山西大学学报(自然科学版)47(3):518—527,2024
Journal of Shanxi University(Nat. Sci. Ed.)

DOI:10.13451/j.sxu.ns.2024003

SSHGCN:基于音形异构图卷积的中文纠错方法

任俊[1,2,3],黄瑞章[1,2,3*]

(1.贵州大学 文本计算与认知智能教育部工程研究中心,贵州 贵阳 550025;
2.贵州大学 公共大数据国家重点实验室,贵州 贵阳 550025;
3.贵州大学 计算机科学与技术学院,贵州 贵阳 550025)

摘 要:中文拼写纠错旨在检测和纠正中文文本的拼写错误,现有方法已尝试将字符相似性建模成图结构信息。但目前方法的图结构忽略汉字之间的深层音近关系,并缺少充分发挥字音和字形作用的多模态信息融合方法。因此,本文根据汉字的声母韵母信息和拼音的重要度得到拼音相似关系,结合汉字形近关系来构建汉字相似拼音-形近异构图。在该图上使用异构图卷积来互补使用汉字的音形信息,充分融合汉字的声韵和形状信息。该方法在SIGHAN15(Special Interest Group on Chinese Language Processing 15)基准上句子纠正级的$F1$值超过所有的对比方法,并在SIGHAN13基准上媲美最优的对比方法,验证了该方法的有效性。

关键词:中文拼写纠错;多模态信息融合方法;字符相似性;拼音相似关系
中图分类号:TP391 **文献标志码**:A **文章编号**:0253-2395(2024)03-0518-10

SSHGCN: A Chinese Error Correction Method Based on Heterogeneous Graph Convolution with Phonological and Visual Features

REN Jun[1,2,3], HUANG Ruizhang[1,2,3*]

(1. *Text Computing and Cognitive Intelligence Engineering Research Center of the Ministry of Education, Guizhou University, Guiyang* 550025*, China;*
2. *State Key Laboratory of Public Big Data, Guizhou University, Guiyang* 550025*, China;*
3. *College of Computer Science and Technology, Guizhou University, Guiyang* 550025*, China*)

Abstract: Chinese spelling correction aims to detect and correct spelling errors in Chinese text. Existing methods have attempted to model character similarity as graph structure information. However, the graph structure of current methods ignores the deep phonetic proximity among Chinese characters and lacks a multimodal information fusion method that fully exploits the role of character sound and shape. Therefore, this paper obtains the phonetic similarity relationship based on the initial and final information of Chinese characters and the importance of pinyin, and combines the shape proximity relationship of Chinese characters to construct a Chinese character similar pinyin-shape proximity heterogeneous graph. The heterogeneous graph convolution is used on this graph to complement the use of the sound and shape information of Chinese characters, and fully integrate the tone and shape information of Chinese characters. This method surpasses all comparison methods in terms of sentence-level $F1$ score on the SIGHAN15 benchmark, and is comparable to the best comparison method on the SIGHAN13 benchmark, verifying the effectiveness of this method.

Key words: Chinese spelling correction; multimodal information fusion method; character similarity; pinyin similarity

收稿日期:2023-12-15;**接受日期**:2024-01-12
基金项目:国家自然科学基金(62066007);贵州省科技支撑计划项目(2022277)
作者简介:任俊(1999-),男,浙江台州人,硕士,研究方向为中文文本纠错。E-mail:384089656@qq.com
* **通信作者**:黄瑞章(HUANG Ruizhang),E-mail:rzhuang@gzu.edu.cn
引文格式:任俊,黄瑞章.SSHGCN:基于音形异构图卷积的中文纠错方法[J].山西大学学报(自然科学版),2024,47(3):518-527. DOI:10.13451/j.sxu.ns.2024003

0 引言

中文拼写纠错是一项旨在自动检测和纠正文本拼写错误的基础性工作。对许多实际应用具有重要意义,如搜索引擎、光学字符识别和自动语音识别。根据Liu等[1]的研究,发音和字形上的相似字符是汉语文本错误的主要因素,具体例子如表1所示。因此有效地捕捉相似汉字之间的关联,训练模型学习引发字符误用的错误模式,可以提高模型错误检测和纠正能力。

最近的研究将汉字视作具有关联性的节点,并尝试使用图卷积网络融合汉字的特征信息。然而现有的方法大致存在以下两种问题:首先,在构建图结构时忽略拼音之间更广泛的音韵关联,缺乏对拼音以及其组件的利用,限制模型在拼音信息上的建模能力。其次,当前的方法使用图卷积网络融合读音信息和字形信息时没能关注不同模态信息之间的差异,直接将字符的不同特征进行简单的融合并输入到模型中,导致无法充分发挥字符信息的作用。

表1 存在音近类型和形近类型拼写错误的语句示例

Table 1 Examples of sentences with phonetic and orthographic spelling errors

错误类型	错误语句以及修正
音近错误	小毛想下个周末跟你们以起玩儿。(以→一)
形近错误	为了祟高理想而奋斗。(祟→崇)
注:其中的错误位置由下划线标出,"→"的左边是该句的错别字,右边是正确字。第一个例句中"以(yi3)"和"一(yi4)"拥有相同的声母韵母,只是音调不同而引起误用。而第二个例句的拼写错误是由于"祟"与"崇"字之间拥有相似的结构。	

基于上述原因,本文结合ChineseBERT(Chinese Pretraining Enhanced by Glyph and Pinyin Information)预训练语言模型,提出一种融合拼音相似关系的图卷积网络SSHGCN(Similar-Pinyin-ShapeNearness Heterogeneous Graph Convolution Network for CSC)。所使用异构图结构的重点在于引入拼音相似关系,这种关系能够有效地捕捉汉字之间在发音上的相似性。在表2中使用ChineseBERT和SSHGCN在文本纠错任务上进行对比,说明本文提出的方法的优势。虽然ChineseBERT[2]是在BERT[3]基础上融合拼音、字形特征的预训练语言模型,依然无法区分"跑(pao3)"和"饱(bao3)"这两个汉字在发音上有相似的声母,并且在字形上也只有细微的差异,导致模型难以识别句子中的错误。本文提出的方法首先基于DIMSIM[4]算法将拼音的声母、韵母以及声调转化为语音距离。然后利用拼音距离得到每个拼音的局部可达密度,以判定拼音的活跃程度。接着为相对活跃的拼音分配更多数量的连接,从而形成拼音相似关系图。在异构图结构中将汉字与其对应的拼音节点连接,并且其中汉字的边是依据开源的形近混淆集构造。使用图卷积网络聚合相邻节点的形近信息,结合拼音相似关系为每个字符生成一个向量表示。这些表示将被构建成一个字符分类器,用于ChineseBERT从上下文提取的语义表示。该字符分类器充分利用多层次的相似性信息并使用ChineseBERT拥有的拼音嵌入层和字符嵌入层,使其能够更好地结合图卷积中的拼音向量表示和字符向量表示。本文在SIGHAN15(Special Interest Group on Chinese Language Processing 15)测试集上进行实验,模型在句子级别的纠错指标$F1$性能达到了0.796,超过了所有对比的基线模型。

表2 ChineseBERT与SSHGCN两个方法对同一个存在拼写错误的输入语句进行纠正

Table 2 ChineseBERT and SSHGCN correct the same input sentence with spelling errors

输入语句	他睡很跑,睡到忘了时间起床。
正确纠正	跑(pao3)→饱(bao3)
ChineseBERT纠正结果	他睡很跑,睡到忘了时间起床。跑(pao3)
SSHGCN纠正结果	他睡很饱,睡到忘了时间起床。饱(bao3)

1 相关研究

中文拼写纠错任务的目标在于发现并纠正中文句子中的错误字符。早期的文本纠错研究主要依赖于n-gram语言模型、基于规则的技术,以及混淆字符集,用于进行字符错误的检测和修正。然而这些方法通常无法有效地对语义上下文等关键信息进行建模。随着端到端网络的迅速进步,相关研究逐渐引入了有监督的语言模型。这些端到端的方法允许模型从大量标注数据中学习,模型能够更好地捕捉语义上下文、语法规则以及字符之间的复杂关系,从而实现更准确的错误检测和纠正。2019年,预

训练语言模型已经在序列标注任务中取得重大进展。Hong等[5]提出了一种解决中文拼写错误的新范式,使用以BERT为基础的编码器DAE(Denoising Autoencoder)生成候选集。在选择候选字时,模型不再仅依赖于一个固定的概率阈值,而是使用解码器CSD结合生成概率和候选字与原始字在音形上的相似度,动态地调整阈值,在SIGHAN15测试集上取得了较好的效果。Wang等[6]提出一种序列到序列的模型,通过指针网络直接从混淆集中复制相似的字符到纠正后的新句子。这样可以有效地缩小候选字符的范围,但同时对于混淆集以外的错误,模型丧失识别这些错误的能力。2020年,Zhang等[7]采用软掩蔽机制将错误检测网络和校正网络相连,其中检测网络和纠错网络分别是基于GRU(Gate Recurrent Unit)和BERT实现的。经过检测网络后,通过软指针掩蔽潜在的错误字符嵌入。检错-纠错的二阶段模型可以对每个字符位置是否存在错误进行充分检测,减轻错误字符带来上下文信息扰动。然而掩蔽错误字符的策略可能会丢失字符音形信息,过于依赖纠错网络。2021年,Wang等[8]尝试将动态连接网络用于文本纠错,通过构建相邻字符之间的依赖关系,因此模型能够得到更加连贯的语句。2022年,Li等[9]引入基于条件随机场的分词方法WSpeller,为拼写纠错模型提供词汇边界信息,纠正词分割不当导致的拼写错误。汉字的音形特点是发生字符误用的重要因素,而上述工作的重点不在字符音形特征的整合方式,这导致它们对拼写错误的校正效果有限。

最近的一些方法也意识到有效地融合字符不同模态的特征信息是提升模型纠错能力的关键。2020年,SpellGCN[10]通过图卷积网络GCN建模字符音形相似关系,从图结构中得到更加丰富的字符特征表示。2021年,Liu等[11]使用GRU编码拼音序列、汉语笔画序列,并且采用基于混淆集的掩蔽策略得到预训练模型PLOME。模型能够在预训练过程中共同学习语义特征和音形相似关系,证明在预训练过程中学习纠错任务知识的有效性。但是上述方法可能会忽略字符特征在音形之间的重要性或相关性,因此模型的纠错能力仍有提升空间。并且在纠错任务中缺乏方法来使用声母韵母等拼音组件建立拼音之间的相似关系,模型难以理解更加深层的字符关系。

2 模型

本文将中文拼写纠错视为序列标注任务,其目标是将长度为n的输入序列$X=(x_1,x_2,x_3,\cdots,x_n)$映射为相同长度的输出序列,其中输出序列$Y=(y_1,y_2,y_3,\cdots,y_n)$应当尽可能不包含错误的字符。本文模型框架由三部分组成,分别是异构图构建模块、异构图卷积模块和语义-字符特征融合模块,其中异构图图卷积模块和语义-字符特征融合模块分别如图1的左部分和图1的右部分所示。

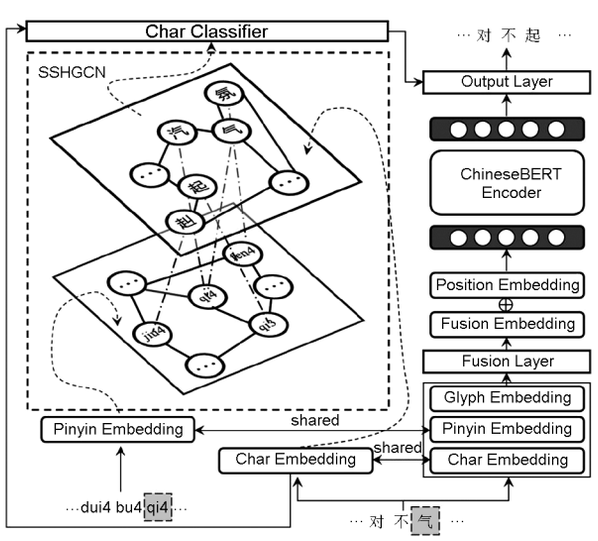

图1 SSHGCN模型结构

Fig 1 Model structure of SSHGCN

为了得到全面的拼音相似关系,异构图构建模块使用基于音韵距离累加的DIMSIM方法取得拼音距离。将得到的拼音矩阵用于计算每个拼音的局部可达密度[12],并在此基础上构建拼音节点之间的连接边。局部可达密度算法不仅可以根据拼音的语音距离构建拼音相似关系,还能够判断某个拼音与其他拼音之间的紧密程度。通过为活跃的拼音节点分配更多的边连接,从而更精准地反映拼音之间的关联程度。同时利用公开的混淆字符集,建立汉字的字形相似关系。而后基于汉字和拼音之间的对应关系构建字形特征和深层拼音特征相互作用

的图结构。

图1中包含模型另外的两个部分,从左至右分别为图卷积字符分类模块和语义-字符特征融合模块。图神经网络字符分类模块旨在建立相似字符的映射关系,通过图卷积操作聚合字符的字形相似关系和拼音特征,并在字音和字形的相互作用中捕捉字符相似性。语义-字符特征融合模块经过语言模型编码得到包含语义信息的字符向量,并且结合字符分类模块所提供的额外特征信息生成融合字符相似性和语义信息的字符表示,最后解析得到最佳纠正选项。

2.1 异构图构建模块

中文拼写错误一般是由于字符在语义、读音或字形的相似性引起。语义相似引起的字符误用通常依靠语言模型的能力解决,本文同样利用Transformer[13]编码器来提取输入语句中的上下文信息形成语义向量。本文针对读音错误基于DIMSIM算法构造拼音相似图,其中只将拼音作为相似图的节点。拼音是汉字发音的描述序列,由声母、韵母以及音调三个语音部分组成,能够提供关于字符发音相似性的重要信息。因此构建拼音相似关系的方式直接影响到模型在字音错误上的纠错效果。同时将拼音特征映射到高维特征,并在纠错任务中加以使用已经成为文本纠错任务的常用方法。但是DIMSIM算法并不直接将整个拼音进行编码,而是利用声母和韵母在高维空间中的距离得到拼音之间相似度。拼音 c 和拼音 c' 之间的语音距离用如下公式表示:

$$S(c,c')=S_p\left(p_c^I,p_{c'}^I\right)+S_p\left(p_c^F,p_{c'}^F\right)+S_t\left(p_c^T,p_{c'}^T\right),\quad(1)$$

其中 p^I, p^F, p^T 分别表示的是拼音的声母,韵母和声调。S_p 表示拼音 c 和拼音 c' 之间的欧式距离,S_t 表示拼音 c 和拼音 c' 之间的音调距离。音调有五种,依次是阴平、阳平、上声、去声和轻声,并且它们的音调值也是按照该次序给出,分别为1,2,3,4,5。音调距离的计算以音调之间的差值得到。DIMSIM算法使用拼音组件计算相似度,对汉字拼音在声母和韵母之间的差异上非常敏感,更加符合实际的应用情况。

通过DIMSIM算法可以得到词汇表中所有汉字拼音之间的语音距离。接着将拼音距离矩阵作为局部可达密度算法的输入,该算法会基于拼音局部可达距离为每个拼音节点选择邻居节点的数量。在拼音距离关系中,部分拼音显然由于具有更加相似的发音而聚集产生类簇效应。同样也会有部分拼音不具备多数拼音的共性成为拼音数据中的离散点。在计算拼音 c 的局部可达密度之前,需要得到拼音 c 关于拼音 c' 的可达距离,公式如下:

$$d_\alpha^r(c,c')=\max\{d^\alpha(c'),d(c,c')\},\quad(2)$$

其中 $d^\alpha(c')$ 是离拼音 c 最近的 α 个拼音的最远距离,并且超参数 α 决定了判断一个拼音是否活跃的严格程度。$d(c,c')$ 是拼音 c 关于拼音 c' 的实际语音距离。局部可达距离越大,意味着该拼音所在的局部区域越密集,越具备拼音之间的共性。而局部可达密度的公式如下:

$$\rho_\alpha(c)=\frac{\left|N_\alpha(c)\right|}{\sum_{c'\in N_\alpha(c)}d_\alpha^r(c,c')},\quad(3)$$

其中 $N_\alpha(c)$ 代表拼音 c 的距离小于等于 $d^\alpha(c')$ 的所有点的集合。拼音的局部可达密度越小,拼音越容易成为离群点,因此会分配相对更少的相似拼音作为它的邻居节点。拼音节点 c 分配到的邻居节点数量的公式如下:

$$m_c=\left\lfloor\frac{\rho_\alpha(c)}{\rho_\alpha^s}\times\beta\right\rfloor,\quad(4)$$

其中 β 是控制拼音图总边数的超参数,ρ_α^s 由以下公式可得:

$$\rho_\alpha^s=\sum_{c\in R_N}\rho_\alpha(c)。\quad(5)$$

得到每个拼音节点的邻居节点数量之后,构造一个包含所有拼音的拼音相似图 A,为拼音 c 选择距离最近的 m_c 个节点作为它的相似节点。在拼音相似图 A 中,第 i 个拼音和第 j 个拼音之间的边表示拼音 i 和拼音 j 是否相似。

对于字形错误,本文直接参考SpellGCN的形近图的构造方式,将形近混淆集中汉字作为形近图的节点并连接混淆字符作为汉字节点的边关系。最后根据汉字与拼音之间的对应关系将字节点与拼音图的拼音节点相连,获得多层次的相似拼音-形近字异构图。

2.2 图卷积字符分类模块

本文构建基于相似拼音-形近字异构图的中文文本纠错模型SSHGCN,以学习汉字节点

的特征表示。模型通过基于异构图融合额外的拼音特征,自动学习音近字符的映射关系。同时会对具有相同拼音的字符进行信息传递,更有效地利用字符之间的深层联系。采用SSHGCN的另外一个优势是可以将读音特征和字形相似信息在图结构的不同层次进行交互,积累相邻节点中不同类型节点的信息,使字形和字音关系能进一步得到融合。最终形成的字符分类器将会为语言模型得到的语义向量,提供字符发音和字符形状上的特征。经过语言模型和字符分类器的字符向量,融合多模态信息,反映汉字之间的复杂关系。

图卷积网络利用图结构中节点之间的连接关系,通过多层图卷积操作逐渐学习得到节点的高级表示。为结合图结构中附加的拼音关系,本文采用包含拼音嵌入层的ChineseBERT预训练模型作为主干模型,并且使用它的字符嵌入和拼音嵌入作为SSHGCN的初始节点特征。字符嵌入$\boldsymbol{H}_w^0$和拼音嵌入$\boldsymbol{H}_p^0$的维度分别为$M\times D$和$N\times D$,其中M表示图结构中所有汉字节点的数量,N表示所有拼音的数量,D表示节点嵌入的维度(字符向量的嵌入维度与拼音向量的嵌入维度是相同的)。下列公式表示每层图卷积在图结构中节点类型为r的一次传播:

$$\boldsymbol{C}_r^{l+1}=\sigma\Big(\sum_{s\in R}\sum_{j\in N_r^s}\boldsymbol{A}_{r,s}^j\boldsymbol{H}_s^l\boldsymbol{W}^j\Big),\tag{6}$$

其中$\boldsymbol{C}_r^l$表示第l层中节点类型为r的字符嵌入,s表示源节点类型,N表示源节点和目标节点类型分别为s和r的边集合,$\boldsymbol{A}_{r,s}^j$表示边类型为j且目标节点类型为r、源节点类型为s的邻接矩阵,$\boldsymbol{W}^j$表示边类型为j的权重矩阵,σ表示ReLU激活函数。采用的图卷积将不同类型的节点连接并应用参数化的线性变换,自适应地学习不同关系对字符特征的作用。

在每一层图卷积之后,使用全局自注意力机制增强汉字节点的信息交互和表示能力得到更新后的特征表示,并且对拼音节点进行同样的操作。这样做的目的是将图结构聚合的局部信息放入全局进行交互。对于所有类型的节点允许每个节点通过计算相同类型的节点之间的注意权重,这相当于将来自其他非相邻节点的信息进行聚合。在加权平均的过程考虑了每个节点的重要性,突破局部邻居节点的限制,从而更好地捕捉字符相似的模式。公式如下:

$$\hat{\boldsymbol{C}}_r^l=\mathrm{softmax}\Big(\frac{\boldsymbol{C}_r^l(\boldsymbol{C}_r^l)^T}{\sqrt{D}}\Big)\boldsymbol{C}_r^l。\tag{7}$$

通过对先是节点维度的平方根进行缩放注意力分数,防止分数过大或过小导致梯度消失或爆炸,然后作为softmax函数的输入。并且将每一列的值归一化为概率分布,使得每个节点的权重都在0到1之间,并确保每个节点与其他节点的关联权重总和为1,从而使得每个值的加权汇聚更加合理。

由于图卷积的初始节点特征来自于ChineseBERT的嵌入层,模型有必要保留原有的信息。为了保留提取器的原始语义,将之前各层的输出累积作为最终输出。累加操作公式如下:

$$\boldsymbol{H}_r^{l+1}=\hat{\boldsymbol{C}}_r^l+\sum_{i=0}^{l}\boldsymbol{H}_r^l。\tag{8}$$

本模块在相似拼音-形近字异构图的基础上设计图卷积操作,因此应考虑拼音节点和汉字节点以及各类节点之间连接边的数量,不宜使用过于复杂的模型结构。

2.3 语义-字符特征融合模块

本文采用ChineseBERT预训练模型将输入字符序列$X=(x_1,x_2,x_3,\cdots,x_n)$转换成已经初步融合字符特征的语义向量$\boldsymbol{V}=(\boldsymbol{v}_1,\boldsymbol{v}_2,\boldsymbol{v}_3,\cdots,\boldsymbol{v}_n)$。ChineseBERT将字符嵌入、字形嵌入和拼音嵌入串联并融合成综合的特征向量。将融合嵌入、位置嵌入和片段嵌入相加,形成编码器的输入。其中编码器由12层Transformer-Encoder组成,每一层都运用自注意力机制和前馈神经网络来对输入序列进行上下文建模。经过编码器的处理之后,使用Transformer-Encoder最后一层的输出作为字符x_i的表示向量$\boldsymbol{v}_i$。在预测时使用权值为$\boldsymbol{W}\in\mathbb{R}^{O\times D}$的线性层进行解码并经过softmax函数获得最终的目标字符,公式如下:

$$P(y_i|X)=\mathrm{softmax}(\boldsymbol{W}\boldsymbol{v}_i),\tag{9}$$

由于词表中存在一定数量的生僻字,构建包含词表所有汉字的图结构会导致模型计算量增加。并且这些节点的连接关系往往是稀疏的,

加入过多的字节点会引入冗余信息,无法发挥模型应有的能力。因此模型中图结构的字节点集合只是词汇表的子集,矩阵 $\boldsymbol{W}$ 的值由图卷积分类模块的输出和 ChineseBERT 的字符嵌入层共同决定,具体公式如下:

$$\boldsymbol{W}_i=\begin{cases}\boldsymbol{H}^{L}_{w,u}, & \text{如果第}i\text{个字符属于图结构字节点,}\\ \boldsymbol{E}_i, & \text{其他,}\end{cases} \tag{10}$$

其中 $u_i\in\{1,\cdots,M\}$ 为 ChineseBERT 预训练模型词汇表中第 i 个图结构词表的索引,$\boldsymbol{E}\in\mathbb{R}_{O\times D}$ 为预训练模型的字符嵌入层。对于给定的序列 X,模型的训练目标是最小化对数似然函数,公式如下:

$$L^c=-\sum_{i=1}^{n}\log P(y_i|X)。 \tag{11}$$

3 实验

3.1 数据集

本文的训练集是由 SIGHAN 训练集(包含 10 k 个样本)和 Wang 等[14]伪造生成的训练集(包含 271 k 个样本)合并组成,其中 SIGHAN 训练集包含三个训练集。同时,使用 SIGHAN13[15]、SIGHAN15[16]测试集作为基准实验的测试数据,评估本文方法的性能。另外参考普遍性的做法,训练集和测试集使用开源工具 OpenCC 将这些数据集中的字符从繁体中文转换为简体中文。关于数据集的统计结果如表 3。

3.2 评价指标

本文采用的中文拼写纠错的性能指标为整句级别在检测级别和纠正级别上的准确率 Precision(P)、召回率 Recall(R)和加权调和平均值 $F1$-Score($F1$)。

表 3 数据集统计信息

Table 3 Statistical information of datasets

本文模型训练集	数据量/条	平均长度	错误个数
Wang 伪造训练集	271 329	44.4	382 704
SIGHAN13 训练集	350	49.2	350
SIGHAN14[17]训练集	6 526	49.7	10 087
SIGHAN15 训练集	3 174	30.0	4 237
本文模型测试集	数据量/条	平均长度	错误个数
SIGHAN13 测试集	1 000	74.3	1 224
SIGHAN15 测试集	1 100	30.5	550

注:SIGHAN13 的训练集和测试集均源自中文学习者的作文。在这些作文中,精选部分句子构成测试集用于评估模型的预测性能。同样地,SIGHAN15 的训练集和测试集也采用了相同的方法。

整句级别　该级别考虑的是模型在整句上的纠错能力。只有当句子中所有的字符被正确预测为正例或负例,才会被视为正确的预测。

检测级别　该级别要求模型能够正确地标识出句子中哪些位置存在错误,但不要求模型能够纠正这些错误。

纠正级别　该级别是指模型能够正确地找到存在错误的位置,并在此基础上将其更正为正确的内容。

准确率 Precision(P)、召回率 Recall(R)和加权调和平均值 $F1$-Score($F1$)的计算公式如下:

$$P=\frac{TP}{TP+FP}, \tag{12}$$

$$R=\frac{TP}{TP+FN}, \tag{13}$$

$$F_1=\frac{2\times P\times R}{P+R}, \tag{14}$$

其中 TP 代表真正例(True Positive),FP 代表假正例(False Positive),TN 代表真负例(True Negative),FN 代表假负例(False Negative)。高准确率意味模型纠错时不会引入新的错误,而召回率展示模型识别错误字符的能力。其中 $F1$-Score 是最为重要的评价指标,用于平衡模型的精确性和全面性。

3.3 实验设置

本文使用 Pytorch 提供的深度学习框架,并在 NVIDIA RTX A6000 上进行实验。ChineseBERT 编码器使用开源的预训练权重进行初始化,而模型其他部分则进行随机初始化。本文模型实验都是在三个 SIGHAN 训练集和 271 k 条的人工构造数据集上进行微调。SIGHAN13、SIGHAN15 测试集取得最优模型参数时学习率为 5×10^{-5}。批处理大小分别设置为 96 和 64,训练轮次设置为 45。其他参数设置如表 4。

3.4 对比实验

表 5 展示 ChineseBERT 以及其他 LLMs(Large Language Models)大型语言模型在 SIGHAN15 测试集上的纠错表现。从表 5 实验

表4 实验相关超参数

Table 4 Experiment related hyperparameters

参数	取值
固定随机种子	2 333
Transformer层数	12
优化器	AdamW
Dropout_rate	0.1
最大序列长度	192
Weight-decay	0.01
局部可达密度(α)	3
局部可达密度(β)	10 000

结果看,T5[18]在预训练时要对整个句子进行预测,因此能比其他语言模型提供更强的纠错能力。而通过结合字形和拼音信息ChineseBERT和PLOME能够在微调之后更好适应中文拼写纠错任务。

表5 LLMs性能对比

Table 5 Performance comparison of LLMs

阶段	方法	Sentence-Correction-Level		
		Precision	Recall	$F1$
预训练	T5	52.2	39.4	44.9
	ChineseBERT	32.1	42.0	36.4
	PLOME	34.2	38.9	36.4
微调	BERT	66.0	74.9	70.2
	T5	83.2	63.9	73.5
	ChineseBERT	75.4	77.4	76.4
	PLOME	75.3	79.3	77.2

本文选择近年在中文拼写纠错任务方面取得杰出进展的若干方法进行对比。下面是具体的基线方法:

SpellGCN 此模型通过图卷积网络来建模汉字之间的语音和视觉相似性知识并映射到字符分类器。最后将字符分类器应用于BERT提取的字符特征表示。

GAD(Global Attention Decoder for Chinese Spelling Error Correction)[19] 该研究提出了一种方法来捕获丰富的全局上下文信息,以减少局部错误上下文信息的影响。

DCN(Dynamic Connected Networks for Chinese Spelling Check) 该方法提出使用拼音来生成候选汉字,并使用基于注意力的网络来模拟相邻两个汉字之间的依赖关系。

WSpeller(Robust Word Segmentation for Enhancing Chinese Spelling Check) 该模型通过预测字形和语音上相似的单词进行训练,并修改嵌入层的输入以整合分词信息。

MDCSpell(A Multi-task Detector-Corrector Framework for Chinese Spelling Correction)[20]使用BERT作为纠正器来捕获字符的视觉和语音特征,并将检测器的隐藏状态集成到纠正器中,以减少错误的影响。

本文的检错级指标和纠错级指标均采用句子级别的评价指标。

本文所有基线模型的实验参数均使用作者提供的数据。在SIGHAN13、SIGHAN15测试集上的对比结果如表6所示。其中的重点是模型在检测级别和纠正级别的$F1$值,因为$F1$值反应模型整体的性能。可以看出SSHGCN在SIGHAN15测试集的$F1$值取得显著的领先,并且在SIGHAN13测试集的表现媲美WSpeller,这表明SSHGCN的可行性。

另外可以发现本文模型在检测和纠正级别上的召回率与基线模型有着较大的领先,只在SIGHAN13测试集上低于WSpeller,这也验证图卷积中多层结构和更多的拼音相似信息能够有效提升模型对于错误的敏感程度。在准确率上本模型与MDCSpell差距不大,而WSpeller在召回率上取得提升的原因可能是其在融入字符发音和形态的基础上还使用分词方法,使得模型拥有更多不同特征的信息。本模型在不牺牲准确率的前提下,极大地提升在中文纠错任务上召回率,这是本模型的性能超过基线模型的原因。

3.5 对比实验

为了深入分析本文方法的有效性,逐步变化模型的不同组件并进行一系列的消融实验,准确评估每个组件对整体性能的贡献,并验证本文方法在不同条件下的性能。

首先为了研究SSHGCN中拼音相似关系和形近字关系对于本模型的影响,设计一组图卷积消融实验,以此探究该模块的多层次的不同特征在图卷积中的贡献。该消融实验的具体设计:

a)移除关系图卷积,仅使用ChineseBERT;

b)移除关系图卷积,改用SpellGCN模型的音形近图卷积模块并使用ChineseBERT;

表6 模型性能对比

Table 6 Comparison of performance among different model

测试集	模型(发表年份)	Detection-Level			Correction-Level		
		Precision	Recall	*F*1	Precision	Recall	*F*1
SIGHAN13测试集	SpellGCN(2020)	80.1	74.4	77.2	78.3	72.7	75.4
	GAD(2021)	85.7	79.5	82.5	84.9	78.7	81.6
	DCN(2021)	86.8	79.6	83.0	84.7	77.7	81.0
	WSpeller(2022)	82.3	**86.9**	**84.6**	81.2	**85.7**	**83.4**
	MDCSpell(2022)	89.1	78.3	83.4	**87.5**	76.8	81.8
	ours	87.1	81.9	84.5	85.9	80.7	83.3
SIGHAN15测试集	SpellGCN(2020)	74.8	80.7	77.7	72.1	77.7	75.9
	GAD(2021)	75.6	80.4	77.9	73.2	77.8	75.4
	DCN(2021)	77.1	80.9	79.0	74.5	78.2	76.3
	WSpeller(2022)	**81.9**	78.0	79.9	**79.9**	76.1	77.9
	MDCSpell(2022)	80.8	80.6	80.7	78.4	78.2	78.3
	ours	80.3	**83.1**	**81.7**	78.2	**81.0**	**79.6**

c)移除SSHGCN中形近结构,使用仅包含音近图的异构图卷积和ChineseBERT;

d)移除SSHGCN中音近结构,使用仅包含形近图的异构图卷积和ChineseBERT;

e)音近结构和形近结构的SSHGCN。

在图结构消融实验时,使用错误分布相对更为合理的SIGHAN15测试集,该数据集的细节可见3.1节以及表3。实验结果如表7所示,本文设计的SSHGCN在单独只使用音近图或形近图的情况下已经能够显著提升纠错能力。特别是在只使用音近图的情况下召回率就已经超过融合拼音和字形信息的SpellGCN方法,说明SSHGCN在纠正音近错误的有效性。在使用SSHGCN将音近和形近关系进行结合之后,可以在音近图卷积的基础上进一步提升模型的纠错能力,验证模型在融合多模态信息的优势。

本文还可以设计对局部可达密度部分的参数敏感性分析,研究局部可达密度对于拼写纠错任务的影响。局部可达密度的参数实验同样使用SIGHAN15测试集,其结果如表8所示。通过局部可达密度算法引入超参数α和β,其中α控制拼音的可达距离,β控制拼音图中边的总数。适当α值能够准确地反映拼音节点之间的密度关系,支持后续的纠错任务。过大的β值会导致拼音图中的边过多,模型难以捕捉细粒度的拼音特征。使用过小的β值会让图的连通性变差,导致信息无法有效传播。

4 结论

本文提出了一种拼音相似性关系和形近字关系结合的异构图卷积网络,专用于解决中文拼写纠错问题。这种模型通过DIMSIM算法和局部可达密度捕捉拼音之间的相似性关系,构建了一个包含拼音关联图的基础图结构。根据拼音与汉字的对应关系将拼音关系中拼音节点与形近图的汉字节点进行连接,形成了相似汉字在读音和形状上的交叉关联图,为文本纠错任务提供了更深层次的视角分析字音和字形在

表7 各模型图结构消融实验对比

Table 7 Comparison of graph structure ablation experiments among different models

方法	Detection-Level			Correction-Level		
	Precision	Recall	*F*1	Precision	Recall	*F*1
ChineseBERT	77.5	79.7	78.6	75.4	77.4	76.4
ChineseBERT+SpellGCN(保留音近和形近)	77.2	81.5	79.3	75.3	79.5	77.4
SSHGCN(仅使用音近图卷积)	77.0	81.6	79.3	74.7	80.2	77.2
SSHGCN(仅使用形近图卷积)	76.1	81.1	78.6	74.3	79.3	76.8
SSHGCN	80.3	83.1	81.7	78.2	81.0	79.6

表8 超参数(α,β)对纠错能力的影响

Table 8 Impact of hyperparameters (α, β) on error correction ability

局部可达密度参数设置	Detection-Level			Correction-Level		
	Precision	Recall	$F1$	Precision	Recall	$F1$
局部可达密度(α=2,β=10 000)	78.0	82.2	80.1	76.5	80.5	78.5
局部可达密度(α=3,β=10 000)	80.3	83.1	81.7	78.2	81.0	79.6
局部可达密度(α=4,β=10 000)	78.6	81.8	80.2	77.0	80.2	78.6
局部可达密度(α=5,β=10 000)	80.8	80.6	80.7	78.4	78.2	78.3
局部可达密度(α=6,β=10 000)	77.0	83.2	80.1	74.8	81.2	78.0
局部可达密度(α=7,β=10 000)	77.9	82.3	80.1	75.8	80.0	77.9
局部可达密度(α=8,β=10 000)	77.3	81.3	79.3	74.9	78.7	76.8
局部可达密度(α=3,β=8000)	74.9	79.9	77.4	73.2	78.0	75.6
局部可达密度(α=3,β=10 000)	80.3	83.1	81.7	78.2	81.0	79.6
局部可达密度(α=3,β=15 000)	78.4	83.0	80.7	76.9	81.3	79.1
局部可达密度(α=3,β=20 000)	78.1	83.2	80.6	76.7	81.7	79.2
局部可达密度(α=3,β=25 000)	78.8	82.4	80.6	77.0	80.6	78.8
局部可达密度(α=3,β=30 000)	78.3	82.6	80.4	75.6	79.8	77.7
局部可达密度(α=3,β=35 000)	76.9	81.9	79.4	75.2	80.0	77.6

错误模式上复杂的相互作用。实验结果表明,本方法可以显著提升语言模型的纠错能力。

鉴于传统图卷积网络在每层中需要频繁地进行矩阵乘法操作,这不可避免地导致了显著的计算负担。尤其在处理大规模图数据时,随着节点数量的增加以及节点邻接关系的不断增加,计算成本愈发显著上升。同时汉字的字符集合形成庞大的字库,进一步提升使用图卷积建模汉字字符相似性的难度。因此,未来的研究方向可能会聚焦在如何更有效地将大规模图数据应用于中文拼写纠错任务中。这将涵盖如何处理更加庞大的字节点集合和如何引入更为复杂的字符关系,以进一步提升模型在中文拼写纠错任务中的性能和泛化能力。

参考文献:

[1] LIU C L, LAI M H, CHUANG Y H, *et al*. Visually and Phonologically Similar Characters in Incorrect Simplified Chinese Words[J]. *ACM Transactions on Asian La Inf Process*, 2011, **10**(2): 10, DOI: 10.1145/1967293.1967297.

[2] SUN Z J, LI X Y, SUN X F, *et al*. ChineseBERT: Chinese Pretraining Enhanced by Glyph and Pinyin Information[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Stroudsburg, PA, USA: Association for Computational Linguistics, 2021: 2065-2075. DOI: 10.18653/v1/2021.acl-long.161.

[3] DEVLIN J, CHANG M W, LEE K, *et al*. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[C]//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, Minnesota: Association for Computational Linguistics, 2019: 4171-4186. DOI: 10.18653/V1/N19-1423.

[4] LI M, DANILEVSKY M, NOEMAN S, *et al*. DIMSIM: an Accurate Chinese Phonetic Similarity Algorithm Based on Learned High Dimensional Encoding[C]//Proceedings of the 22nd Conference on Computational Natural Language Learning. Brussels, Belgium:Association for Computational Linguistics, 2018: 444-453. DOI: 10.18653/v1/K18-1043.

[5] HONG Y Z, YU X G, HE N, *et al*. FASPell: a Fast, Adaptable, Simple, Powerful Chinese Spell Checker Based on DAE-decoder Paradigm[C]//Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019). Stroudsburg, PA, USA: Association for Computational Linguistics, 2019: 160-169. DOI: 10.18653/v1/d19-5522.

[6] WANG D M, TAY Y, ZHONG L. Confusionset-guided Pointer Networks for Chinese Spelling Check[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: Association for Computational Linguistics, 2019: 5780-5785. DOI: 10.18653/v1/p19-1578.

[7] ZHANG S H, HUANG H R, LIU J C, *et al*. Spelling Error Correction with Soft-masked BERT[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: Associa-

tion for Computational Linguistics, 2020: 882−890. DOI: 10.18653/v1/2020.acl-main.82.

[8] WANG B X, CHE W X, WU D Y, *et al*. Dynamic Connected Networks for Chinese Spelling Check[C]//Findings of the Association for Computational Linguistics: ACL−IJCNLP 2021. Stroudsburg, PA, USA: Association for Computational Linguistics, 2021: 2437−2446. DOI: 10.18653/v1/2021.findings-acl.216.

[9] LI F F, SHAN Y R, DUAN J W, *et al*. WSpeller: Robust Word Segmentation for Enhancing Chinese Spelling Check [C]//Findings of the Association for Computational Linguistics: EMNLP 2022. Stroudsburg, PA, USA: Association for Computational Linguistics, 2022: 1179−1188. DOI: 10.18653/v1/2022.findings-emnlp.84.

[10] CHENG X Y, XU W D, CHEN K L, *et al*. SpellGCN: Incorporating Phonological and Visual Similarities into Language Models for Chinese Spelling Check[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA, USA: Association for Computational Linguistics, 2020: 871−881. DOI: 10.18653/v1/2020.acl-main.81.

[11] LIU S L, YANG T, YUE T C, *et al*. PLOME: Pre-training with Misspelled Knowledge for Chinese Spelling Correction[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Online: Association for Computational Linguistics, 2021: 2991−3000. DOI: 10.18653/v1/2021.acl-long.233.

[12] BREUNIG M M, KRIEGEL H P, NG R T, *et al*. LOF: Identifying Density-based Local Outliers[C]//Proceedings of the 2000 ACM SIGMOD international conference on Management of data. New York, NY, USA:Association for Computing Machinery, 2000: 93−104. DOI: 10.1145/342009.335388.

[13] VASWANI A, SHAZEER N, PARMAR N, *et al*. Attention is all you Need[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook, NY, USA:Curran Associates, 2017: 6000−6010. DOI: 10.5555/3295222.3295349.

[14] WANG D M, SONG Y, LI J, *et al*. A Hybrid Approach to Automatic Corpus Generation for Chinese Spelling Check[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: Association for Computational Linguistics, 2018: 2517−2527. DOI: 10.18653/v1/d18-1273.

[15] WU S H, LIU C L, LEE L H. Chinese Spelling Check Evaluation at SIGHAN Bake-off 2013[C]//Proceedings of the Seventh SIGHAN Workshop on Chinese Language Processing. Brussels, Belgium: Association for Computational Linguistics, 2013: 35−42. DOI: 10.18653/v1/W13-4406.

[16] TSENG Y H, LEE L H, CHANG L P, *et al*. Introduction to SIGHAN 2015 Bake-off for Chinese Spelling Check [C]//Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing. Stroudsburg, PA, USA: Association for Computational Linguistics, 2015: 32−37. DOI: 10.18653/v1/w15-3106.

[17] YU L C, LEE L H, TSENG Y H, *et al*. Overview of SIGHAN 2014 Bake-off for Chinese Spelling Check [C]//Proceedings of The Third CIPS−SIGHAN Joint Conference on Chinese Language Processing. Stroudsburg, PA, USA: Association for Computational Linguistics, 2014: 126−132. DOI: 10.3115/v1/w14-6820.

[18] COLIN R, NOAM S, ADAM R, *et al*. Exploring the Limits of Transfer Learning with a Unified Text-to-text Transformer[J]. *J Mach Learn Res*, 2020, **21**(140): 1−67.

[19] GUO Z, NI Y, WANG K Q, *et al*. Global Attention Decoder for Chinese Spelling Error Correction[C]//Findings of the Association for Computational Linguistics: ACL−IJCNLP 2021. Stroudsburg, PA, USA: Association for Computational Linguistics, 2021: 1419−1428. DOI: 10.18653/v1/2021.findings-acl.122.

[20] ZHU C X, YING Z Q, ZHANG B Y, *et al*. MDCSpell: a Multi-task Detector-corrector Framework for Chinese Spelling Correction[C]//Findings of the Association for Computational Linguistics: ACL 2022. Stroudsburg, PA, USA: Association for Computational Linguistics, 2022: 1244−1253. DOI: 10.18653/v1/2022.findings-acl.98.