doi:10.3969/j.issn.1001-4616.2025.03.016

基于类型驱动及模型融合的中文语法纠错研究

许婉秋[1,2],曲维光[1,3],魏庭新[4],谷静平[3],顾彦慧[1],周俊生[1]

(1.南京师范大学计算机与电子信息学院/人工智能学院,江苏 南京 210023)
(2.江苏海事职业技术学院信息工程学院,江苏 南京 211170)
(3.南京师范大学中北学院,江苏 丹阳 212334)
(4.南京师范大学国际文化教育学院,江苏 南京 210097)

[**摘要**] 中文语法纠错旨在通过模型自动识别并修正中文文本中的语法错误,从而提升文本的准确性和可读性. 然而,现有的中文语法纠错模型在纠错过程中常面临暴露偏差问题,并且对大模型的应用仍显不足,导致纠错效果欠佳. 为此,本文提出了一种基于类型驱动的中文语法纠错模型 CTDGC(Chinese Types Driven Grammatical Correction). 该模型通过深入探讨中文四种主要语法错误(冗余、缺失、错词、乱序)之间的依赖关系,设计了两阶段训练策略,有效缓解了训练与预测的不匹配问题,在 CGED2020 数据集上单模型 $F_{0.5}$ 达到 34.18%,优于以往的方法. 此外,本文还提出了一种基于 ChatGLM 的中文语法纠错模型 CorGLM(Chinese Grammatical Correction Model based on ChatGLM),并对 Baichuan 大模型设计了特定的 Prompt. 通过与 CTDGC 等模型的融合,$F_{0.5}$ 显著提升至 40.33%,验证了本文方法的有效性和优越性.

[**关键词**] 中文语法纠错,类型依赖关系,两阶段训练,大规模语言模型,模型融合
[**中图分类号**]TP391　[**文献标志码**]A　[**文章编号**]1001-4616(2025)03-0139-10

Research on Chinese Grammar Correction Based on Type-Driven and Model Fusion

Xu Wanqiu[1,2], Qu Weiguang[1,3], Wei Tingxin[4], Gu Jingping[3], Gu Yanhui[1], Zhou Junsheng[1]

(1.School of Computer and Electronic Information/Artificial Intelligence, Nanjing Normal University, Nanjing 210023, China)
(2.School of Information Technology, Jiangsu Maritime Institute, Nanjing 211170, China)
(3.Nanjing Normal University Zhongbei College, Danyang 212334, China)
(4.International College for Chinese Studies, Nanjing Normal University, Nanjing 210097, China)

Abstract: Chinese grammar error correction aims to automatically identify and rectify grammatical errors in Chinese text using models, thereby improving the accuracy and readability of the text. However, existing Chinese grammar correction models often face exposure bias issues during correction, and their application of large models remains inadequate, resulting in suboptimal correction performance. To address this, this paper proposes a Chinese Types Driven Grammatical Correction (CTDGC) model. By thoroughly exploring the dependency relationships among four major types of Chinese grammatical errors (redundancy, omission, incorrect word usage, and word order errors), the model employs a two-stage training strategy that effectively alleviates the mismatch between training and prediction. The CTDGC model achieves an $F_{0.5}$ score of 34.18% on the CGED2020 dataset, outperforming previous methods. Additionally, this paper introduces a Chinese grammar correction model based on ChatGLM (CorGLM) and designs specific prompts for the Baichuan large model. Through integration with models like CTDGC, the $F_{0.5}$ score significantly improves to 40.33%, demonstrating the effectiveness and superiority of the proposed approach.

Key words: Chinese grammar error correction, type dependency relationship, two-stage training, large language model, model fusion

中文语法纠错(Chinese grammatical error diagnosis, CGED)的目标是通过自动化工具识别并修正中文文本中的语法错误,从而提升文本的准确性和可读性. 准确而有效地检测和纠正这些错误对提升写作质量、加

收稿日期:2024-06-26.
基金项目:国家社会科学基金重大项目(21&ZD288)、江苏省教育厅哲社项目(2024SJYB1630).
通讯作者:曲维光,博士,教授,研究方向:自然语言处理,计算语言学、语言工程、人工智能. E-mail:wgqu_nj@163.com

深语言理解及确保有效沟通至关重要. Rao 等[1]将语法错误归为四种类型:冗余(Redundant Words,R)、缺失(Missing Words,M)、错词(Word Selection,S)和乱序(Word Ordering Errors,W):

例 1:那么他们的年收入大概多少的呢?

例 2:每年生产近 300 万吨油籽.

例 3:还有其他人受被害.

例 4:我彻底看到了对方的无情,当时死都想了.

例 5:他懂人生道理,他总是那样和渴可近.

其中,例 1 是一个包含冗余错误的句子,去除“的”得到正确句子“那么他们的年收入大概多少呢?”;例 2 中缺失“菜”字,应改正为“每年生产近 300 万吨油菜籽. ”;例 3 中“被”字为“伤”的错字,应改正为“还有其他人受伤害. ”;例 4 中存在乱序错误,应改正为“我彻底看到了对方的无情,当时都想死了. ”此外,很多时候一句话都是包含多个错误的,如例 5 所示,包含缺失和错字错误,应改正为“他懂得人生的道理,他总是那样和蔼可亲. ”

尽管 CGED 在方法上已经取得了一些进展,但仍存在一些亟待解决的挑战. 首先,现有的方法在处理中文语法错误时普遍存在暴露偏差的情况,这种偏差主要源于模型在训练时采用了一次迭代的方式,但预测时却需要多轮纠错,这种训练与预测之间的不匹配影响了模型的性能. 受 Lai 等[2]探究英文三种错误类型的关系来提升英文语法纠错效果的启发,本文着眼于针对中文语法错误的特点,探究中文语法错误中四种类型的相互依赖关系. 这些错误类型之间可能存在紧密的联系,有利于缓解暴露偏差的问题.

随着大模型在自然语言处理领域的广泛应用,其在多个任务上所取得的显著性能提升已成为学术界和工业界的研究热点. 然而,在 CGED 领域,大模型的应用相对较为有限,主要原因是存在过纠正问题,这一限制可能阻碍了其在该任务中充分发挥潜力的可能性. 因此,为了解决这一问题,迫切需要更为深入的研究,以使大模型更好地适应 CGED 任务的特殊性.

本文的贡献如下:

(1)提出了基于类型驱动的中文语法纠错模型 CTDGC(Chinese Types Driven Grammatical Correction). 该模型通过分析中文语法错误类型之间的依赖关系,设计了两阶段训练策略,有效缓解了训练与预测过程中的不匹配问题,显著提升了中文语法纠错的性能.

(2)提出了基于 ChatGLM 的中文语法纠错模型 CorGLM(Chinese Grammatical Correction Model based on ChatGLM). 通过微调结合拼写纠错特征的 ChatGLM 模型,为 Baichuan 大模型设计了特定的 Prompt,并与 CTDGC 等模型进行融合,实现了性能的进一步提升,验证了大模型在中文语法纠错任务中的潜力和实际应用价值.

本文方法的逻辑关系框架如图 1 所示,我们首先探究了中文语法错误类型的关系,将其作为理论基础设计了 CTDGC 模型、CorGLM 模型和大模型 Prompt,最后进行模型融合实验.

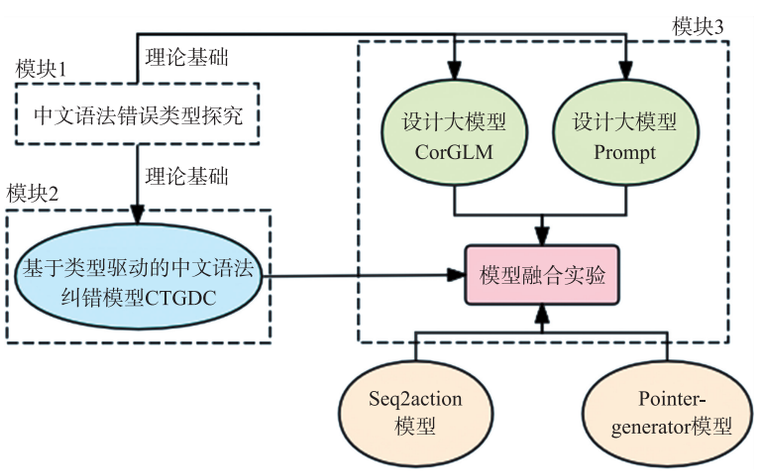


图 1　逻辑关系框架图

Fig. 1　Logical relationship framework diagram

1 相关工作

CGED 是自然语言处理领域中的关键下游任务之一. 早期多为规则和统计学习的方法,由于 CGED 任务比较复杂,当时只能做到对中文语法错误进行检测,直到深度学习的引入,才使得中文语法错误的纠正成为可能,纠错任务主要分为基于序列生成的方法和基于序列标注的方法.

序列生成方法直接从原始句子生成正确的句子,类似机器翻译任务. Fu 等[3]将 CGED 任务视为翻译任务,让神经网络学习错误句子与正确句子之间的关系,并将错误的句子翻译为正确的句子. Ren 等[4]提出多层卷积和注意力机制的全卷积 encoder-decoder 模型,它可以有效捕获局部上下文信息,并且根据源词之间的相关性为其分配权重. Wang 等[5]将预训练模型采用两种方法与 encoder-decoder 模型相结合,第一种为使用 BERT 模型的参数初始化 Transformer 的 encoder,第二种将句子输入到预训练模型中,将最后一层获得的表示通过注意力机制与 encoder-decoder 融合. Li 等[6]提出 TtT 模型,引入 BERT 来学习上下文表示,然后堆叠 CRF 层建模相邻 token 之间的相关性. Wu 等[7]提出 SG-GEC 模型,将 CGED 任务分为拼写错误纠正和语法错误纠正两个步骤,并设计词性特征和语义类特征来增强神经网络模型.

序列生成方法虽然能解决 CGED 问题,但其生成速度较慢且可解释性较差. 为此,序列标注方法应运而生,旨在克服这些不足. Chen 等[8]给数据集打上错误类型和词性的标签,然后基于条件随机场(CRF)构建 CGED 系统,在 CGED2015 任务中准确率最高. Omelianchuk 等[9]提出了 GECToR 模型,并为语法操作定义了 5 000 个标签,包括基本标签,如 KEEP,DELETE,APPEND_X 和 REPLACE_X,以及 29 个英语语法变化的新标签. Liang 等[10]首次将 GECToR 模型应用于 CGED,他们删除了英语的 29 个语法变化标签,并为常见的中文字符构造了 8 772 个基本标签. Li 等[11]提出 Seq2action 模型,结合了序列生成和序列标注模型. Seq2action 模型首先为每个 token 打上动作标签,其中包括 SKIP、COPY 和 GENerate 三个选项,当动作为 GENerate 时做生成任务.

2 不同类型错误的相互依赖关系探究

中文语法错误分为冗余、缺失、错词和乱序四种,我们在 GECToRm 模型(http://github.com/Anil999999/MS_Seq2Edit)的基础上深入探讨这四种语法错误之间的相互依赖关系,并设计了相应的实验方法进行研究.

2.1 GECToRm 模型

GECToRm 模型是在 GECToR 模型的基础上引入移动(MOVE)操作来解决乱序错误,标签体系如表 1 所示. KEEP 标签表示保持当前字符不变,DELETE 标签表示删除当前位置的字符,APPEN_N 标签表示在当前位置的后面插入 N 字符,REPLACE_N 标签表示将当前字符替换为 N 字符,MOVE_N 标签表示将当前字符向右移动 N 个位置,只有存在局部路径和局部字符编辑标签向左移动<负数>和向右移动<正数>的累积和为 0 的情况下才接受该操作.

表 1 模型标签体系

Table 1 Model label system

标签	描述	备注
KEEP	保持	保持当前字符不变
DELETE	删除(冗余)	删除当前位置的字符
APPEND_N	插入(缺失)	在当前位置的后面插入 N 字符
REPLACE_N	替换(错词)	将当前字符替换为 N 字符
MOVE_N	移动(乱序)	将当前字符向右移动 N 个位置

2.2 依赖关系探究方法

本文将一条数据分为五个部分,如图 2 所示,分别是:正确部分、缺失错误、冗余错误、错词错误和乱序错误. 假设错误为 X,X 为四种错误中的一种,为了深入了解 X 错误对其他错误的影响,我们设计了一系列步骤. 首先将语料库中含有 X 错误的数据集筛选出来,并赋予其一个新的标识——D(X). 接着对 D(X)中的错误进行纠正,从而得到了经过修正的数据集 D(X✓). 这一操作的目的在于建立一个对比基准,使我们能够准确评估 X 错误对其他错误类型的影响. 随后,将这两个数据集分别输入 GECToRm 模型进行纠错,以观察其纠正效果. 最后,我们通过对比的纠正效果得出先纠正 X 错误对其他错误类型的作用. 假设 X 为缺失错误,我们首先从语料库中筛选出含有缺失错误的数据集,并将其命名为 D(D). 接下来,我们对 D(D)中的冗余错误进行修正,即将多余的字词删除,得到新的数据集 D(D✓). 通过对比 D(D)和 D(D✓)的性能,旨在探讨在先纠正冗余错误的前提下是否对其他错误类型的纠正产生了提升效果. 这一研究设计旨在加深我们对错误

纠正过程中不同类型之间相互影响的理解.

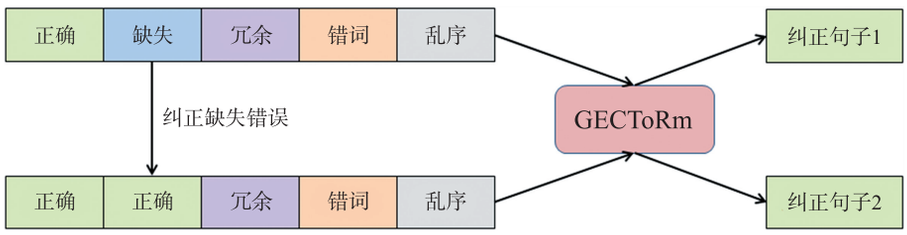


图 2　探究不同类型错误相互依赖性的方法

Fig. 2　Methods for exploring the interdependence of different types of errors

2.3　依赖关系实验结果分析

本文使用在 Lang8 数据集上经过两次训练的 GECToRm 模型进行测评,每组测试数据集均从 CGED 历年数据中随机抽取了 2 000 条句子,实验结果如表 2 所示. 第一组对比实验显示,先纠正缺失错误对其他三种错误的纠正效果均有提升,尤其是对乱序错误的纠正效果提升了 15.41%. 第二组实验先纠正冗余错误,第三组实验先纠正错字错误,第四组实验则先纠正乱序错误. 通过这些对比实验发现,先纠正一种错误对其他三种错误的纠正效果均有所提升,尤其是先纠正错字错误对其他三种错误的促进作用最大,分别对缺失、冗余和乱序错误的纠正效果提升了 12.21%、13.27%和 15.19%.

表 2　四种错误依赖关系实验结果

Table 2　Experimental results of four error dependence relationships

错误类型	标签	D(X)			D(X✓)		
		P%	R%	F1%	P%	R%	F1%
M	R	22.55	22.13	22.34	28.04	26.72	27.36(+5.02)
	S	29.55	7.30	11.71	33.10	8.30	13.27(+1.56)
	W	37.93	2.53	4.74	58.24	12.18	20.15(+15.41)
R	M	32.38	7.99	12.82	36.13	8.69	14.02(+1.2)
	S	31.18	7.45	12.02	32.60	7.66	12.40(+0.38)
	W	30.00	1.35	2.58	42.31	2.47	4.66(+2.08)
S	M	32.94	8.19	13.11	53.13	16.62	25.32(+12.21)
	R	22.26	21.99	22.12	37.26	33.71	35.39(+13.27)
	W	50.00	2.63	5.00	76.81	11.62	20.19(+15.19)
W	M	17.88	8.41	11.44	25.55	8.02	12.21(+0.77)
	R	18.06	26.65	21.53	20.52	26.78	23.24(+1.71)
	S	14.44	8.14	10.41	26.58	7.50	11.70(+1.29)

3　基于类型驱动的语法纠错模型设计

本文采用 GECToRm 作为基础模型,将 CGED 任务视为序列标注任务. 由于先纠正一种错误有助于提升其他错误类型的纠正效果,本文将每个训练实例的纠错过程分解为两阶段,使模型能够逐步学习纠错的过程. CTDGC 模型如图 3 所示,包含四个层级:(1)Levenshtein 编辑转换层:将错误文本序列-纠正文本序列对转换为编辑标记形式,作为 GECToRm 模型的输入;(2)两阶段训练层:通过构造中间实例并设计训练策略,对模型进行分阶段训练;(3)StructBERT 编码层:将训练数据输入进行编码,生成一系列向量表示;(4)分类器层:设计不同的目标训练模型,以生成最终的编辑标记序列.

3.1　Levenshtein 编辑转换层

莱文斯坦距离(Levenshtein Distance)是由俄罗斯科学家弗拉基米尔·莱文斯坦于 1965 年提出的一种算法. 该算法通过计算将一个字符串转换为另一个字符串所需的最少编辑操作次数来衡量两个字符串之间的差异,允许的编辑操作包括:字符替换、字符插入和字符删除. 在本研究中,我们首先利用莱文斯坦距离和 MOVE 编辑转换,将错误文本序列-纠正文本序列对转换为编辑标记序列,作为模型所需的纠错训练数据.

3.2　两阶段训练层

两阶段训练的关键在于构造一个中间实例. 具体来说,每个训练实例是由错误句子 X_e 和正确句子 X_c

组成的句子对(X_e,X_c). 为了构造其中间句 X',我们随机选择部分语法错误,并在保持其他语法错误不变的情况下更正. 然后,X'分别与 X_e 和 X_c 配对,生成两个新的句子对:(X_e,X')和(X',X_c). 图 4 展示了从一对句子中构造两个额外训练实例的示例. 在这个例子中,有两个语法错误,分别是“天”和“圆”,我们用“园”替换“圆”,而保持“天”错误不变,这样就形成半更正句“他昨天天去公园散步了”. 需要注意的是,这些构建的训练实例源于原始训练语料库,因此半更正句的语法错误也是人为造成的.

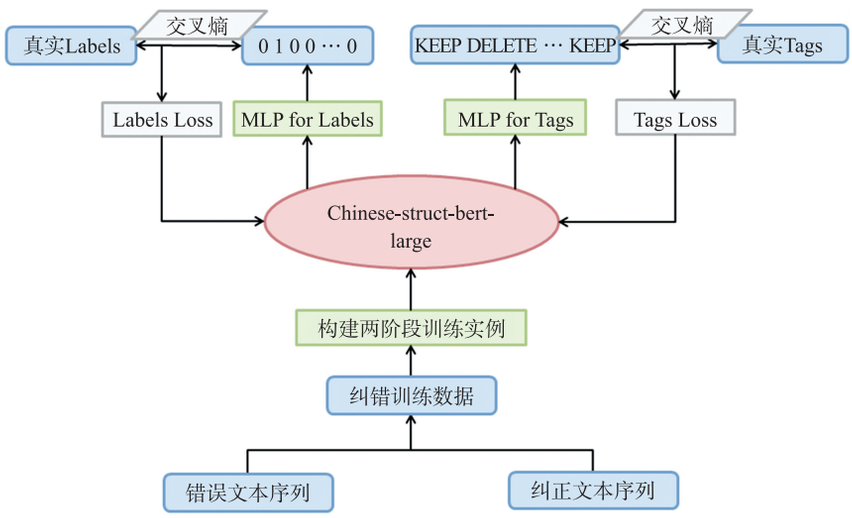


图 3　CTDGC 模型图

Fig. 3　CTDGC model diagram

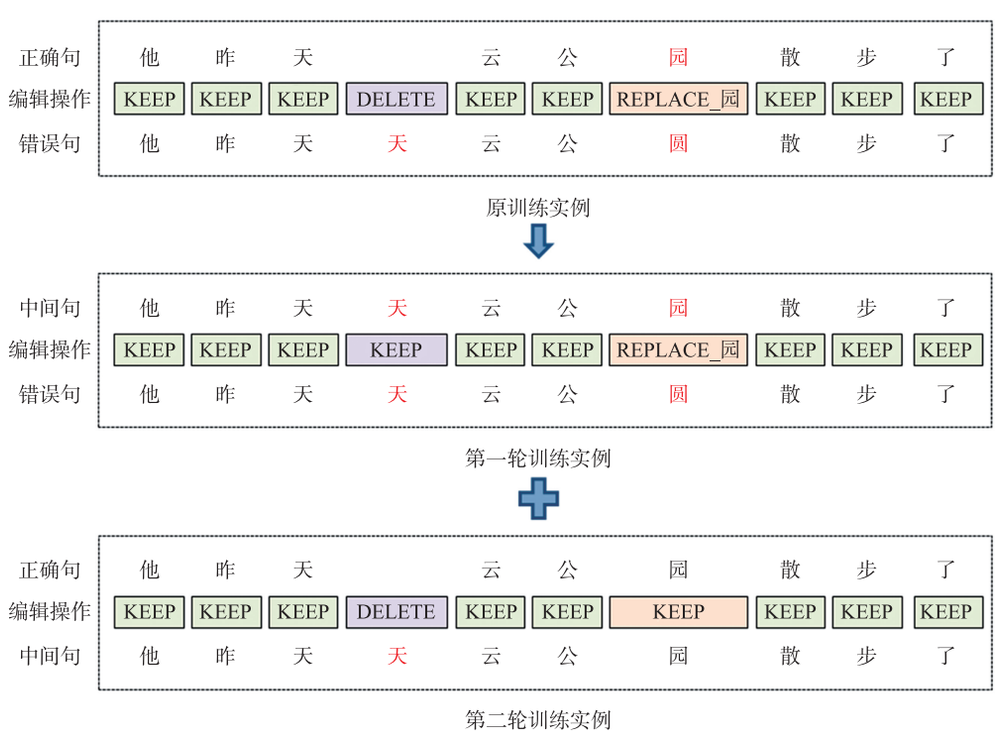


图 4　两阶段训练实例

Fig. 4　Two-Stage training example

接着设计了四种训练策略:APPEND-first、DELETE-first、REPLACE-first 和 MOVE-first. 图 4 的例子即是采用了 REPLACE-first 策略,它是指在第一轮训练中,模型优先学习 REPLACE 校正,即在构建中间句时仅纠正 REPLACE 编辑标签,而保持其他错误不变. 然后,在第二轮训练中,模型再学习纠正其他类型的错误.

3.3　StructBERT 编码层

StructBERT(https://huggingface.co/junnyu/structbert-large-zh)是由阿里巴巴达摩院提出的 NLP 预训

练模型,在传统 BERT 的基础上进行了改进,增强了模型重构打乱句子的能力. 与 BERT 一样,StructBERT 对 15%的 token 进行 MASK 处理,但它进一步随机选择未 MASK 的词组成 trigram,将其顺序打乱后输入 softmax 分类器,以预测原始 token. 模型的目标是最大化打乱顺序前原始序列的重构概率. 这种方法与中文语法纠错中的乱序纠错任务类似,因此,利用 StructBERT 编码层可以充分捕捉错误文本序列中字和词级别的信息,同时对上下文进行建模,从而提高序列标注任务的准确性.

3.4　分类器层

我们为模型的连续训练设置了不同的目标. 具体来说,定义第一轮的训练目标 $L_c^{(1)}$ 和第二轮的训练目标 $L_c^{(2)}$,分别对应公式(1)和公式(2). 由于与原始训练实例相比,我们构建的额外训练实例中包含的语法错误较少,导致模型在纠错时面临较少的语法错误,这可能影响模型的训练效果. 为了解决这一问题,我们将原始训练实例也纳入模型训练,并使用公式(3)定义训练目标 L,最后训练目标为 $L'=L_c^{(1)}+L_c^{(2)}+L$.

$$L_c^{(1)}=-\sum_{i=1}^{N}\mathbb{I}(t_i',t_i)\cdot\log p(t_i'|X_e,\theta),\tag{1}$$

$$L_c^{(2)}=-\sum_{i=1}^{\bar{N}}\log p(\bar{t}_i|X',\theta),\tag{2}$$

$$L=-\sum_{i=1}^{\tilde{N}}\log p(y_i,\theta)+\log p(t_i|X_e,\theta),\tag{3}$$

$$\mathbb{I}(t_i',t_i)=\begin{cases}0, & \text{if } t_i'\neq t_i\\ 1, & \text{if } t_i'=t_i\end{cases},\tag{4}$$

式中,θ 代表模型参数,X_e、X'、X_c 分别是错误句子、中间构造句子、参考句子,$\{t_i'\}_{i=1}^{N}$、$\{\bar{t}_i\}_{i=1}^{\bar{N}}$、$\{t_i\}_{i=1}^{\tilde{N}}$分别是训练实例$(X_e,X')$、$(X',X_c)$、$(X_e,X_c)$的编辑动作标签序列,$\{t_i\}_{i=1}^{N}$是选择的训练策略在第一轮中优先纠正的编辑标签,$\{y_i\}_{i=1}^{\tilde{N}}$是二分类判断正误的标签序列,指示函数$\mathbb{I}(t_i',t_i)$使用公式(4),用于屏蔽第一轮训练中未纠正错误带来的损失影响.

4　基于类型驱动的语法纠错实验

4.1　参数设置及评价标准

本文使用 Lang8 和 CGED 历年的训练集作为训练数据集,以 CGED 历年测试集的数据作为验证集,并将 CGED2020 的数据作为测试集. 后文数据集也采用此数据集,将不再赘述. 实验中采用的预训练模型为 Chinese-struct-bert-large,具体的数据划分和模型超参数设置详见表 3 和表 4.

表 3　数据集划分

Table 3　Dataset partitioning

数据集	句子数量(条)	用途
Lang8	1 102 794	预训练、第一阶段微调
CGED 历年训练集	38 956	第二阶段微调
CGED 历年测试集	2 783	验证集
CGED2020	1 457	测试集

表 4　超参数设置

Table 4　Hyperparameter setting

超参数	含义	值
epochs	迭代次数	10
Batch_size	单批次样本数量	8
Learning Rate	初始学习率	1e-5
Patience	早停轮数	3
Max_len	最大输入长度	200

本文实验采用了 Zhang 等[12]提出的 CHERRANT 评价方法. 该方法基于字符进行评价,从而避免了分词错误对实验结果的影响. 具体的计算公式如式(4)~式(6)所示.

$$P=\frac{TP}{TP+FP},\tag{4}$$

$$R=\frac{TP}{TP+FN},\tag{5}$$

$$F_{0.5}=\frac{(1+0.5^2)*P*R}{0.5^2*P+R},\tag{6}$$

式中,TP 表示模型正确纠正的字符数量,FP 表示模型错误纠正的字符数量,FN 表示模型未能正确纠正的

字符数量.

4.2 实验结果与分析

本文在 CGED2020 数据集上进行实验,选用 GECToR 和 GECToRm 模型作为基线模型,并在此基础上进行了语法错误类型驱动实验. 实验结果如表 5 所示. 通过在 CGED2020 数据集上的实验比较发现,添加了 MOVE 标签的 GECToRm 模型相较于仅包含三种编辑标签的 GECToR 模型,$F_{0.5}$ 得分提升了 1.26%. 即使在采用相同的优先策略下,GECToRm 模型的表现也优于 GECToR 模型. 这表明,针对 CGED 特点设计的 GECToRm 模型确实具有更好的纠错效果. 此外,无论是采用随机策略还是设计的四种训练策略,GECToRm 模型的性能均有所提升,表明先纠正一种错误再纠正另一种错误的两阶段迭代纠错方法是有效的. 特别是 REPLACE-first 策略,在两个基线模型中的提升效果最为显著,相比 GECToRm 模型,$F_{0.5}$得分提升了 1.21%. 这可能是因为先纠正错词错误对其他三种错误的促进作用最大,这与本文之前探讨的四种错误依赖关系的实验结果一致.

表 5 中文语法纠错实验结果

Table 5 Experimental results of CGED %

模型	P	R	$F_{0.5}$
GECToR	34.93	**23.19**	31.71
GECToR-random	35.43	22.99	31.97
GECToR-delete	36.21	23.31	32.60
GECToR-replace	38.39	22.62	33.69
GECToR-append	37.65	23.02	33.41
GECToRm	38.70	20.71	32.97
GECToRm-random	39.80	19.72	33.07
GECToRm-delete	**45.18**	16.98	33.91
GECToRm-replace	45.03	17.41	**34.18**
GECToRm-append	41.74	19.21	33.81
GECToRm-move	39.91	19.99	33.28

表 6 不同模型的中文语法纠错实验结果对比表

Table 6 Comparison of CGED for different models %

模型	P	R	$F_{0.5}$
Transformer	31.69	11.43	23.39
BERT-fuse[13]	31.50	14.99	25.81
BART	38.89	**20.13**	32.78
BART+MaskGEC[14]	40.72	18.63	32.91
Seq2action[11]	26.82	16.12	23.68
Pointer-generator	40.75	18.46	32.82
SG-GEC[7]	40.97	20.05	33.90
Our model	**45.03**	17.41	**34.18**

通过在 CGED2020 数据集上对本文模型与以往研究的对比,实验结果如表 6 所示. 本文模型的 $F_{0.5}$ 得分比当前最优的 SG-GEC 模型高出 0.28 个百分点,比 Pointer-generator 模型高出 1.36 个百分点. 值得注意的是,这两种模型都是基于序列生成的方法,在推理速度上也要低于本文所提出的基于序列标记的方法. 因此,本文模型在 CGED2020 数据集上实现了单模型的最佳性能,能够更高效地完成 CGED 任务.

5 大模型融合

不同的模型能够学习到不同的语义特征,因此,将多个性能略逊但差异较大的模型组合在一起,可以显著提升整体性能. 随着大模型的快速发展,它们在自然语言处理中的应用效果显著提高. 然而,大模型在应用于 CGED 任务时可能会引发过纠正问题,即对正确的句子进行不必要的改写,从而导致假阳性率上升,进而降低整体性能. 为此,我们将大模型与精心设计的模型相结合,以充分发挥大模型的优势,同时缓解其过纠正问题. 本文采用了 CTDGC 中的 GECToRm-replace 模型、Baichuan_3 模型、Pointer-generator 模型、Seq2action 模型和 CorGLM 模型进行融合.

5.1 CorGLM 模型

ChatGLM-6B 是清华大学提出并开源的对话语言大模型,基于 General Language Model(GLM)架构,支持中英双语,拥有 62 亿参数. 然而,直接使用 ChatGLM-6B 进行中文语法纠错时,效果并不理想. 为此,我们通过 LoRA(低秩分解适配器)进行微调,模拟参数的变化,以极小的参数量实现对大模型的间接训练,从而得到 ChatGLM-LoRA 模型. 此外,由于两阶段训练可以提升模型性能,我们首先将拼写纠错 LoRA 模型与 ChatGLM 模型进行合并,然后在 Lang8 和 CGED 历年训练集上进一步微调,最终生成 CorGLM 模型,如图 5 所示.

5.2　Baichuan 大模型

Baichuan-7B 是由百川智能开发的一个开源可商用的大规模预训练语言模型. 基于该模型,本文设计并测试了不同的 Prompt,以评估其性能. 其中,Prompt1 由角色、要求和任务组成,为“你是一个中文语法纠错系统,可以修改输入句子中的语病. 要求:只修改最明显的语病,尽可能减少修改量. 只返回修改后的结果,不需要对结果进行解释和说明. 请以‘修改后的句子是:xxx’的格式输出修改后的句子.”;Prompt2 在 Prompt1 的基础上进一步要求大模型逐步修改错误,为“你是一个中文语法纠错系统,可以修改输入句子中的语病. 要求:在修改错误时,你首先需要理解整个句子的意思,一步步地修改错误. 只修改最明显的语病,尽可能减少修改量. 只返回修改后的结果,不需要对结果进行解释和说明. 请以‘修改后的句子是:xxx’的格式输出修改后的句子.”;Prompt3 则受到先纠正用词错误有助于提升模型纠错性能的研究启发,在 Prompt2 的基础上,要求模型优先修改用词错误,为“你是一个中文语法纠错系统,可以修改输入句子中的语病. 要求:在修改错误时,你首先需要理解整个句子的意思,然后先修改错词错误,一步步地修改错误. 只修改最明显的语病,尽可能减少修改量. 只返回修改后的结果,不需要对结果进行解释和说明. 请以‘修改后的句子是:xxx’的格式输出修改后的句子.”.

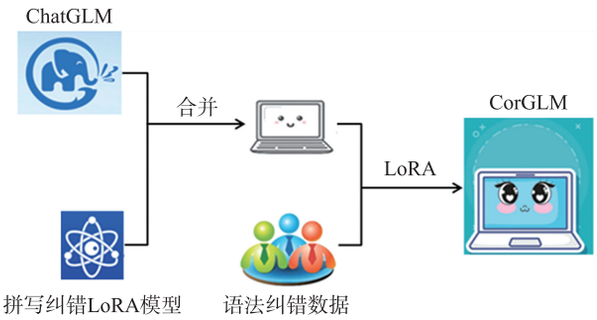


图 5　CorGLM 模型

Fig. 5　CorGLM model diagram

5.3　模型融合

本文基于源句子和五个单一模型的输出,提出了三种集成策略. 第一种是传统投票法,不同模型对最终结果进行投票. 对于每个句子,我们认为不少于 T 个模型推荐的编辑操作是正确的. 我们使用了 Zhang 等[12]提供的代码实现这一投票策略,并对 T 值从 1 到 5 进行了实验. 第二种是句子级别的融合,利用 BERT 语言模型计算源句子和五个单一模型输出的困惑度,选择困惑度最低的输出作为最终结果. 第三种是编辑级别的融合,对于每个编辑操作生成新句子,并计算这些新句子的困惑度,然后选择困惑度最低的编辑操作作为最终的编辑结果. 我们使用了 Zhao 等[15]提供的代码来实现句子级和编辑级的融合策略,三种模型融合策略如图 6 所示.

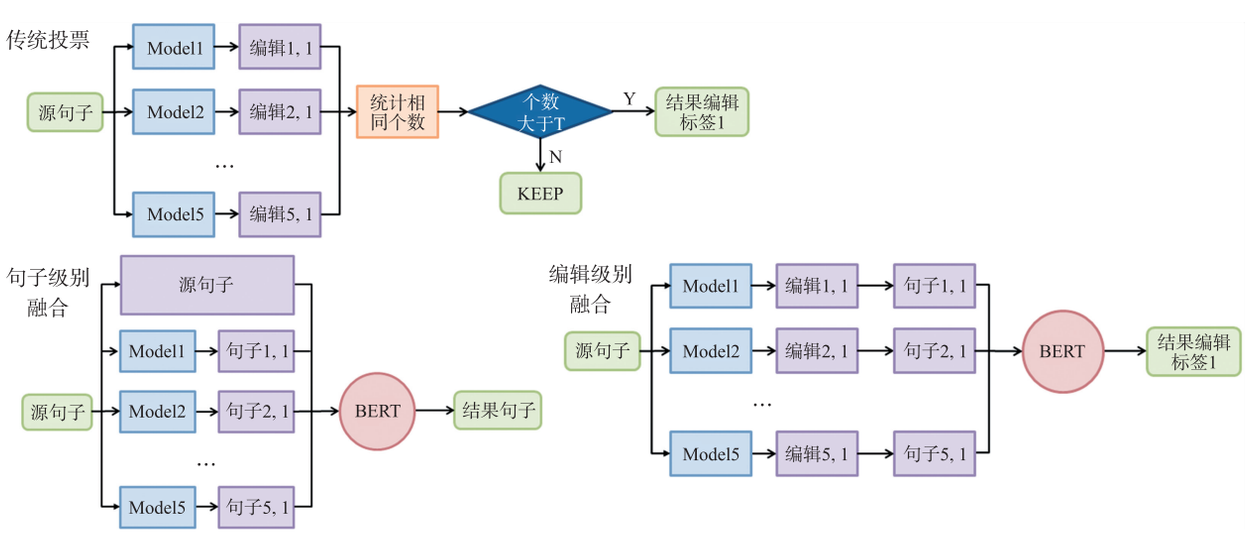


图 6　模型融合策略

Fig. 6　Model fusion strategy

5.4　实验结果与分析

在 CGED2020 测试数据集上,我们首先对单个大模型进行了实验,实验结果如表 7 所示. 直接使用 ChatGLM 进行语法纠错的效果很差,但经过 LoRA 微调后,性能提升了 14.83%,表明微调使大模型在 CGED 数据上表现更好. 此外,CorGLM 相较于 ChatGLM-LoRA 提升了 0.9%,这说明两阶段训练是有效

的. 同时,在对 Baichuan 大模型进行 Prompt 设计实验时,我们的方法也表现出了有效性. 具体而言,Baichuan_2 模型在 Baichuan_1 模型基础上增加了分步纠错,性能提升了 1.2%;Baichuan_3 模型在 Baichuan_2 模型基础上增加了先纠正用词错误的 Prompt,性能提升了 0.26%. 这些结果表明我们的研究对大模型 Prompt 设计具有实际意义.

表 7　大模型实验结果

Table 7　Experimental results of large language models　%

模型	P	R	$F_{0.5}$	模型	P	R	$F_{0.5}$
ChatGLM	8.54	11.12	8.96	Baichuan_1	13.5	21.82	14.61
ChatGLM-LoRA	26.77	16.47	23.79	Baichuan_2	14.86	21.25	15.81
CorGLM	28.07	16.66	24.69	Baichuan_3	15.14	21.31	16.07

在本研究中,尽管大模型存在过度纠正的问题,导致其在性能上不及精心设计的模型,但当它与深度学习模型相结合时,却展现出了意想不到的效果. 我们将 CTDGC 中的 GECToRm-replace 模型和 Pointer-generator 模型分别与 CorGLM、Baichuan_3 和 Seq2action 模型进行了融合,实验结果如表 8 所示. 虽然大模型与 Seq2action 模型的性能相当,但在模型融合后,性能进一步提升. 这表明,利用大模型辅助 CGED 是一种有效的策略.

表 8　三种模型的融合实验结果

Table 8　Experimental results of the fusion of three models　%

模型	CorGLM			Baichuan_3			Seq2action		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$	P	R	$F_{0.5}$
投票	**60.7**	17.14	**40.24**	55.68	17.92	39.17	55.56	17.06	38.28
句子级	34.77	21.3	30.86	21.22	**23.75**	21.69	37.49	20.76	32.29
编辑级	32.86	19.93	29.09	22.71	21.61	22.48	35.00	19.39	30.15

根据表 8 所示,BERT 语言模型在句子级和编辑级的模型融合效果不如投票策略融合. 这表明,在模型融合过程中,由于预训练语言模型计算的 PPL 与 $F_{0.5}$ 目标之间存在不一致性,导致了性能的差异. 因此,本文采用了投票策略对五个模型进行融合,并通过调整阈值 T 进行了实验,结果如表 9 所示. 随着阈值 T 的增加,P 值逐渐上升,而 R 值则逐渐下降. 这表明,随着阈值 T 的提高,模型的纠正准确率显著提升,但提供的纠正选项相应减少. 当 T 值为 2 时,$F_{0.5}$ 值达到最大,CGED 的性能最优,为 40.33%. 这一融合策略相比最佳单一模型的性能提升了 6.15%,相比实现最佳性能的 SG-GEC 模型提升了 6.43%.

表 9　五种模型投票融合实验结果

Table 9　Experimental results of voting fusion of five models　%

阈值 T	P	R	$F_{0.5}$
1	17.25	33.23	19.08
2	47.28	25.41	**40.33**
3	66.35	15.10	39.52
4	78.18	8.08	28.60
5	84.71	3.87	16.35

6　结论

本文通过探究中文语法中四种错误的相互依赖关系,在 GECToRm 的基础上提出基于类型驱动的 CTDGC 模型,实验表明,采用 REPLACE-first 策略的模型性能最佳,$F_{0.5}$ 达到 34.18%,单模型性能优于以往的方法. 此外,通过大模型辅助中文语法纠错,并采用模型投票融合策略,$F_{0.5}$ 进一步提升至 40.33%. 未来,我们将继续研究中文语法纠错,不仅研究非中文母语者的中文语法纠错,还要拓宽研究面,探究针对中文母语者的中文语法纠错,并进一步优化模型以提高其性能.

[参考文献]

[1] RAO G, ZHANG B, XUN E, et al. IJCNLP-2017 task 1: Chinese grammatical error diagnosis[C]//Proceedings of the IJCNLP 2017, Shared Tasks. Taipei, China: Asian Federation of Natural Language Processing, 2017: 1-8.

[2] LAI S, ZHOU Q, ZENG J, et al. Type-driven multi-turn corrections for grammatical error correction[C]//Findings of the Association for Computational Linguistics. Dublin, Ireland: Association for Computational Linguistics, 2022: 3225-3236.

[3] FU K,HUANG J,DUAN Y. Youdao's winning solution to the NLPCC-2018 task 2 challenge:a neural machine translation approach to Chinese grammatical error correction[C]//Proceedings of NLPCC-2018. Hihhot,China:Springer,Cham,2018:341-350.

[4] REN H,YANG L,XUN E. A sequence to sequence learning for Chinese grammatical error correction[C]//Natural Language Processing and Chinese Computing. Hihhot,China:Springer,Cham,2018:401-410.

[5] WANG H,KUROSAWA M,KATSUMATA S,et al. Chinese grammatical correction using BERT-based pre-trained model [C]//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: Association for Computational Linguistics,2020:163-168.

[6] LI P,SHI S. Tail-to-Tail non-autoregressive sequence prediction for Chinese grammatical error correction[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. Online:Association for Computational Linguistics,2021:4973-4984.

[7] WU X,WU Y. From spelling to grammar:a new framework for Chinese grammatical error correction[C]//Conference on Empirical Methods in Natural Language Processing. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics,2022:889-902.

[8] CHEN P,WU S,CHEN L,et al. Chinese grammatical error diagnosis by conditional random fields[C]//Proceedings of the 2nd Workshop on Natural Language Processing Techniques for Educational Applications. Beijing, China: Association for Computational Linguistics,2015:7-14.

[9] OMELIANCHUK K,ATRASEVYCH V,CHERNODUB A,et al. GECToR-grammatical error correction:tag,not rewrite[C]// Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications. Seattle,WA,USA: Association for Computational Linguistics,2020:163-170.

[10] LIANG D,ZHENG C,GUO L,et al. BERT enhanced neural machine translation and sequence tagging model for Chinese grammatical error diagnosis [C]//Proceedings of the 6th Workshop on Natural Language Processing Techniques for Educational Applications. Suzhou,China:Association for Computational Linguistics,2020:57-66.

[11] LI J,GUO J,ZHU Y,et al. Sequence-to-Action:grammatical error correction with action guided sequence generation[C]// AAAI Conference on Artificial Intelligence. Online:AAAI press,2022:10974-10982.

[12] ZHANG Y,LI Z,BAO Z,et al. MuCGEC:a multi-reference multi-source evaluation dataset for Chinese grammatical error correction[C]//Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics:Human Language Technologies. Seattle, United States: Association for Computational Linguistics, 2022:3118-3130.

[13] TANG C,WU X,WU Y. Are pre-trained language models useful for model ensemble in Chinese grammatical error correction [C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. New York,USA:AAAI Press, 2023:893-901.

[14] KANEKO M,MITA M,KIYONO S,et al. Encoder-Decoder models can benefit from pre-trained masked language models in grammatical error correction[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Toronto,Canada:Association for Computational Linguistics,2020:4248-4254.

[15] ZHAO Z,WANG H. MaskGEC:improving neural grammatical error correction via dynamic masking[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Toronto,Canada:Association for Computational Linguistics. Online:Association for Computational Linguistics,2020,34(1):1226-1233.

[责任编辑:杜忆忱]