文章编号：

基于神经网络无监督藏文正字检错法

色差甲[1] 慈祯嘉措[1] 才让加[1,2] 华果才让[1]

1.青海师范大学 计算机学院，青海 西宁 810008；
2. 青海省藏文信息处理与机器翻译重点实验室，青海 西宁 810008；

摘要：在缺乏标注数据的条件下本文把藏文正字检错任务视为一个分类问题，首先从语言知识中构建音节混淆子集并给每个原句加噪，然后建立深层双向表征的 BERT 作为分类模型，最后为了证明本方法的有效性，构建两个基线模型和三种不同领域的测试集，实验结果显示本方法的结果优于两个基线模型。本方法在相同领域测试集上句子分类的正确率达到 93.74%，不同领域测试集上也能达到 83.6%。对错误音节的识别结果为 74.53%，同时对无错误音节的误判率只有 2.3%。

关键词：藏文正字检错；加噪算法；无监督； BERT 模型

中图分类号：TP391 **文献标识码：**A

Unsupervised Tibetan Character Detection Errors Based on Neural Network

SECHA Jia[1] , CIZHEN Jiacuo [1], CAIRANG Jia[1,2] and HUAGUO Cairang[1]
1. Computer College of Qinghai Normal University;
2. Tibetan intelligent information processing and Machine Translation Key Laboratory Qinghai

Abstract : Under the condition of lack of labeled data, this paper put the Tibetan character error detection task as a classification problem, first of all, build syllable confusion subset from the language knowledge and add noise to each other, and then establish a deep two-way characterization of BERT as classification model, and finally to prove the effectiveness of this method, build two baseline model and test set, in the field of three different experimental results show that this method is superior to the result of the two baseline model. The accuracy of sentence classification in this method can reach 93.74% in the same field and 83.6% in different fields. The recognition result of wrong syllables is 74.53%, while the misjudgment rate of the wrong syllables is 2.3%.

Key words: character detection errors, noise addition algorithm, unsupervised, BERT model

0 引言

随着互联网的快速发展和普及，产生了大规模的藏文电子文本，同时在这些文本中也存在着大量的字词和语法错误，亟需有效的藏文语法检测器来提高电子文本的质量。语法检错和纠正不仅是自然语言处理中一个基础性任务，而且在语音识别、文字识别、搜索引擎、Word 拼写校对以及作文自动修改等领域应用广泛[1-2]。近期对自动语法检错或纠正引起越来越多的重视，因为世界各地有大量的外语学习者需要即时准确的反馈来帮助改进他们的撰写文本能力[3-4]。

近年来，在语法检错和纠正任务中主流方法有神经序列标注模型和神经机器翻译模型等。像英语和汉语都有共享的语法错误标注数据，这些共享数据推进了语法检错及纠正的研究。尽管取得了很多进展，但标注数据的稀疏问题仍然是语法检错和纠正研究的主要限制因素。对于藏文语法检错任务而言，标注数据的缺乏直接限制了该任务的发展空间。通过人工标注来构建数据是一个非常耗力耗时的任务，并且数据质量会受到人类

收稿收稿日期：2019-08-15 **定稿日期**：2019-08-15
基金项目：国家重点研发计划项目（2017YFB1402200）；国家自然科学基金（61063033, 61662061）；国家社科基金（14BYY132）

判断力和项目支持力度等的影响。Zhouran Liu 和 Yang Liu 提出了利用未标注数据的英语语法检错方法，也就是努力探索如何从无错误的文本中自动伪造现实中出现的语法错误类型，从而获得大量的伪标注数据[5]。

在文献[5]的基础上本文对藏文正字检错任务探索了两个新的关键点：

第一是构建混淆集，由容易被混淆的音节组成的数据集，构建于前辈学者们分析好的藏文动词时态、音相似及形相似等正字法有关的著作，在文本的正字检错中起着关键作用。然后由混淆集给数据随机加噪（噪点是指无错误音节变成错误音节），这种制造方法不仅有逼真现实文本中常见错误音节的能力，而且具有更大的覆盖面；

第二是使用了深层双向表征的 BERT 模型，该模型在自然语言处理的很多任务中取得目前最佳的效果[6]。

本文中把藏文正字检错任务作为二元分类问题来解决，最后实验证明了本方法的有效性，不仅能判断整个句子是否为病句，而且也能判断句子中每个音节是否符合藏文正字法。正字错误的纠正是一项具有挑战性的任务，在无监督训练的情况下更是一项非常困难的任务。因为现实文本中会出现很多错误类型，而人造错误的自动生成方法可能不够复杂，无法覆盖所有错误类型，所以本实验中只关注了一个句子中每个音节是否符合藏文正字法的检测任务，而没有涉及到冗余音节、遗漏音节、音节序列错误及其纠正等工作。

1 相关工作

早期的语法检查工作中采用了基于词典、规则以及统计[7-8]等方法来检测和纠正特定的错误类型，如有冗余词、遗漏词、误用词以及词序错误等。随后在自然语言处理中引入深度学习方法之后，各个领域的处理结果都优于传统方法。对于语法错误检测和纠正任务来说，最常用的方法有神经序列标注模型和神经机器翻译模型[9-11]。由于英语和汉语有公开的语法检测标注数据集，所以提升了该语种的语法检测及纠正的研究成果。比如汉语的有 CSL(Chinese as Second Language) 和 NLPCC2018-GEC[11] 等，英语的公开数据有 Hoo、CoNLL(2013、 2014) [12-13]等。近期逐渐开始专注于连续句子的语法纠正工作，不再仅限于局部信息中的语法检查，也不是非母语学习者的语法错误纠正，而是着重研究于连续两句及以上句子的流畅度。

藏文文本的错误类型类似于英语文本的错误，一般分为真字错误和非真字错误两类。目前藏文对非真字错误的检测技术研究比较成熟，且提到多种方法，比如有字典匹配、藏文字的构建规则法[14-15]和 CNN 的藏文音节拼写检查[16]等。但是对于真字错误和语法错误的检测及纠正工作相对较少，因为藏文缺乏带标注的数据。针对这个问题本文利用未标注的藏文纯文本、藏文正字法以及深层双向表征的 BERT 模型来解决藏文真字错误的检测任务。

2 模型架构

2.1 基线模型

据文献查阅所知，关于藏文词级语法错误检测任务的相关研究非常少，没有相关的基线作为参照，因此本文将设定两个基线方法分别为浅层双向表征的 BiLSTM 模型和与 Attention 混合使用的 BiLSTM 模型。结构如图 1 和 2 所示：

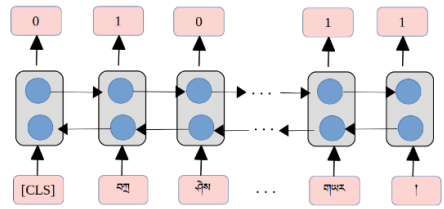

图 1 BiLSTM 的结构

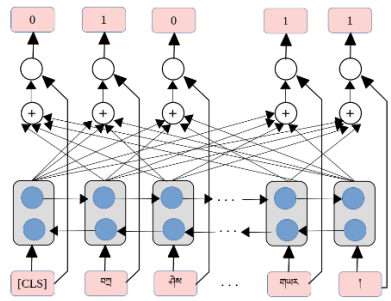

图 2 BiLSTM-Atten 的结构

图 1 中 BiLSTM 模型是一个常见的模型，图 2 的 BiLSTM-Atten 表示 BiLSTM 的基础上使用了 Attention 机制，其计算过程如下所示：

$$h_i = \left(\left[\overrightarrow{h_i};\overleftarrow{h_i}\right] \times U\right) + b_i \quad (1)$$

$$\alpha_{ij} = softmax\left(w_i \times \mathrm{W} \times h_j^T\right) \quad (2)$$

$$c_i = \sum_{j=1} \alpha_{ij} \times h_j \quad (3)$$

公式(1)中$\overrightarrow{h_i}$和$\overleftarrow{h_i}$分别是 BiLSTM 中第 i 时刻的正

向和反向的计算结果;w_i是输入句子中第i个音节的向量；[·]表示两个向量或矩阵的拼接算法。

2.2 本文模型

由于正字检错任务中对上下文的信息要求极高，所以本实验采用深层双向表征的 BERT 模型。该模型是近期广泛使用的一个模型，基本架构是 Transformer[17]的编码器部分,其结构图 3 所示：

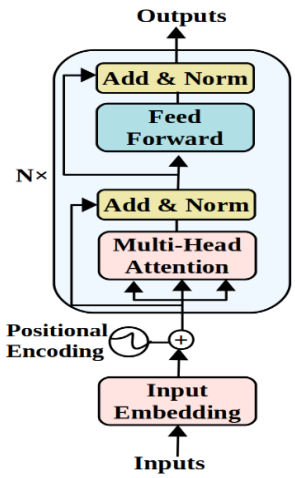

图 3 BERT 模型的结构

图 3 中 N 表示网络层数，*Add*、*Norm* 和 *Feed Forward* 分别是残差方法、参数正规化以及前馈网络。主要的组成部分 Multi-Head Attention 的计算过程如公式（4）、（5）、（6）所示：

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (4)$$

$$MultiHead(Q,K,V) = [head_1, head_2, \dots, head_h] \quad (5)$$

$$head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right) \quad (6)$$

其中 d_k为输入向量 K的维度，Q、K、V分别是 query（查询）、key（键）以及 value（值）的缩写。若其中的 Attention 使用了 Self-Attention 时，则有 Q=K=V，即每个位置上的音节都可以无视方向和距离，有机会直接和句子中的每个音节计算上下文关联程度。该编码器中恰好使用了 Self-Attention。另一个主要的组成部分 Positional Encodig(简称 *PE*)，因为序列顺序信息对模型很重要，所以 transformer 计算位置信息时使用正弦波（见公式（7）和（8）），这样一定程度上增加模型的泛化能力。

$$PE(pos,2i) = sin\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \quad (7)$$

$$PE(pos,2i+1) = cos\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \quad (8)$$

其中 d_{model}是位置向量的维度数；*pos* 是位置的索引值；*2i* 和 *2i+1* 分别表示 *pos* 位置向量中的偶数和奇数位置。

为了与 BERT 模型得到一致性，本文的所有模型的输入向量由音节向量和位置向量之和组成，其结果见图 4。这与原本的 BERT 模型相比，输入向量中缺少一个分别表示两个句子的向量，因为本实验中只考虑了单个句子的检错任务，而没有涉及连续两个或更多的句子。

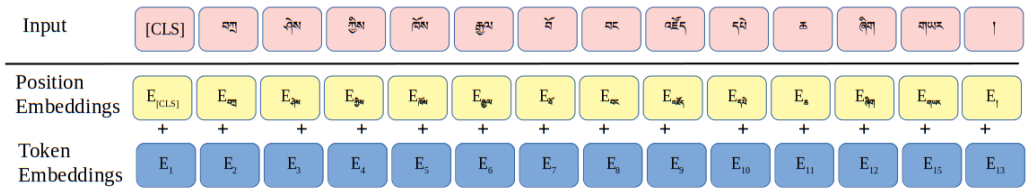

图 4 输入向量的结构

本文的所有模型中都有两个主要任务，分别为：

第一是能正确判别每个句子是否为病句，这与图 4 中的[CLS]对应,是专门用来学习整个句子是否为病句的参数；

第二是能够正确定位出句子中的错误位置，这与图 4 中的其余输入向量密切相关。

因此训练模型时有两个损失值并分别为公式（9）和（10）所示：

$$loss1(X,Y) = \sum_{x\in X}\sum_{i=0}^{1} Y_x \log(p_i(x)) \quad (9)$$

$$loss2(X,Y) = \sum_{x\in X}\sum_{i=0}^{4} Y_x \log\left(p_i(x)\right) \quad (10)$$

其中 Y_x和 $p_i(x)$分别是 x 的真实和预测标签，由于模型的收敛问题，最终用公式（11）作为模型的损失值。

$$loss(X,Y) = loss1(X,Y) + loss2(X,Y) \quad (11)$$

另外参数优化方法用 Adam 优化器。

3 实验数据

3.1 加噪算法

由于藏文缺乏标注数据，采用人工加噪的方法来构建训练数据的负样本。一般的加噪方式有两种：一种是把所有音节集作为混淆集后统一进行随机替换。另一种是利用先验知识给每个音节

构建不同的混淆子集，该子集在语言信息方面具有一定的关联，如动词时态变化、虚词搭配、音相似及形相似等信息。由于第一种方法缺乏错误逼真的能力，所以本文中没有采用该方式。第二种方法藏文著作中具有非常详细的总结和分析，因此不仅容易收集，而且有效。比如动词时态变化的例子为"ཟེན་ཟེན་འདུག་པ་འཕྲི་བཞིན་པའི། །གནས་སྐབས་དེ་ཉེ་དུ་ཕྱུག །འཕྲོ་ནས་ཟིན་འཐུན་མ་ཚོངས་པ། །"；虚词搭配关系的例子为"དུ་བ་ས་ཀྱི་གང་གི །ནི་མ་རུ་ལའི་རྩེས་ལུགས། །"；音相似的例子为"གཅིལ་དུ་དུལ་བ་བྱལ་རྣམས། །བསྒྲིགས་དང་རལ་བ་ཀྱིས་བཟོ། །"；形相似的例子为"ཚང་དང་ཚེང་དང་ཁིང་རྣམས་ཤེ། །ཚང་ཆེན་ཚེང་གཞི་སྐམ་ཟའི་ཁིང་། །"。目前共收集了 3912 个音节的混淆子集，部分见表 1。

表 1 混淆子集的部分例子

类型	原词	加噪词
动词时态	འབྱེ	འབྱེ བྱེ བྱེས སྤྱེས
虚词搭配	ཀྱི	ཀྱི ཀྱིས ཀྱི ཀྱིས གི གིས
音相似词	ཀྱལ	ཀྱལ ངྱལ བྱལ
形相似词	ཚང	ཚང ཚེང ཁིང

收集混淆子集之后，给训练句子加噪就比较容易，其加噪步骤见加噪算法。

加噪算法：

输入：训练集
输出：加噪的训练集
给定含有混淆子集的全集 D
给定 step
for 句子 S∈训练集 *do*
  W←从 S 中随机不重复地选取$\lceil \frac{|s|}{step} \rceil$个音节
  for w ∈ W *do* //制造误用错误
    C←从 D 中抽取包含 w 的替换子集
    if |C| > 1 then
      v←从 C 中随机选取一个与 w 不同的音节
    else v←音节加噪法（w）
    把 S 中的 w 替换成 v
  end for
end for
return 加噪的训练集

音节加噪法

输入：一个音节 w
输出：加噪音节 v
if w 中有再后加字“ས” then v←去掉 w 中再后加字
else if w 中有元音 then
  v←把 w 中的元音用其余三个元音随机替换
else v←给 w 随机添加一个元音
return v

其中$\lceil x \rceil$表示x的向上取整，子集 C 中最多一个音节时说明该音节没有混淆子集，需要特殊处理。由于句子越长其中出现错误音节的概率也越大，因此加噪算法对该句子有必要添加多个错误音节，而参数 step(该参数对结果的影响见表 5)刚好可以限制每个句子中错误音节的个数。

3.2 数据规模

本实验使用的训练数据为高质量的藏文古典著作。数据预处理的步骤为：首先，对文本进行粗略分句（因为藏文句子的边界特征比较模糊）；然后按音节个数筛选句子，只保留了 5 到 80 个音节数的句子；最后对每个句子按音节切分。目前共收集了 420705 个句子，其中含有 1.07 千万个音节，去重后有 6580 个，详细统计情况见表 2。

表 2 数据的分布情况

数据	训练集	验证集	测试集
原句	40 万	1 万	1 万
噪句	80 万	1 万	1 万
Step	10	10	10

表 2 中的噪句是由加噪算法对原句处理后的句子，并且该过程中具有随机性，因此噪句的个数可以限制。为了得到错误音节的多样化及更广的覆盖面，训练集中原句和噪句的比例设定为 1:2（共 120 万句），而验证集和测试集中比例设定为 1:1。step 表示噪句S中含有$\lceil \frac{|s|}{10} \rceil$个错误音节。

4 实验结果及分析

4.1 实验结果

本实验在相同的超参和数据规模下对比了各个模型，其中隐藏层的层数为 2、 音节向量的维度和隐藏层的神经单元个数都为 128、学习率和 Dropout 的值分别为 0.001 和 0.1、加噪算法中的 step 为 10、原句和噪句的个数比例为 1:2，其识别结果表 3 所示：

表 3 实验结果

模型		BiLSTM	BiLSTM -Atten	BERT
验证集	句子	50.00	50.00	94.14
	音节	95.64	97.08	97.92
测试集	句子	50.00	50.00	93.74
	音节	96.50	97.01	97.86

通过分析训练集后得知，其中 89.68%的句子中音节个数少于 50，而只有 10.32%的句子中音节个数多余 50，因此训练集的大多数句子中需要补充 30 及以上个 0。若每个句子中补充的 0 太多之后，对 BiLSTM 和 BiLSTM-Atten 模型对噪句的识别受到很大的影响，即把所有句子都识别成噪句，所以其正确率只有 50%。三个模型对音节正字检测的正确率都高于 96.5%，其主要原因之一是无误音节的个数远远多于错误音节的个数，与此同时也能发现这三个模型对无错误音节的误判率都较低。表 3 中可见，BERT 模型在音节及句子检测结果中都高于两个基线模型，其主要原因归功于 BERT 模型的深层双向的表征学习能力。

4.2 实验分析

4.2.1 数据领域的影响

表 3 中的验证集和测试集都是与训练集同一个领域的句子，因此其正确率也相对较高。为了验证本方法在不同领域中的可靠性，另外准备了两种不同类型的测试集分别为：中小学藏文教材中的 3199 个原句（测试集 2）和新闻语料中的 6474 个原句（测试集 3），并通过加噪算法来自动生成病句，且比例限制为 1:1，对音节及句子检测结果分别表 4 和 5 所示：

表 4 音节检测结果

	正->正		负->负	
	BiLSTM-Atten	BERT	BiLSTM-Atten	BERT
测试集 1	97.54	99.26	80.75	74.53
测试集 2	96.24	97.76	62.67	63.214
测试集 3	96.75	98.09	59.84	59.62

表 5 BERT 对句子的检测结果

	正->正	负->负
测试集 1	94.17	89.92
测试集 2	78.03	81.31
测试集 3	76.63	85.93

表 4 和 5 中“正->正”和“负->负”分别表示对音节或句子的正样本正确地识别为正样本，而负样本正确地识别为负样本。从表 4 中可见，BERT 模型对音节检测中正样本的误判率低于不仅低于 BiLSTM-Atten 模型，而且在最坏的情况下也低于 2.3%。对于负样本的误判率而言，两者模型的误判率都相对较高。主要难以处理的有动词时态的变换、出现未登录音节、虚词之间的变换，部分例子如表 6 所示：

表 6 难以处理的例子

原句	加噪句
ངས་སློབ་སྒྲོང་ཁྱམ་ར།	ངས་སློབ་སྒྲོང་ཁྱོད་ར།
ཁོ་ཡིས་བགོད་སྐྱེག་གནང་།	ཁོ་ཡིས་འགོད་སྐྱེག་གནང་།
འཁྲ་ཤེས་ཀྱིས་ཨ་ཞང་ལ་བཤད།	འཁྲ་ཤེས་ཀྱི་ཨ་ཞང་ལ་བཤད།
ཨོ་རྒྱན་རིན་པོ་ཆེ།	ཨོས་རྒྱན་རིན་པོ་ཆེ།

第一种生成的噪句并非是一个病句，只是在这一句中很难确定动词的时态，需要更长的上下文信息才能解决。第二种生成的噪句是虽然是病句，但是藏文文本中真实存类似使用法，比如：འགོ་འདུག་སྐྱོད་、ཐོས་བསམ་སྒོམ་ 和 ཚགས་ཐང་རྩོམ་ 等，因此这种错误模型无法有效判别。第三种生成的噪句跟第一种有点类似，把原句中的属格和作格相互替换后，该句子任然不是一个病句，只是语义发生了变化，同样这种类型也需要依赖更长的上下文信息才能解决。第四种生成的噪句中，该错误音节成为未登录音节，从而导致模型的误判。

4.2.2 参数 step 的影响

为了验证每个原句中设置多少个错误音节才能达到最佳的效果。对 BERT 模型检测了同一个句子中不同的错误个数的影响，其结果如表 7 所示：

表 7 不同错误音节个数的影响

step		6	8	10	12	14
测试集 1	句子	96.50	95.80	93.74	93.59	92.21
	音节	96.41	97.43	97.86	97.95	98.04
测试集 2	句子	87.06	85.92	83.63	81.43	79.77
	音节	94.41	95.23	96.15	96.01	96.14
测试集 3	句子	87.10	85.88	83.71	82.88	81.25
	音节	94.28	95.37	96.01	96.15	96.39

从表 7 中可知，若句子中错误音节越多时对病句的识别越容易，即该句子中错误特征很明显，反之显而易见，句子中缺乏错误特征不易判别。对于检测错误音节而言，若句子里错误音节越少时检测结果越好，主要原因是错误越少时任务越简单，上下文信息越正确。总之使用无监督的藏文正字检错时病句中的错误音节个数的控制是一

个不可忽略的一个环节。

4 总结与展望

由于藏文缺乏语法错误标注数据，所以本文针对这个问题：首先提出一个通过语言知识来自动生成语法错误样本的算法，即该算法可以自动生成训练数据中的负样本，且其个数可控制；然后构建 BERT 模型，其输入向量只有两个向量的之和，另外没有使用遮蔽语言模型。最后为了证明本文所提出的方法的有效性，构建两个基线模型和三种不同领域的测试数据，实验结果证明了本方法的结果优于两个基线模型。该模型对句子分类的效果很好，在同领域测试集上能达到 93.74%，不同领域测试集上也能达到 83.6%。对错误音节而言，同领域测试集上识别结果能达到 74.53%，不同领域测试集上相对较低只有 59.62%，但是对无错误音节的误判率只有 2.3%。对于一个无监督的方法而言，这个结果较理想。

今后将进一步扩充藏文文本的规模，利用更加丰富的语言知识，提升语法检错的识别结果，同时研究语法错误纠正工作。

参考文献

[1] 张仰森,俞士汶.文本自动校对技术研究综述[J].计算机应用研究,2006(06):8-12.

[2] 刘磊,梁茂成.英语学习者书面语法错误自动检测研究综述[J].中文信息学报,2018,32(01):1-8.

[3] 张梅,印勇.英语作文计算机评分技术综述[J].外语电化教学,2010(06):44-47+52.

[4] 杨晓琼,戴运财.基于批改网的大学英语自主写作教学模式实践研究[J].外语电化教学,2015(02):17-23.

[5] Zhuo-Ran Liu,Yang Liu. Exploiting Unlabeled Data for Neural Grammatical Error Detection[J]. Journal of Computer Science and Technology, 2017,32(4).

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[J]. arXiv preprint arXiv: 1810.04805, 2018.

[7] 骆卫华,罗振声,宫小瑾.中文文本自动校对技术的研究[J].计算机研究与发展, 2004,41(1): 244-249.

[8] Hao Li, Aodengbala, Gong Zheng, et al. A Research on Automatic Proofreading for Mongolian Text Based on Bayes Algorithm[J]. Journal of Inner Mongolia University, 2010,41(4):440-442.

[9] Ren H., Yang L., Xun E. A Sequence to Sequence Learning for Chinese Grammatical Error Correction[J]. NLPCC 2018: Natural Language Processing and Chinese Computing,Springer:401-410.

[10] Zhou J., Li C., Liu H., Bao Z., Xu G., Li L. Chinese Grammatical Error Correction Using Statistical and Neural Models[J]. NLPCC 2018: Natural Language Processing and Chinese Computing,Springer:117-128.

[11] Youdao's Winning Solution to the NLPCC-2018 Task 2 Challenge: A Neural Machine Translation Approach to Chinese Grammatical Error Correction[J]. NLPCC 2018: Natural Language Processing and Chinese Computing,Springer:341-350.

[12] Dale R,Anisimoff I and Narroway G.Hoo 2012: A report on the preposition and determiner error correction shared task[C]//Proceedings of the Workshop on Building Educational Applications Using Nlp.Association for Computati Linguistics, 2012:54-62.

[13] Ng H T,Wu S M,Briscoe T,et al.The CoNLL-2014 Shared Task on Grammatical Error Correction[C]//Proceedings of the Eighteenth Conference on Computational Natural Language Learning:Shared Task, 2014:1-14.

[14] 才智杰,孙茂松,才让卓玛.一种基于向量模型的藏文字拼写检查方法[J].中文信息学报,2018,32(09):47-55.

[15] 珠杰,李天瑞,刘胜久.TSRM 藏文拼写检查算法[J].中文信息学报,2014,28(3):92-98.

[16] 色差甲,贡保才让,才让加.藏文音节拼写检查的 CNN 模型[J].中文信息学报,2019,33(01):111-117.

[17] Vaswani A, Shazeer N, Parmar N, et al. Attention Is All You Need[C]// Advances in Neural Information Processing Systems. Long Beach: NIPS,2017:6000-6010.