

基于差分融合句法特征的英语语法纠错模型^①



罗 松, 汪春梅, 袁非牛, 戴 维

(上海师范大学 信息与机电工程学院, 上海 201418)

通信作者: 罗 松, E-mail: 2606982224@qq.com

摘 要: 当前的英文语法纠错模型往往忽略了有利于语法纠错的文本句法知识, 从而使得英语语法纠错模型的纠错能力受到影响. 针对上述问题, 提出一种基于差分融合句法特征的英语语法纠错模型. 首先, 本文提出的句法编码器不仅可以直接从文本中无监督地生成依存关系图和成分句法树信息, 而且还能将上述两种异构的句法结构进行特征融合, 编码成高维的句法表征. 其次, 为了同时利用文本中的语义和句法信息, 差分融合模块先使用差分正则化加强语义编码器捕获句法编码器未能生成的语义特征, 然后采用协同注意力将句法表征和语义表征进一步融合, 作为 Transformer 编码端的输出特征, 最终输入到解码端, 从而生成语法正确的文本. 在 CoNLL-2014 英文纠错任务数据集上进行对比实验, 结果表明, 该方法的准确率和 $F_{0.5}$ 值优于基于 Copy-Augmented Transformer 的语法纠错模型, 其 $F_{0.5}$ 值提升了 5.2 个百分点, 并且句法知识避免了标注数据过少问题, 具有更优的文本纠错效果.

关键词: 自然语言处理; 语法纠错; 句法知识; 协同注意力; 差分融合

引用格式: 罗松, 汪春梅, 袁非牛, 戴维. 基于差分融合句法特征的英语语法纠错模型. 计算机系统应用, 2023, 32(10): 293-300. <http://www.c-s-a.org.cn/1003-3254/9259.html>

Grammatical Error Correction Model Based on Differential Fusion Syntactic Feature

LUO Song, WANG Chun-Mei, YUAN Fei-Niu, DAI Wei

(School of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai 201418, China)

Abstract: Current English GEC methods tend to ignore the syntactic knowledge in texts, which plays an important role in grammatical error correction, and thus the error correction ability of English GEC models is affected. To address this problem, the study proposes a GEC method which is based on the differential fusion syntactic features. First, the proposed syntactic encoder can generate dependency graph and constituency syntactic tree information from raw data in an unsupervised way and conduct the feature fusion of these two heterogeneous syntactic structures to encode high-dimensional syntactic representation. Second, to utilize both semantic and syntactic information in the text, the differential fusion module first uses differential regularization to enhance the semantic encoder to capture the semantic features that the syntactic encoder fails to generate. Then the syntactic representation and semantic representation are further fused by cross attention as the output features of the Transformer encoder, which are finally input to the decoder to generate grammatically correct text. The comparison experiment on the CoNLL-2014 task dataset shows that the precision and $F_{0.5}$ value of this method are better than those of the GEC model based on the Copy-Augmented Transformer, and the $F_{0.5}$ value of this method is improved by 5.2 percentage points. The syntactic knowledge avoids the problem of lacking high-quality annotated training corpora and has a better performance in text error correction.

Key words: natural language processing (NLP); grammatical error correction; syntactic knowledge; cross attention; differential fusion

① 基金项目: 国家自然科学基金 (62272308)

收稿时间: 2023-03-24; 修改时间: 2023-04-28; 采用时间: 2023-05-06; csa 在线出版时间: 2023-07-14

CNKI 网络首发时间: 2023-07-17

语法纠错 (grammatical error correction, GEC) 任务旨在利用自然语言处理技术, 自动识别并纠正非英语母语学习者书写的文本中所包含的动词时态、拼写、标点符号等语法错误^[1]. 如表 1 所示, 第 1 行表示系统的输入, 第 2 行表示系统的输出, 加粗单词“about”属于句子中多余的单词.

表 1 语法纠错系统的输入输出示例

输入/输出	示例
输入	I appreciate all about this.
输出	I appreciate all this.

现阶段的 GEC 常被视为神经机器翻译任务的特例^[2], 将标准的 encoder-decoder 机器翻译模型, 运用到语法纠错任务中. 除此之外, 基于预训练模型的语法纠错模型也渐渐的展露其优势^[3]. 尽管上述工作均大幅度地提高了语法纠错的性能, 但都只是将输入句子处理成 token 序列, 都未能充分挖掘句子的语义 (semantic) 和句法 (syntactic) 信息.

句法作为语法的一个分支, 包含句子中单词的排列次序、短语组成等结构信息. 根据句法结构的表现形式, 可分为依存关系图 (dependency graph) 和成分句法树 (constituent syntactic tree). 前者是描述词与词之间的相互依存关系, 后者则是关注短语及短语之间的层次关系. 多项自然语言处理的下游任务, 如机器翻译^[4]、语义角色标注^[5]、情感分析^[6], 表明添加句子的句法信息能够提高模型的性能. 因此, 在 GEC 任务中, 希望模型可以更多地提取句法层面的信息, 捕捉源语句与目标语句的句法差异, 提高模型的纠错性能.

语法纠错目前存在两个问题. 首先, 语法纠错任务中难以获得大量的标注数据, 模型训练往往达不到很好的效果. 其次, 源语句与目标语句之间可能只存在一个或几个字符有所不同, 导致源语句与目标句子之间的文本编辑距离很小, 很难定位错误的字符. 但在句法结构上存在较大的差异. 如图 1 所示, 在例句“I appreciate all this.”中, 如果输入时将冗余的“about”写入句子中, 可以明显地看出错误句子 (ungrammatical) 的句法结构发生了很大的变化.

不过, 语法纠错任务下的输入句子往往存在语法错误, 现有的句法解析模型解析出来的句法信息存在错误.

针对以上问题, 本文在 Transformer 纠错模型基础上, 新增了句法编码器和差分融合两个模块. 本文提出

的模型可以将句法特征融入到 Transformer 纠错模型中. 句法编码器不仅能够从原始句子中无监督地生成依存关系图和成分句法树信息, 而且更进一步地将两者进行特征融合, 形成深层次的句法向量信息. 差分融合模块将句法向量信息和 Transformer 捕获的语义向量信息进行差分融合, 使模型可以更有效地利用句法特征信息, 提高 GEC 模型的性能. 实验结果表明所提模型的准确率 P (precision)、召回率 R (recall) 和 $F_{0.5}$ 都有很大的提高.

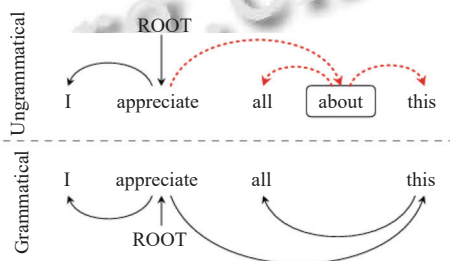


图 1 因冗余的字引发的句法结构变化实例

1 相关工作

研究者将语法纠错任务视为机器翻译的特例^[7], 序列到序列的网络结构常被用于语法纠错任务中. 其中, Transformer 语法纠错模型成为基线模型^[8]. 由于 BERT^[9]、BART^[10] 等预训练语言模型的语言性能优异, 最近被广泛应用在语法纠错模型的编码端, 获取丰富的语义信息, 提高了语法纠错性能. 语法纠错任务目前存在的挑战依然是缺少大量的数据集. 为此, 多种数据增强的方法涌现出来^[11], 一方面使用人工预设的语法错误规则对正确的句子进行噪声扰动, 从而合成平行句子对; 另一方面则是反向翻译的方式, 自动生成平行句子对. 为了缩短人造数据与标注数据的分布差异, 对抗学习 (adversarial learning) 也常被使用到数据增强模块中^[12]. 与此同时, 部分研究者将语法纠错任务看作序列到编辑 (sequence-to-edit, Seq2Edit), 不再直接预测正确的句子序列, 而是预测句子的编辑操作, 从而对输入句子的语法错误进行修改^[13]. 除此之外, 结合句子间的上下文信息^[14]、模型集成和重排序技术也能提高语法纠错模型的性能^[15].

在语法纠错研究中, 研究者们也探索了将句法信息融入到语法纠错模型. Li 等人^[16] 采用了现有的句法解析模型对输入句子进行解析, 获取依存关系图和成

分句法树的信息,然后通过图注意力机制将依存关系图和成分句法树的信息分别编码到语法纠错模型中。但 Li 等人只利用了单一的句法结构,既没有考虑两者融合的句法结构信息,也没有考虑现有的句法解析模型处理存在语法错误的句子可能会生成错误的依存关系图或成分句法树信息的问题。

早期的论文工作证明,依存关系图和成分句法树具有互补性,两者融合能提高下游任务的性能^[17]。本文将依存关系图和成分句法树的融合成互补的依存-成分句法信息,输入到语法纠错模型中,最终生成语法正

确的句子。

2 本文模型

为了将句法知识融入语法纠错模型中,本文提出一种新颖的差分融合句法特征的英语语法纠错模型,实现英语文本的语法纠错。在 Transformer 结构外添加了句法融合器 (syn-fusion)、句法编码器 (syn-encoder) 和差分融合模块 (diff-fusion), 框架如图 2 所示。其中 N 、 N_1 、 N_2 、 N_3 分别是句法融合器卷积层的层数、句法编码器、语义编码器和解码器的个数。

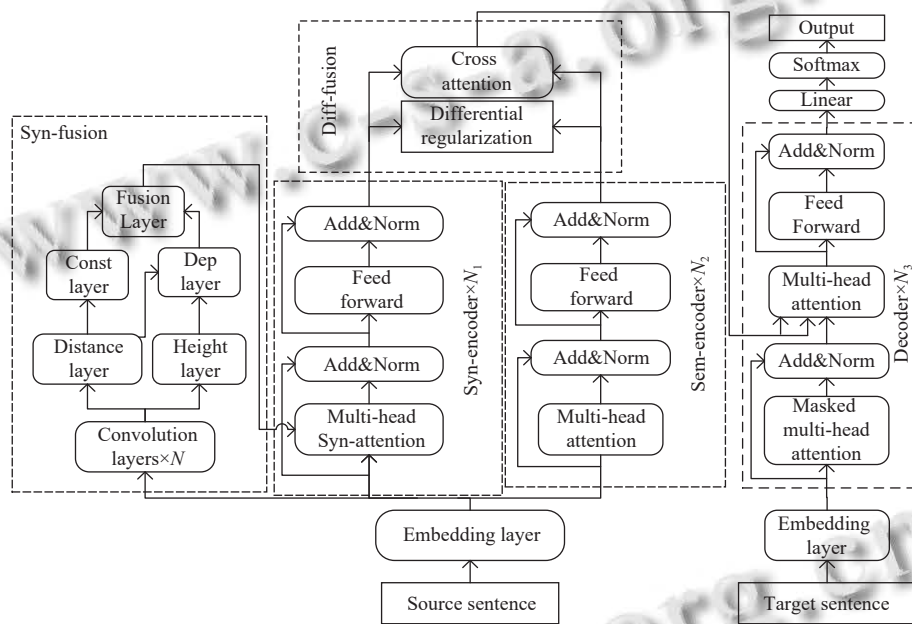


图 2 模型整体框架图

首先输入句子经过词嵌入层 (embedding layer), 得到句子向量, 作为语义编码器、句法编码器以及句法融合器的输入。其次, 语义编码器将句子向量编码为高维隐含语义特征, 句法编码器将句法融合器生成的依存-成分句法向量与句子向量编码为高维隐含句法特征, 然后通过协同注意力将语义特征和句法特征进行特征融合, 最后将其输入到解码器, 从而生成正确句子。特别的, 在句法编码器和语义编码器之间设计了一个差分正则器 (differential regularization), 用于减少句法和语义的重叠信息, 鼓励语义编码器生成句法编码器未能捕捉到的语义特征。

2.1 Transformer 模型

Transformer 模型是一个可获取句子深层语义信息

的端到端的序列生成模型, 由编码器和解码器两个部分组成, 即图 2 中的语义编码器 (sem-encoder) 和解码器 (decoder)。

语义编码器由 N_2 个相同的神经模块组成, 每一模块由两个子层组成, 分别是多头注意力层 (multi-head attention) 和前馈层 (feed forward)。其中, 多头注意力层是由多个注意力层拼接组成的, 每个注意力层计算采用的是放缩点乘 (scaled dot-product), 如式 (1) 和式 (2) 所示。

$$Att(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

$$\text{MultiHead} = [Att_1; Att_2; \dots; Att_h] \quad (2)$$

其中, Q 、 K 和 V 分别表示注意力层的查询矩阵 (query)、

键值矩阵 (key) 以及实值矩阵 (value), 它们是由输入向量经过 3 个不同的线性层得到, 用以计算输入向量的注意力 Att . d_k 表示输入向量的特征维度大小, $MultiHead$ 为多个注意力的拼接.

前馈层 ($FFN(x)$) 是由两个线性层串行连接而成, 其计算如式 (3) 所示.

$$FFN(x) = ReLU(xW_1 + b_1)W_2 + b_2 \quad (3)$$

其中, W_1 、 b_1 、 W_2 和 b_2 均为可学习参数, $ReLU$ 为激活函数.

解码器也是由 N_3 个相同的模块构成, 它的功能是结合编码器生成的高维向量预测目标句子. 解码器在编码器的基础上多了一个掩码多头注意力层 (masked multi-head attention), 目的是保证模型不能看到未预测词的信息.

为了防止过拟合和加快模型收敛速度, 语义编码器和解码器的每个子层后面都有一个残差结构和归一化层 (Add&Norm).

实际上, 纠错任务的源语句和目标语句的绝大部分单词是相同的. 因此本文引入了复制机制, 直接从源句子中复制正确的单词, 可以很好地解决未登录词 (out-of-vocabulary words) 的问题.

2.2 句法融合器

在介绍句法融合器时, 先介绍句法距离^[18]和句法高度^[19]的定义以及两者的联系.

句法距离是用来记录将句子分割成更小成分的过程. 分割点的顺序记录了句子的层次结构, 通过预测句法距离序列就可以得到成分句法树.

句法高度是用来描述依存关系图中非根节点到根节点的距离. 设 D 为句子 $(w_1, w_2, \dots, w_{n-1})$ 的依存关系图, D 中的 w_i 的高度为 δ_i . 依存关系图 D 的句法高度 $\Delta = (\delta_1, \delta_2, \dots, \delta_n)$ 可以由任意 n 个实值元素 δ 组成的序列, 但要求与 $(\delta_1, \delta_2, \dots, \delta_n)$ 保持同样大小关系顺序. 由于不知道单词应该是属于父节点的左边还是右边, 所以无法通过句法高度重构原始的依存关系图. 针对此问题, 将无标签成分句法树的父节点 (分割点) 与最大句法高度的 token 一一对应, 借助句法距离将无标签的成分句法树转换为依存关系图.

句法融合模块目的是从输入的句子向量中直接生成融合的依存-成分句法信息. 首先, 将词嵌入层输出的句子向量输入到 N 层卷积网络中, 如式 (4) 所示.

$$S_{l,i} = \tanh(\text{conv}(S_{l-1,i-w}, \dots, S_{l-1,i+w})) \quad (4)$$

其中, $S_{l,i}$ 代表第 l 层的第 i 个位置输出, $S_{0,i}$ 是单词 w_i 的词向量, 卷积核大小为 $2w + 1$, \tanh 为激活函数.

其次, 将 N 层卷积网络的输出向量 $S_{N,i}$ 输入到距离层 (distance layer) 和高度层 (height layer) 分别得到单词 w_i 和 w_{i+1} 第 i 个分割点的距离 τ_i 和单词 w_i 的高度 δ_i , 如式 (5) 和式 (6) 所示.

$$\tau_i = ReLU(W_1^T [S_{N,i}, S_{N,i+1}]^T) W_2^T \quad (5)$$

$$\delta_i = ReLU(W_1^\delta S_{N,i} + b_1^\delta) W_2^\delta + b_2^\delta \quad (6)$$

其中, W_1^T 、 W_2^T 、 W_1^δ 、 W_2^δ 、 b_1^δ 和 b_2^δ 均为可训练参数, $ReLU$ 为激活函数.

再次, 将句法距离 $T = (\tau_1, \tau_2, \dots, \tau_{n-1})$ 输入到成分句法层 (const layer), 通过式 (7) 计算单词 x_j 和单词 x_i 的成分句法分布概率 $p_{i,j}^{\text{const}}(j|i)$.

$$p_{i,j}^{\text{const}}(j|i) = \text{Sigmoid}(\tanh(\tau_j - \tau_i) + 1) \quad (7)$$

然后, 遵循 Shen 等人^[20]提出的计算方法, 依存关系层 (dep layer) 通过式 (8)–式 (10) 计算单词 x_j 是单词 x_i 父节点的概率 $p_{i,j}^{\text{dep}}(j|i)$.

$$p_C([l, r]|i) = p(\tau_{l-1} < \delta_i) \sum_{l \leq k < r} p(\tau_k > \delta_i) p(\tau_r - \delta_i) \quad (8)$$

$$p_{Pr}(j|[l, r]) = \text{Softmax}(\delta_l, \dots, \delta_r) \quad (9)$$

$$p_{i,j}^{\text{dep}}(j|i) = \sum_{[l,r]} p_{Pr}(j|[l, r]) p_C([l, r]|i) \quad (10)$$

其中, $p_C([l, r]|i)$ 表示包含单词 x_i 最小成分跨度 $[l, r]$ 的概率, p 表示累计分布概率函数, 本文使用了 Sigmoid 函数; $p_{Pr}(j|[l, r])$ 表示单词 x_j 是最小成分跨度 $[l, r]$ 父节点的概率, 对应的句法高度值最大.

最后, 与 Shen 等人^[20]不同的是, 并没有直接使用 $p_{i,j}^{\text{dep}}(j|i)$, 而是新增了门控融合单元将依存关系图概率分布 $p_{i,j}^{\text{dep}}(j|i)$ 和成分句法树概率分布 $p_{i,j}^{\text{const}}(j|i)$ 进行了特征融合, 一方面可以捕获依存关系图和成分句法树两种异构的句法互补信息, 另一方面可以减少语法错误对单一的句法结构影响. 融合层 (fusion layer) 通过一个门控参数 g 平衡两者的概率分布, 如式 (11) 和式 (12) 所示.

$$g = \sigma(W^{(g1)} p_{i,j}^{\text{const}}(j|i) + W^{(g2)} p_{i,j}^{\text{dep}}(j|i) + b^{(g)}) \quad (11)$$

$$p_{i,j}^{\text{fuse}}(j|i) = g \times p_{i,j}^{\text{const}}(j|i) + (1 - g) p_{i,j}^{\text{dep}}(j|i) \quad (12)$$

其中, $W^{(g1)}$ 、 $W^{(g2)}$ 和 $b^{(g)}$ 均为可训练参数, σ 表示 Sigmoid

激活函数; $p_{i,j}^{\text{fuse}}(j|i)$ 表示 x_i 和 x_j 的依存-成分句法概率分布.

2.3 句法编码器

句法编码器同样由 N_1 个相同的层组成, 每一层包含两个子层, 分别是多头句法注意力层 (multi-head syn-attention) 和前馈层.

如果将句子视为单词的全连接图, 可以将 Transformer 的多头注意力层视为完全图 $G = (X, E)$, 其中, 顶点集合 X 表示句子中的所有单词, 边集合 E 是这些顶点之间的所有可能的边 (x_i, x_j) . 句法编码器使用依存-成分句法图 $G = (X, A)$ 代替 G , 其中 $A_{i,j} = p_{i,j}^{\text{fuse}}(j|i)$ 表示 x_i 和 x_j 的依存-成分句法概率分布. 这样, 传统的 Transformer 就可以学习到句子中的句法信息. 本文的多头句法注意力 Att 计算过程如式 (13)–式 (15) 所示.

$$p_{i,j} = w_{\text{parent}} p_{i,j}^{\text{fuse}}(j|i) + w_{\text{dep}} p_{ji}^{\text{fuse}}(i|j) \quad (13)$$

$$w_{\text{parent}} + w_{\text{dep}} = 1 \quad (14)$$

$$Att(Q_i, K_j, V_j, p_{i,j}) = \text{Softmax}\left(\frac{p_{i,j} \cdot (Q_i K_j^T)}{\sqrt{d_k}}\right) V_j \quad (15)$$

其中, w_{parent} 和 w_{dep} 均为可学习参数, $p_{i,j}(j|i)$ 表示 x_j 和 x_i 之间信息传播概率.

然而, 语法纠错任务下的输入句子往往是存在语法错误的, 句法编码器可能会产生语法错误的句法信息. 因此, 对 $p_{i,j}$ 进行了正则化, 类似于 Dropout 方式, 在训练过程中, 主动舍弃部分信息. 通过概率 q 随机地将 $p_{i,j}$ 行向量的值设为 0, 断开某些节点间的依赖关系. 句法编码器通过梯度下降逐渐收敛成一个合理的依存-成分句法关系图.

2.4 差分融合

本文使用差分正则器限制语义编码器的输出, 旨在减少句法编码器和语义编码器输出的信息冗余, 捕获新的语义特征. 计算公式如式 (16) 所示.

$$R_D = \frac{1}{\|H^{\text{sem}} - H^{\text{syn}}\|_F} \quad (16)$$

其中, $\|\cdot\|_F$ 表示 Frobenius 范数.

句法和语义作为句子不同视角的表征, 本文没有简单地将其线性相加, 而是使用协同注意力进一步特征融合, 得到协同注意力的输出 H^{Out} . 协同注意力计算过程如式 (17) 和式 (18) 所示.

$$\begin{aligned} Q_c &= H^{\text{sem}} W_q^T \\ K_c &= H^{\text{syn}} W_k^T \\ V_c &= H^{\text{syn}} W_v^T \end{aligned} \quad (17)$$

$$H^{\text{Out}} = \text{Softmax}\left(\frac{Q_c K_c^T}{\sqrt{d_k}}\right) V_c + H^{\text{sem}} \quad (18)$$

其中, W_q^T 、 W_k^T 和 W_v^T 是可学习参数, H^{sem} 和 H^{syn} 分别是语义编码器和句法编码器的输出特征.

2.5 损失函数

在训练过程中, 采用最小化交叉熵损失函数 (cross-entropy loss function) 评估模型, 同时引入 L2 正则化和差分正则器 R_D 损失函数的计算如式 (19) 所示:

$$\text{loss}_T = - \sum_i y_i \log \hat{y}_i + \lambda_1 R_D + \lambda_2 \|\theta\|^2 \quad (19)$$

其中, λ_1 为差分正则器的系数, $\lambda_2 \|\theta\|^2$ 为 L2 正则化, λ_2 为正则化参数, θ 为模型参数.

3 数据集与模型参数

3.1 数据集及评价指标

Lang-8 语料来自一个在线的英语学习网站 Lang-8, 作为 GEC 公开语料库中最大的一组, 语料嘈杂且没有明确的标注规则, 侧重于动词时态错误. NUCLE 是第一个提供错误标注且免费的语料库, 由专业英语教师进行错误标注, 用于对 GEC 系统的训练与性能测试. 其中, 共有 27 种错误类型, 比例最高的 5 种是搭配错误、冗余错误、限定词错误、名词单复数错误和结构性错误.

FCE 语料库是剑桥学习者语料库的子集, 包括了 7 种错误类型, 9 种母语类型.

本文将 GEC 任务的官方的 CoNLL-2013 数据集作为验证集, CoNLL-2014 测试集评测本文的模型. 所有公开数据集的统计信息如表 2 所示.

表 2 实验数据集统计信息

类别	数据集	句子对
训练集	Lang-8	1097274
	NUCLE	57119
	FCE	32073
验证集	CoNLL-2013 test set	1381
测试集	CoNLL-2014 test set	1312

本文使用最大匹配分数 (max match score, M^2 -score) 评测模型的性能. 此度量方法首先将 GEC 系统输出的纠正句子与人工标注的标准句子进行单词对齐, 计算将一个句子转换成另一个句子所需要的单词级别上的编辑数, 以准确率 P 、召回率 R 和 $F_{0.5}$ 作为模型的评价指标.

3.2 模型参数设置

本文的基线模型是 Copy-Augmented Transformer^[1]. 该方法首次利用复制机制直接将输入句子中正确的字复制到输出端, 可以有效地解决未登录词问题, 提高模型的纠错效率.

本文的模型具体参数如表 3 所示, 其他参数配置与基于 Copy-Augmented Transformer^[1] 语法纠错模型一致.

参数	值	参数	值
Embed_dim	512	卷积层 (N)	3
FF-dim	4096	卷积核大小	9
Heads	8	Dropout	0.2
Lr	0.001	Beam size	12
λ_1	0.2	λ_2	0.0004
语义编码器层数 (N_2)	6	句法编码器层数 (N_1)	6
解码器层数 (N_3)	6		

4 实验

为了证明融合句法特征的语法纠错模型可以有效地提高模型的纠错性能, 进行了消融实验和对比实验.

4.1 消融实验

本文基于 CoNLL-2014 数据集进行 4 组实验, 以验证所提各模块的有效性.

实验 1: 验证句法融合器卷积层层数对本文模型性能的影响, 结果如图 3 所示.

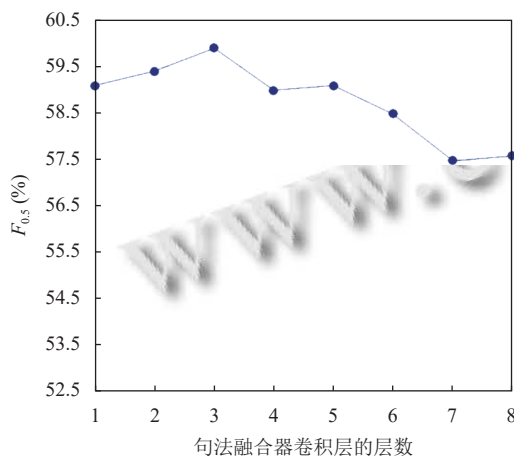


图 3 句法融合器卷积层层数的影响

从图 3 中可以看出, 句法融合器卷积层数为 3 时, 模型取得最高的性能. 一方面是句子的句法结构信息存在较低层; 另一方面, 如果层数过多, 模型参数会大幅度增多, 模型会因梯度的消失变得不稳定.

实验 2: 验证添加不同句法结构对语法纠错模型的纠错性能影响. 第 1 组实验仅使用了基线模型进行纠错; 第 2 组实验是在基线模型的编码端添加成分句法树信息; 第 3 组实验是在基线模型的编码端添加依存关系图信息; 第 4 组是在基线模型的编码端添加依存-成分句法信息, 即成分句法树和依存关系图的融合信息. 实验结果如表 4 所示.

表 4 在 CoNLL-2014 测试集上不同句法结构对模型性能的影响 (%)

模型	P	R	$F_{0.5}$
基线模型 ^[1]	65.2	33.2	54.7
+成分句法树	68.0	36.0	56.5
+依存关系图	68.2	36.3	57.5
+依存-成分信息	69.6	38.0	59.9

从表 4 中可以看出, 相较于基线模型, 第 2、第 3 组较第一组结果都有了明显的提升, 在 $F_{0.5}$ 指标上分别提高了 1.8% 和 2.8%, 验证了语法纠错模型中添加不同的句法信息均能提高语法纠错性能. 由于依存关系图是在成分句法树的基础上得到的, 第 3 组的实验结果相对好一些.

通过对比第 2-4 组实验, 本文提出的依存-成分信息能进一步地提高了语法纠错的性能, $F_{0.5}$ 值提高了 5.2%, 表明了两者不同结构的句法融合存在互补的信息, 能更好地减少解析语法错误的句子所带来的错误句法信息, 增强了模型的鲁棒性.

实验 3: 验证差分正则器 R_D 对模型的纠错性能的影响. 实验结果如表 5 所示.

表 5 在 CoNLL-2014 测试集上 R_D 对模型的影响 (%)

模型	P	R	$F_{0.5}$
未使用 R_D	68.7	36.4	58.2
使用 R_D	69.6	38.0	59.7

由表 5 的实验结果可以得出, R_D 可以减少句法与语义之间冗余的信息, 进而提高模型的纠错性能.

实验 4: 验证语义和句法不同融合方式对模型性能的影响. 实验结果如表 6 所示.

表 6 在 CoNLL-2014 测试集上语义和句法不同融合的结果比较 (%)

模型	P	R	$F_{0.5}$
基准模型 ^[1]	65.2	33.2	54.7
+线性相加	68.4	35.8	57.7
+协同注意力	69.6	38.0	59.7

由表6可以得出,在基线模型中直接添加句法编码器输出的句法信息,使模型提高了3.0%,表明句法特征有利于模型的纠错性能。为了将句法编码器优势和基线模型有效地融合在一起,采用了协同注意力模块,使得Transformer模型可以更好地融合句法特征,在各方面性能优于直接将两者线性相加的方式, $F_{0.5}$ 值又提升了2.0%。

4.2 对比实验

为了验证本文提出模型的有效性,与8种目前先进的英语语法纠错模型在CoNLL-2014测试集上的实验结果进行对比。实验统计数据如表7所示。

表7 不同模型在CoNLL-2014测试集上的结果

组别	Models	Data size (M)	P (%)	R (%)	$F_{0.5}$ (%)
w/o syntax	CNN+EO (4 ens. with LM) ^[7]	2.2	65.5	33.1	54.8
	NMT SMT hybrid (4 ens.) ^[15]	2.0	66.7	34.5	56.2
	Cross-sentence GEC model ^[14]	1.3	64.3	35.9	55.5
	Copy-augmented model ^[1]	30	65.2	33.2	54.7
	Transformer+CNN ^[12]	1.1	64.6	22.6	47.1
	BERT-fuse GED ^[3]	—	63.6	33.0	53.6
w/ syntax	Transformer+BIFI ^[11]	10	64.4	35.6	55.5
	SynGEC ^[16]	30	66.7	38.3	58.1
	本文的模型	1.2	69.6	38.0	59.9

本文将近几年未使用预训练模型的语法纠错模型分为两组,第1组是未使用句法知识的模型(w/o syntax),第2组是使用了句法知识的模型(w/ syntax),其中“4 ens.”表示4个模型集成的结果,“LM”表示利用了语言模型。Data size表示模型使用的训练集大小,1.2M表明模型未使用额外的伪数据集。

由表7可以看出,在没有使用伪数据集、模型集成和语言模型的情况下,除了R指标外,本文提出的模型的P指标和 $F_{0.5}$ 指标均优于其他模型, $F_{0.5}$ 值达到了59.9%,本文提出的深度网络模型的纠错效果总体优于其他算法。相较于使用现有的句法解析器生成句法知识的SynGEC模型,本文模型的P和 $F_{0.5}$ 指标分别提高2.9%、1.8%,但在R指标上略低于SynGEC模型,可能是因为该模型使用了额外30M伪数据集。这些实验结果验证了本文提出的方法可以帮助模型更好地提取句法与语义特征,表明了语法纠错模型中添加句法特征能够有效地提升纠错性能。

5 结束语

本文提出一种新颖的基于差分融合句法特征的英

语语法纠错模型,能很好地将句法特征融入Transformer模型。通过句法编码器对输入句子进行句法特征的提取,并对两种异构的句法结构进行融合。采用差分融合模块将句法信息融入到Transformer模型中,有效地减少句法和语义信息的冗余信息,捕获新的语义特征。实验结果表明,添加句法特征不仅可以有效地提高语法纠错性能,而且还能减轻语法纠错任务缺乏大量标注数据的影响。

参考文献

- Zhao W, Wang L, Shen KW, *et al.* Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data. 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). Minneapolis: Association for Computational Linguistics, 2019. 156–165.
- Junczys-Dowmunt M, Grundkiewicz R, Guha S, *et al.* Approaching neural grammatical error correction as a low-resource machine translation task. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). New Orleans: Association for Computational Linguistics, 2018. 595–606.
- Kaneko M, Mita M, Kiyono S, *et al.* Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2020. 4248–4254.
- Bugliarello E, Okazaki N. Enhancing machine translation with dependency-aware self-attention. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2020. 1618–1627.
- He SX, Li ZC, Zhao H. Syntax-aware multilingual semantic role labeling. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: Association for Computational Linguistics, 2019. 5350–5359.
- Li RF, Chen H, Feng FX, *et al.* Dual graph convolutional networks for aspect-based sentiment analysis. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. Association for

- Computational Linguistics, 2021. 6319–6329.
- 7 Chollampatt S, Ng HT. A multilayer convolutional encoder-decoder neural network for grammatical error correction. Proceedings of the 32nd AAAI Conference on Artificial Intelligence and 30th Innovative Applications of Artificial Intelligence Conference and 8th AAAI Symposium on Educational Advances in Artificial Intelligence. New Orleans: AAAI Press, 2018. 706.
 - 8 Rothe S, Mallinson J, Malmi E, *et al.* A simple recipe for multilingual grammatical error correction. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. Association for Computational Linguistics, 2021. 702–707.
 - 9 Devlin J, Chang MW, Lee K, *et al.* BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis: Association for Computational Linguistics, 2019. 4171–4186.
 - 10 Lewis M, Liu YH, Goyal N, *et al.* BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2020. 7871–7880.
 - 11 Yasunaga M, Leskovec J, Liang P. LM-Critic: Language models for unsupervised grammatical error correction. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Punta Cana: Association for Computational Linguistics, 2021. 7752–7763.
 - 12 Raheja V, Alikaniotis D. Adversarial grammatical error correction. Proceedings of the 2020 Findings of the Association for Computational Linguistics. Association for Computational Linguistics, 2020. 3075–3087.
 - 13 Awasthi A, Sarawagi S, Goyal R, *et al.* Parallel iterative edit models for local sequence transduction. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: Association for Computational Linguistics, 2019. 4260–4270.
 - 14 Chollampatt S, Wang WQ, Ng HT. Cross-sentence grammatical error correction. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: Association for Computational Linguistics, 2019. 435–445.
 - 15 Grundkiewicz R, Junczys-Dowmunt M. Near human-level performance in grammatical error correction with hybrid machine translation. Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: Human Language Technologies, Volume 2 (Short Papers). New Orleans: Association for Computational Linguistics, 2018. 284–290.
 - 16 Li ZC, Parnow K, Zhao H. Incorporating rich syntax information in grammatical error correction. Information Processing & Management, 2022, 59(3): 102891.
 - 17 Fei H, Wu SQ, Ren YF *et al.* Better combine them together! Integrating syntactic constituency and dependency representations for semantic role labeling. Proceedings of the 2021 Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Association for Computational Linguistics, 2021. 549–559.
 - 18 Shen YK, Lin ZH, Jacob AP, *et al.* Straight to the tree: Constituency parsing with neural syntactic distance. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: Association for Computational Linguistics, 2019. 1171–1180.
 - 19 Luo HY, Jiang L, Belinkov Y, *et al.* Improving neural language models by segmenting, attending, and predicting the future. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: Association for Computational Linguistics, 2019. 1483–1493.
 - 20 Shen YK, Tay Y, Zheng C, *et al.* StructFormer: Joint unsupervised induction of dependency and constituency structure from masked language modeling. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. Association for Computational Linguistics, 2021. 7196–7209.

(校对责编: 孙君艳)