（BiLSTM）编码模块、多层感知机注意力（MLP）模块和双仿射打分模块。

输入模块：把输入句子映射为向量表示。该模块把输入序列映射为输入向量表示$X=[x_0,x_1,\cdots,x_n]$，每一个输入向量x_i都由对应的词向量和词向量对应的字符集双向长短期记忆网络（CharBiLSTM）表示组成。

$$x_i = e(w_i) \oplus \boldsymbol{C}^{\mathrm{BiLSTM}}(w_i)$$

其中，$e(w_i)$是冻结的预训练词向量与随机初始化可更新词向量的和；$\boldsymbol{C}^{\mathrm{BiLSTM}}(w_i)$是将$w_i$对应的字符输入到一层的双向长短期记忆网络，然后拼接前向、后向最后的隐向量获得的[11]，本研究发现在输入模块中用 CharBiLSTM 替换文献[1]中提到的词性嵌入会带来稳定的模型性能提升。

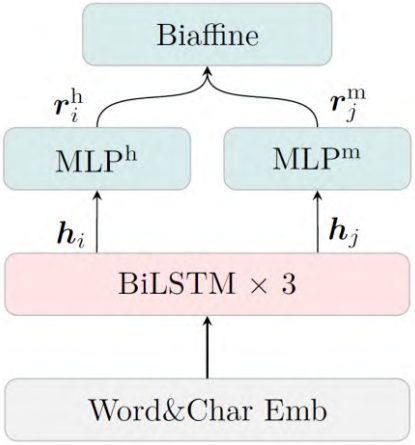


图 1 双仿射句法分析器模型框架
Fig.1 Framework of Biaffine dependency parser

BiLSTM 编码模块：把输入映射为上下文信息可感知的表示。该模块把句子的输入$X=[x_0,x_1,\cdots,x_n]$输入到 3 层双向长短期记忆网络得到上下文信息可感知的表示向量$H=[h_0,h_1,\cdots,h_n]$。每一个隐向量h_i都是由编码模块顶层的前向输出向量$\overrightarrow{h_i}$和后向输出向量$\overleftarrow{h_i}$拼接得到。

$$h_i=\overrightarrow{h_i}\oplus\overleftarrow{h_i}\ .$$

MLP 模块：对 BiLSTM 的输出进行降维，去除与句法无关的信息。该模块以隐向量作为输入使用两个不同的 MLP 模块得到句法向量表示。

$$[r_i^h;r_i^m]=f_{MLP}(h_i)\ ,$$

其中，r_i^h和r_i^m分别是词w_i作为依存核心词和依存修饰词的向量表示，f_{MLP}是以$ReLU$为激活函数具有一层隐藏层的多层感知机。

双仿射打分模块：使用双仿射函数对有所依存弧进行打分。

$$s(\boldsymbol{x}_i,\boldsymbol{x}_j)=\begin{bmatrix}r_j^m\\1\end{bmatrix}^T W^b r_i^h\ ,$$

其中，$\mathrm{s}(\boldsymbol{x}_i,\boldsymbol{x}_j)$是依存弧$i\rightarrow j$的分值，$W^b$为双仿射层的参数。本研究参照 DOZAT 等[1]的研究，采用额外的多层感知机对依存关系进行打分，此处不再赘述。

训练损失：采用词级别的非结构化损失函数，即最大化每个词对应的正确核心词的局部概率。假定词w_j的核心词是w_i，其对应的交叉熵损失为：

$$\ell(\boldsymbol{x}_i,\boldsymbol{x}_j)=-lg\frac{\mathrm{e}^{s(x_i,x_j)}}{\sum_{0\le k\le n,k\ne j}\mathrm{e}^{s(x_i,x_j)}}\ ,$$

其中，n为句子长度。最终所有词的损失累加起来构成句子的损失，不考虑任何树结构约束。

解码策略：分两个阶段按序进行依存骨架树的解析和依存关系的解析。首先是用最大生成树算法[12]得到分值最高的一颗依存骨架树$\boldsymbol{y}^*$。

$$\boldsymbol{y}^*=\arg\max_{\boldsymbol{x}}\mathrm{s}\left(\boldsymbol{x}_i,\boldsymbol{x}_j\right)\equiv\sum_{(x_i,x_j)\in\boldsymbol{x}}\mathrm{s}(\boldsymbol{x}_i,\boldsymbol{x}_j)\ ,$$

然后采用贪婪分类的策略为树中每条依存弧分配一个依存关系，得到一个最优的依存标签树。

1.2 多模型决策协同训练策略

把经典的 self-training 应用到跨领域句法分析上包括以下几步:

1）在源领域训练集上训练一个初始模型。
2）使用初始模型预测目标领域的无标注数据。
3）把置信度高的自动标注句子加入到源领域训练集构成一个新的训练集。
4）在新训练集上重新训练一个自学习模型。
5）重复上述过程直到新模型的性能不再提升。

自动标注数据的质量对模型的性能至关重要，但是目前仍缺乏将其应用于跨领域依存句法分析任务的系统性深入研究。为了解决这个问题，本研究使用 3 种不同的 tri-training 算法进行自动标注数据的筛选。即两种横向、一种纵向的 tri-training 算法，相应的伪代码见附录（http://jxmu.edu.cn/upload/html/202204xx.html）。

对于算法 1（v-tri）：首先在源领域训练集上使用不同的随机种子训练 3 个初始模型，其中一个主模型M_0两个辅助模型M_1, M_2（主模型和辅助模型按照时间顺序进行划分，即第一个训练的模型为主模型，其它的为辅助模型）。如果这 3 个模型在无标注数据上的预测结果一致，则把预测的自动标注数据加入到训练集中。接着在新训练集上重新训练主模型，本研究把主模型的性能在五轮内是否提升作为算法的停止条件。每一轮训练期间都进行停止条件判断，如果不满足停止条件则训练两个辅助模型；否则，将满足停止条件的峰值性能最高的主模型作为最终的模型。先进行模型性能的比较，可以省去不必要的辅助模型训练。最后使用主模型M_0在目标领域上进行性能测试。

算法 2（c-tri）采用经典的 tri-training[4]算法思想：与算法 1 类似，同样训练三个初始模型，对于每一个模型，把另外两个模型预测一致的数据加入到本模型的训练集中。

算法 3（i-tri）与 Li 等[9]类似，把当前轮次之前两轮的模型预测一致的自动标注数据加入到当前轮次的训练集中。不同于 Li 等[9]使用 BERT 作为编码层，本研究使用 BiLSTM 作为编码层并使用 BERT 进行微调（详情见 1.3 节）。

1.3 BERT 微调

以 BERT[10]为代表的大规模语料上的预训练语言模型往往采用 Transformer[13]架构，其优势在于可以使用近乎无限量的无标注数据且学习到的表征可以在多个任务中进行快速迁移。将预训练模型接入一个任务特定的模型，然后在新的数据集上进行微调是一种比较典型的预训练模型应用方式。例如 Howard 等[14]提出通用语言模型微调，在六种文本分类任务上达到了最佳; Sun 等[15]使用目标领域的数据和多任务学习在文本分类任务上进行微调；Xu 等[16]提出了评论阅读理解任务并使用后-训练策略进行微调。

Li 等[17]的工作表明使用目标领域无标注文本对通用预训练模型进行多轮微调，即继续使用掩码语言模型损失函数进行训练，可以在依存句法分析领域移植任务上取得显著的性能提升。鉴于此，本研究使用 Google 开源的 bert-base-chinese（https://github.com/google-research/bert）模型作为原始模型，在所有领域的 train/unlabeled 原始文本上仅使用掩码语言模型损失进行训练。与 Li 等[18]的工作保持一致，使用 BERT 的输出$b(w_i)$替换词向量$e(w_i)$作为模型的输入特征进行模型训练，其中$b(w_i)$是把 BERT 最后四层的输出进行加权求和然后线性映射为的 100 维字表示向量。在训练过程中冻结 BERT 的参数，不进行更新。

2 实验

2.1 实验设置

实验数据：本研究在 NLPCC-2019 评测任务发布的中文依存句法分析领域移植数据集上进行实验。此数据集包含 4 个领域，共计 4×10^4 句，四个领域分别是一个源领域和三个目标领域，其中源领域是来自新闻领域的平衡语料库（balanced corpus, BC），3 个目标领域语料库分别是来自淘宝标题的产品博客（product blog, PB）、来自淘宝的产品评论（product comment, PC）、来自网络小说的“诛仙”（ZhuXian, ZX）。本研究直接采用官方划分好的训练集、开发集和测试集，具体的各个领域以句子为单位的统计结果如表 1 所示。

表1 NLPCC-2019评测数据的统计信息

Tab.1 Data statistics of NLPCC-2019 shared task

	训练集	验证集	测试集	无标注数据集
BC	16 339	998	1 995	0
PB	5 129	1 300	2 600	291 481
PC	6 885	1 300	2 600	349 922
ZX	1 645	500	1 100	33 792

评价标准：使用无标签/有标签依存分值（unlabeled/labeled attachment score, UAS/LAS）作为评价指标，其中 UAS 为核心词正确的词数占总词数的比值，即 UAS 只考虑依存骨架，不考虑依存关系；LAS 为核心节点正确且对应依存关系类型也正确的词占总词数的比值，即 LAS 同时考虑依存骨架和依存关系。评价时，采用官方的评价脚本，直接忽略没有正确核心词标注的词。

预训练词向量：通过 Google 开源的 word2vec（https://code.google.com/archive/p/word2vec/）在 Chinese Gigaword V3（约 1.1×10^6 句、自动分词）和所有领域 train/unlabeled 原始文本上训练 10 次迭代得到。

基线方法（baseline）：源领域训练集上得到的模型直接在目标领域进行测试。

超参设置：除了数据预处理和输入层的参数外，其它的参数和双仿射句法分析器[1]的参数设置与文献[1]保持一致。在预处理阶段根据句子长度对原始文本进行聚类，然后把聚类后的文本划分为若干个批次，每个批次不超过 5 000 个词。输入层的字符集双向长短期记忆网络的输入维度是 50，输出维度是 100。

2.2 主实验

最终实验结果如表 2 所示。其中 Li 等[9]使用 BERT 作为模型的编码层，其它方法均使用 BERT 表示作为额外的输入特征。实验结果表明在基线方法的基础上无论是结合 FBERT（fine-tuned BERT）还是 tri-training 训练策略，模型的性能都有大幅度的提升。且基线方法的性能越低，结合 FBERT 或 tri-training 训练策略后提升越大。例如，加入 v-tri-training（v-tri）后，模型的 LAS 值在 PB、PC、ZX 领域上分别同比增长了 13.85%（v-tri vs. baseline）、34.78%和 14.53%；加了 FBERT（baseline-FB）后，模型的 LAS 值在 PB、PC、ZX 领域上分别同比增长了 24.51%（baseline-FB vs. baseline）、66.63%和 21.97%。即使加了 FBERT 和 v-tri-training 后准确率已经有了显著的提升，把两者结合依然可以进一步提升模型性能。这表明两种方法分别是从不同的角度对模型有帮助，在下一节我们会进行更深入的探究。

表2 测试集上的最终结果

Tab.2 Final results on the test set

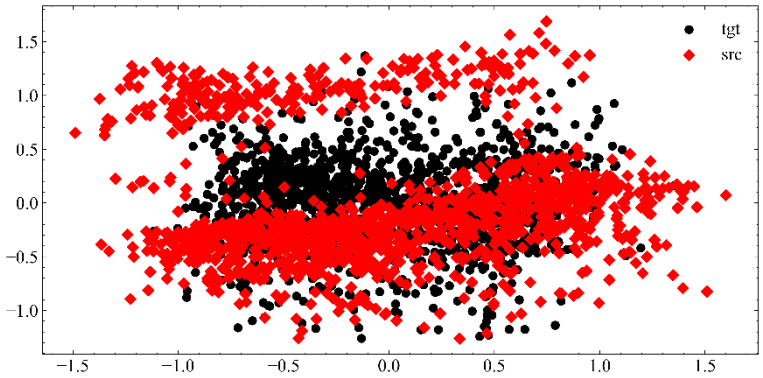
模型	PB		PC		ZX		AVG	
	UAS/%	LAS/%	UAS/%	LAS/%	UAS/%	LAS/%	UAS/%	LAS/%
Yu 等[19]	71.48	65.43	49.82	36.86	73.90	66.54	65.07	56.28
baseline	68.20	61.52	41.82	28.32	69.81	62.21	59.94	50.68
v-tri	75.96	70.04	50.94	38.17	78.11	71.25	68.34	59.82
c-tri	75.82	70.21	50.28	37.74	78.36	71.34	68.15	59.76
i-tri	75.29	69.87	49.72	37.32	78.21	71.19	67.74	59.46
Li 等[9]	81.61	76.77	60.50	49.49	79.74	74.32	73.95	66.86
baseline-FB	81.47	76.60	59.58	47.19	81.95	75.88	74.33	66.56
v-tri-FB	83.68	79.33	64.90	54.37	83.82	78.14	77.47	70.61
c-tri-FB	83.70	79.38	65.62	55.17	84.30	78.32	77.87	70.96
i-tri-FB	83.72	79.34	64.65	53.71	84.57	78.66	77.65	70.57

注：FB 表示使用 FBERT 的输出作为句法模型的输入特征。v-tri、c-tri、i-tri 代表前文提到的三个算法，即 vote based tri-training、classic tri-training 和 iterative tri-training 算法。

2.3 分析实验

2.3.1 为什么跨领域模型性能会下降，领域的差异在哪里？

从机器学习建模角度来说，跨领域性能大幅度下降的直接原因是测试集和训练集中的潜在数据分布不同。图 2 给出了把源领域和目标领域训练集 ZX 上得到的模型的 BiLSTM 模块的输出进行主成分分析（principal component analysis, PCA）降维可视化（具体地：在目标领域开发集上采样 100 句输入到源领域和目标领域训练集上得到的模型，然后把模型输出的 BiLSTM 特征进行 PCA 降维可视化），可以发现两者在高维空间上的特征分布确实存在着不同。



图中黑色表示目标数据集 ZX，红色表示源数据集 BC，下同。

图 2 源领域数据和目标领域数据上得到的模型的特征分布

Fig.2 The feature distributions of different models which are trained with the source domain or the target domain data

从文学语法层次结构角度来说，句子的分布可能会有词语、句法和语义层面的变化。首先，不同领域使用的词语分布是不同的，有些词语是某些领域所特有的，有些词语在不同的领域的出现频率是不同的。图 3 是不同领域词频最高的 10 个词，不同领域中词语分布不尽相同。第二，从句法层面，不同领域的语法结构的分布也会发生变化。例如，祈使句、疑问句、陈述句等在不同领域的文本中相对频率发生变化。句法结构的变化会直接导致语序发生变化，无论是传统基于离散特征的模型，还是深度学习模型，都会受到影响。第三，从语义层面，不同领域的文本涉及的话题、概念的分布也可能发生变化。

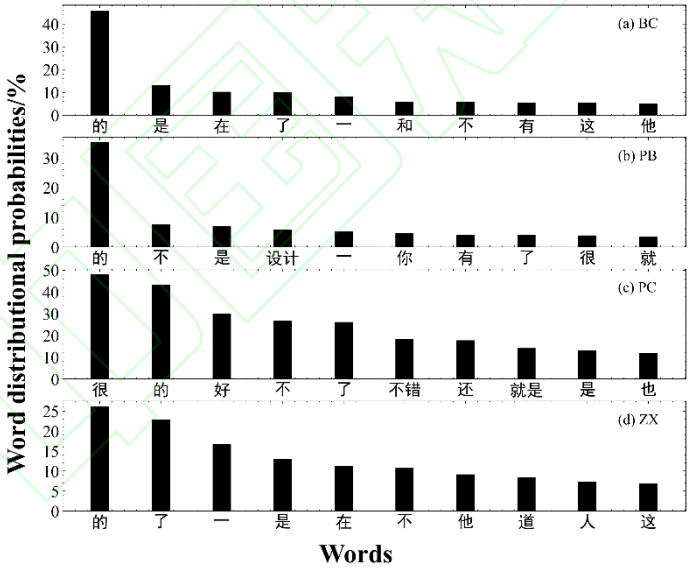


图 3 不同领域中出现频率最高的 10 个词

Fig.3 The top 10 words with the highest frequency on different domains.

2.3.2 为什么 tri-training 有用?

前面的实验表明 tri-training 可以大幅度提升模型性能，在此小节探究为什么 tri-training 有效。语法是对蕴含在语言中的客观规律的描述，反之，当给定大量的语言实例时，有可能总结出语法规律。而 tri-training 通过对目标领域的大量无标注数据进行迭代，目标领域的部分语法规则有可能被逐步保留下来。对经过 tri-training 训练策略后的 BiLSTM 模块的输出进行 PCA 降维，图 4 的红色部分给出了 v-tri-training 得到的模型的特征分布情况。图 4 的黑色部分与图 2 的黑色部分一样，都是目标领域训练集上得到的模型的特征分布情况。可以看出，经过 v-tri-training 之后学习到的模型与目标领域训练数据上得到的模型的编码空间更加接近了，从而使得模型的跨领域性能有所提升，这也在一定程度

上印证了上面的猜想。

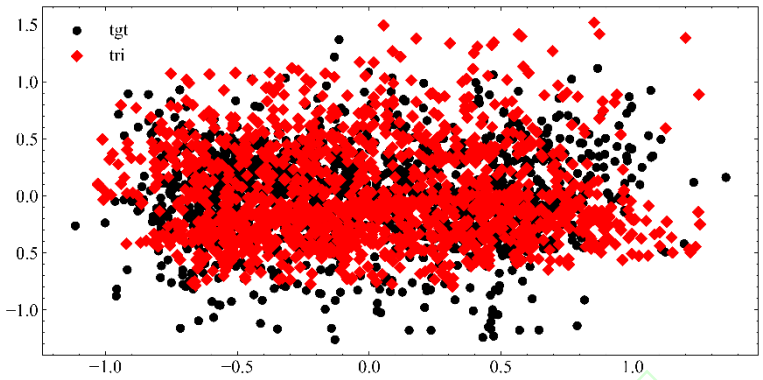


图 4 Tri-training 对模型特征分布的影响
Fig.4 The effect of tri-training on feature distributions

2.3.3 为什么 fine-tuned BERT 有用?

预训练语言模型是在海量的无标注文本上进行训练的，这些文本往往已经涵盖了所有源领域和目标领域，因此预训练模型对跨领域句法分析非常有用[20]。同样的，本研究将使用 FBERT 得到的源领域模型和目标领域上的模型的特征进行对比，可以看到使用 FBERT 后的模型的编码空间变得更广，从而使得以 FBERT 表示作为输入的源领域模型与目标领域模型的编码空间有更多的重合，进而提高了模型在目标领域上的泛化能力。

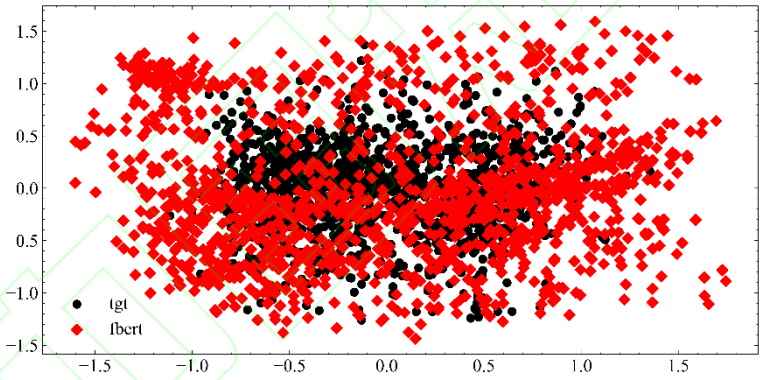


图 5 FBERT 对模型特征分布的影响
Fig.5 The effect of FBERT representations on feature distributions

2.3.4 不同 tri-training 算法的训练趋势

图 6 是不同 tri-training 算法在开发集上的训练趋势。可以看出，不同的 tri-training 训练算法皆是增量式的利用目标领域的无标注数据。模型的性能也有着相似的提升趋势，但模型性能的提升主要在前两轮。自学习方法有效的条件是使用大量无标注数据带来的正面影响超过自动标注数据引入的噪音所产生的负面影响。本文猜测在训练的后期，无标注数据带来的负面影响超过了正面影响以至于后期模型的性能不提升甚至有所下降。

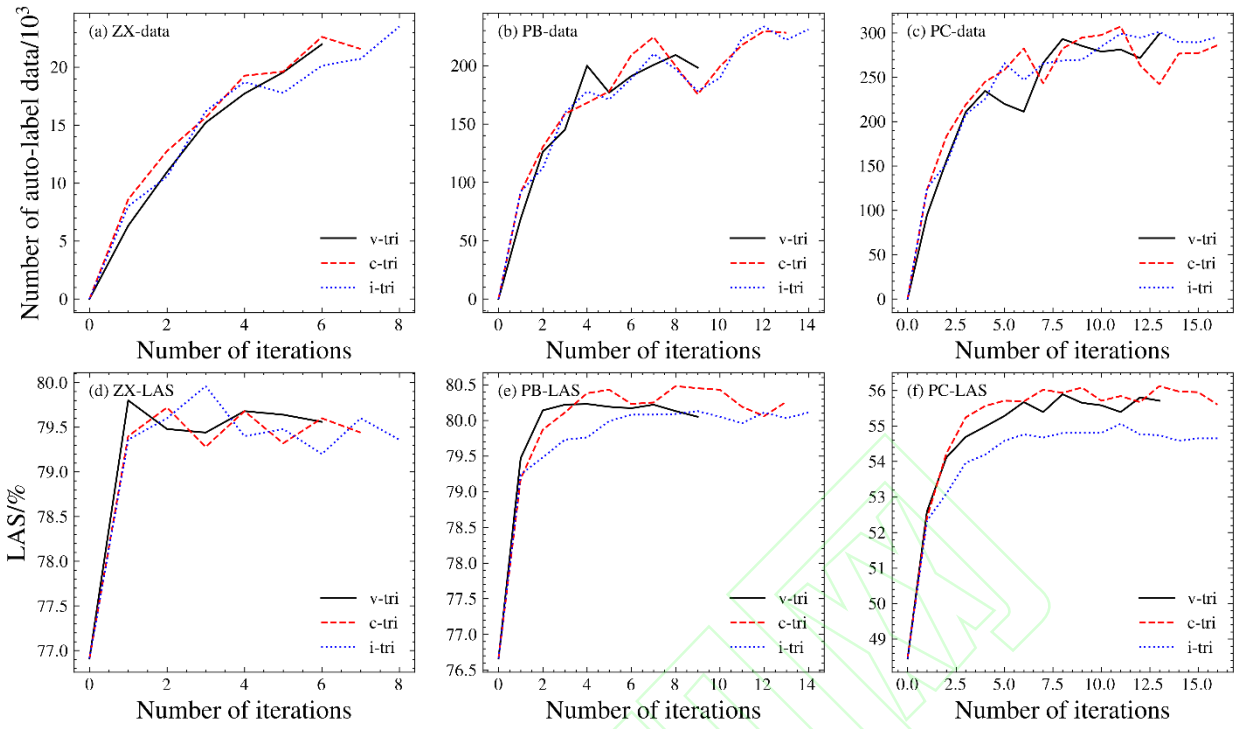


图 6 不同 tri-training 训练算法自动标注数据数量和模型性能的变化曲线
Fig.6 The change curves of the amount of auto-labeled data and the model performance on different tri-training algorithms

2.3.5 错误分析

图 7 是不同 tri-training 算法在 ZX 开发集上对不同依存标签的预测准确率。得益于 tri-training 和 FBERT，三种算法几乎在所有的依存关系上都有提升，但是各个方法又各有所长。比如说在“adv”依存关系上，vote based tri-training 算法取得了最高的准确率。Classic tri-training 在“adjct”依存关系上取得了最高的准确率。而“subj”依存上的最高准确率则是由 iterative tri-training 算法取得。

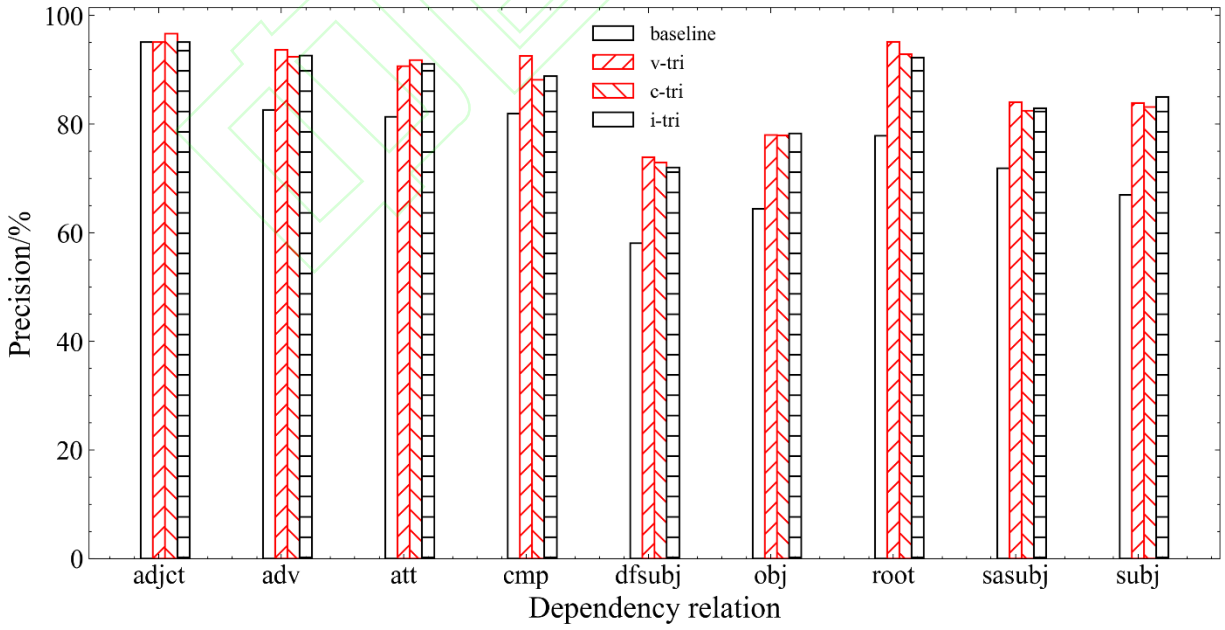


图 7 不同 tri-training 算法在 ZX 开发集上对依存关系的预测准确率
Fig.7 The precision of dependency relations by different tri-training algorithms on the ZX dep dataset

3　结 论

自动标注数据的质量是自学习方法成功与否的关键，本文使用 3 种简单的多模型决策系统训练算法（tri-training）来保证自动标注数据的质量进而提升跨领域依存句法分析性能。在 NLPCC-2019 评测任务发布的数据集上取得了显著的提升并成为了当前最佳，同时还深入探究了导致跨领域模型性能下降的原因以及各个模块的作用。结果表明，经过 tri-training 之后学习到的模型与目标领域训练数据上得到的模型的编码空间更加接近了，从而使得模型的跨领域性能有所提升；使用 FBERT 后的模型的编码空间变得更广，从而使得以 FBERT 表示作为输入的源领域模型与目标领域模型的编码空间有更多的重合，进而提高了模型在目标领域上的泛化能力。

附录

算法 1 vote based tri-training
输入：源领域标注数据 S，目标领域无标注数据 T。
输出：性能最好的模型 M_0^t
停止条件：stop ← M_0^t.LAS 五轮内不提升

```
1：for i ∈ 0,1,2:
2：    M_i^0← model_training(S) with different random seed
3：for t ∈ 1,2,3…:
4：    for x ∈ T:
5：        if M_0^(t -1) (x)==M_1^(t -1) (x)==M_2^(t -1) (x):
6：            S^t←S∪\{(x,M_1^(t -1) (x))\}
7：    M_0^t← model_training(S^t)
8：    if not stop:
9：        for i ∈ 1,2:
10:        M_i^t ← model_training(S^t)
11:   else:
12:        return M_0^(t -5)
```

算法 2 classic tri-training
输入：源领域标注数据 S，目标领域无标注数据 T。
输出：性能最好的模型 M_0^t
停止条件：stop ← M_0^t.LAS 五轮内不提升

```
1：for i ∈ 0,1,2:
2：    M_i^0← model_training(S)with different random seed
3：for t ∈ 1,2,3…:
4：    for i ∈ 0,1,2:
4：        for x ∈ T:
5：            if M_((i+1)%3)^(t -1) (x)==M_((i+2)%3)^(t -1) (x):
6：                S_i^t←S∪\{(x,M_((i+1)%3)^(t -1) (x))\}
7：    M_0^t← model_training(S_0^t)
8：    if not stop:
9：        for i ∈ 1,2:
10:        M_i^t ← model_training(S_i^t)
11:   else:
12:     return M_0^(t -5)
```

算法 3 iterative tri-training
输入：源领域标注数据 S，目标领域无标注数据 T。
输出：性能最好的模型 M^t
停止条件：stop ← M^t.LAS 五轮内不提升

```
1：for t ∈ 0,1:
2：    M^i ← model_training(S)with different random seed
3：for t ∈ 2,3,4…:
4：    for x ∈ T:
5：        if M^(t-1) (x)==M^(t -2) (x):
6：            S^t←S∪\{(x,M^(t-1) (x))\}
7：    M^t ← model_training(S^t)
8：    if not stop:
9：        continue
11:   else:
12:     return M^(t -5)
```

参考文献

[1] DOZAT T, MANNING C D. Deep biaffine attention for neural dependency parsing[EB/OL]. [2021-10-01]. https://arxiv.org/abs/1611.01734.
[2] ZHANG Y, LI Z H, ZHANG M. Efficient second-order TreeCRF for neural dependency parsing[C]//Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2020: 3295-3305.
[3] STEEDMAN M, OSBORNE M, SARKAR A, et al. Bootstrapping statistical parsers from small datasets[C]//Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2003: 331-338.
[4] ZHOU Z H, LI M. Tri-training: exploiting unlabeled data using three classifiers[J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(11): 1529-1541.

[5] MCCLOSKY D, CHARNIAK E, JOHNSON M. Reranking and self-training for parser adaptation[C]//Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2006: 337-344.
[6] KAWAHARA D, UCHIMOTO K. Learning reliability of parses for domain adaptation of dependency parsing[C]//International Joint Conference on Natural Language Processing. Stroudsburg: ACL, 2008: 709-714.
[7] YU J T, ELKAREF M, BOHNET B. Domain adaptation for dependency parsing via self-training[C]//International Conference on Parsing Technologies. Stroudsburg: ACL, 2015: 1-10.
[8] SOGAARD S, RISHOJ C. Semi-supervised dependency parsing using generalized tri-training[C]//International Conference on Computational Linguistics. Stroudsburg: ACL, 2010: 1065-1073.
[9] LI Z C, ZHOU J R, ZHAO H, et al. Cross-domain transfer learning for dependency parsing[C]//Natural Language Processing and Chinese Computing. Cham: Springer, 2019: 835-844.
[10] DEVLIN J, CHANG M W, LEE K, et al. BERT: pre-training of deep bidirectional transformers for language understanding[C]// Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg: ACL, 2019: 4171-4186.
[11] LAMPLE G, BALLESTEROS M, SUBRAMANIAN S, et al. Neural architectures for named entity recognition[C]// Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg: ACL, 2016: 260-270.
[12] SHIMIZU N. Maximum spanning tree algorithm for non-projective labeled dependency parsing[C]//Conference on Computational Natural Language Learning. Stroudsburg: ACL, 2006: 236-240.
[13] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Advances in Neural Information Processing Systems. Long Beach: NeurIPS, 2017: 5998-6008.
[14] HOWARD J, RUDER S. Universal language model fine-tuning for text classification[C]// Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2018: 328-339.
[15] SUN C, QIU X P, XU Y G, et al. How to fine-tune BERT for text classification[C]//China Nation Conference on Chinese Computational Linguistics. Cham: Springer, 2019:194-206.
[16] XU H, LIU B, SHU L, et al. BERT post-training for review reading comprehension and aspect-based sentiment analysis[C]// Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg: ACL, 2019: 2324-2335.
[17] LI Y, LI Z H, ZHANG M. Semi-supervised domain adaptation for dependency parsing via improved contextualized word representations[C]//International Conference on Computational Linguistics. Stroudsburg: ACL, 2020: 3806-3817.
[18] LI Y, LI Z H, ZHANG M, et al. Self-attentive biaffine dependency parsing[C]//International Joint Conference on Artificial Intelligence. Macao: IJCAI, 2019: 5067-5073.
[19] YU N, LIU Z L, ZHEN R, et al. Domain information enhanced dependency parser[C]// Natural Language Processing and Chinese Computing. Cham: Springer, 2019: 801-910.
[20] 屠可伟， 李俊． 句法分析前沿动态综述[J]． 中文信息学报， 2020， 34（7）： 30-41．