Computer Science and Application 计算机科学与应用, 2020, 10(6), 1075-1086
Published Online June 2020 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2020.106112

Hans汉斯

Subword-Level Chinese Text Classification Method Based on BERT

Sirui Li

School of Computer Science, Chengdu University of Information Technology, Chengdu Sichuan
Email: 635840708@qq.com

Received: May 15th, 2020; accepted: May 28th, 2020; published: Jun. 4th, 2020

Abstract

With the development of the times, the number of text in the network is growing rapidly. In order to extract and process the text efficiently, it is necessary to classify the text. Based on the BERT model, this paper proposes a Chinese text classification method at the seed word level. In this method, the subword-level masking method is used to improve the original masking language model, so that it can effectively mask the complete Chinese words, and increase the word vector expression ability of BERT model for Chinese text. At the same time, Chinese word position embedding is added to make up for the lack of Chinese word position information in BERT model. The experimental results show that the BERT model of this text classification method has the best classification effect compared with other models in multiple Chinese data sets.

Keywords

BERT Model, Subword Level, Text Classification, Masked Language Model

基于BERT的子词级中文文本分类方法

李思锐

成都信息工程大学，计算机学院，四川 成都
Email: 635840708@qq.com

收稿日期：2020年5月15日；录用日期：2020年5月28日；发布日期：2020年6月4日

摘　要

随着时代的发展，网络中文本数量飞速增长，为了高效地提取和处理，对文本进行分类必不可少。该文以BERT模型为基础，提出了一种子词级的中文文本分类方法。在该方法中，使用子词级遮蔽方法改进

文章引用: 李思锐. 基于 BERT 的子词级中文文本分类方法[J]. 计算机科学与应用, 2020, 10(6): 1075-1086.
DOI: 10.12677/csa.2020.106112

原有遮蔽语言模型，使其能有效遮蔽完整中文单词，增加了BERT模型对中文文本的词向量表达能力。同时新加入了中文单词位置嵌入，弥补了BERT模型对中文单词位置信息的缺失。实验结果表明，使用了该文文本分类方法的BERT模型，在多个中文数据集中对比其他模型均拥有最好的分类效果。

关键词

BERT模型，子词级，文本分类，遮蔽语言模型

1. 引言

文本分类是自然语言处理领域的核心问题之一，也是大数据时代高效获取和处理数据的基础，使用范围非常广泛。近几年来，随着计算能力的提升，以及 2006 年 Hinton [1]等人提出了深度学习的概念以来，文本分类迎来了新一轮的技术进步。在图形计算单元(gpu)以及并行计算技术的支持下，研究者可以更轻易地训练拥有更深、更多参数的模型，深度学习开始被广泛应用于文本分类研究与应用中，如 Yoonkim [2]首次将卷积神经网络(CNN)应用在文本分类任务上，Liu 等人[3]提出的基于循环神经网络(RNN)的文本分类方法，Joulin 等人[4]提出的 Fasttest 模型，以及 Szegedy 等人[5]提出的 Inception 结构，都在文本分类任务中取得了优秀的成果。

但是随着互联网技术的发展，传统深度学习分类模型在面对文本类型越来越丰富的网络文本时，都存在着泛化性较差的问题，比如将处理新闻分类任务的模型用来处理商品评论分类或其他差别较大的其他文本时，分类准确率就会大幅降低，需要调整模型参数并再次训练。

针对这个问题，2018 年华盛顿大学提出了预训练模型 ELMO [6] [7]，使用多层双向长短时记忆网络(multi-layer bilstm)对语句建模，并通过下一个词预测训练任务构建通用词向量表达。这种通过预训练的语言模型，能直接使用得到的词向量处理自然语言处理任务，不仅拥有很强的特征提取和学习能力，同时能极大提高文本处理的泛化性。实验表明 ELMO 模型在 6 个自然语言处理任务上取得了领先成绩，将结果平均提高了两个百分点。但不久之后，Radford 等人[8]提出了预训练模型 OpenAI GPT，使用 Transformer [9]模型中的解码器(Decoder)来代替 Elmo 中的双向长短时记忆网络，在同样使用下一个词预测的训练任务的情况下，在 12 个自然语言处理任务中刷新了其中 9 个任务的最好成绩。

但 Elmo 与 GPT 都受限于下一词预测的单向限制问题[10]，导致模型无法准确预测部分词语[11]。针对这个问题谷歌实验室提出了预训练模型 BERT [12]，使用全新预测任务遮蔽语言模型[13] [14] (MLM masked language model)来解决单向限制的问题。同时使用不同的 Transformer 编码器(Encoder)部分，使模型的参数量比 GPT 少 4 倍左右。通过大量实验，BERT 模型再次刷新了 11 个自然语言处理任务上的最好成绩，是目前预训练方法中最优秀的模型。自此之后许多研究者开始着手于 BERT 模型的研究，如 Liu 等人[15]将预训练的 BERT 模型和多任务学习进行结合，以求获取更好的效果。Sun 等人[16]通过修改模型的输入处理，将单句分类问题改造成 BERT 更擅长的双句分类问题进行处理。Sun 等[17]着重研究了 BERT 在多个文本分类任务上的表现，详细分析了 BERT 的特点与优势。

然而，本文发现 BERT 模型在处理中文文本时，由于遮蔽语言模型只会遮蔽并预测单个的中文字符，而不是完整的中文单词，且输入模型的向量中，缺少中文单词位置信息的原因，导致模型构建的中文词

向量表示差，分类准确率不如英语等其他语言。为了解决 BERT 处理中文时存在的不足，本文提出了一种基于 BERT 的子词级中文文本分类方法。在遮蔽语言模型中，通过子词级的文本表达，和独特的子词级遮蔽方法，实现了对完整中文单词的遮蔽。同时修改了 BERT 模型的输入处理，加入中文单词位置嵌入，弥补了中文单词位置的缺失。实验证明，相较于原始 BERT 模型，本文提出的方法在面对中文时，拥有更好的词向量表达能力与分类准确率。

2. BERT 相关理论

BERT 模型的全称是基于预训练的深度双向 Transformer 语言模型(Bidirectional Encoder Representations from Transformers)，作为一种预训练模型，不同于卷积神经网络与循环神经网络，该模型使用 Transformer 模型的编码器部分作为模型的基础。得益于 Transformer 编码器的强大能力，BERT 模型可以增加到非常深的深度，充分发掘深度神经网络模型的特性，提升模型准确率。

2.1. BERT 模型的输入处理

BERT 在设计之初就是为了构建出一个通用的语言模型，所以 BERT 在输入处理上充分考虑了各式各样任务的需求。BERT 模型除了与 Transformer 一样在词向量嵌入(Token Embedding)的基础上添加位置嵌入(Position Embedding)之外，为了处理一些涉及句子对(Sentence Pair)的问题，还添加了句子位置嵌入(Segment Embedding)。如图 1 所示：

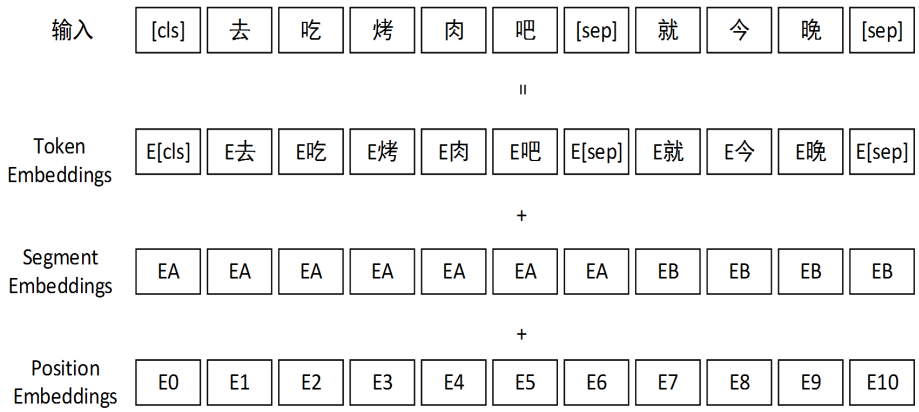

Figure 1. BERT model input processing
图 1. BERT 模型输入处理

由图 1 可知，每一个文本序列的首位添加了一个特殊标记[cls]，不同句子之间也添加了一个特殊标记[sep]。[cls]标记是最终输出时整个句子或句子对的特征表征，[sep]标记是句子之间的界线。

2.2. 遮蔽语言模型

遮蔽语言模型是 BERT 模型中使用的全新预测任务。与传统预测任务的词预测不同，该预测任务通过完全遮蔽的输入文本中的部分单词，使得预测被遮蔽单词时能充分考虑上下文信息，以一种更符合人类语言习惯的过程来学习表达词向量。在遮蔽语言模型中，为了获得更好的训练效果，会进行以下两步操作：

第一步为遮蔽区域的选择，在 BERT 中模型会随机选择 15%的单词作为遮蔽区域。保证训练的有效性。

第二步为遮蔽方式的选择，在 BERT 中，为了解决遮蔽语言模型在完全遮蔽单词后，被遮蔽的单词将无法被下游任务感知，从而导致模型预处理阶段与微调阶段文本不统一的问题，在训练过程中对遮蔽

词会选择以下三种遮蔽方式：

a) 遮蔽词在训练中 80%的时间会被特殊记号[mask]代替。如图 2 所示：

Figure 2. Mask of BERT model
图 2. BERT 模型的遮蔽

b) 而在训练中 10%的时间会被词典中的一个随机词代替。如图 3 所示：

Figure 3. The replacement of BERT model
图 3. BERT 模型的替换

c) 在剩下的 10%的时间里遮蔽词会保持不变。如图 4 所示：

Figure 4. Invariance of BERT model
图 4. BERT 模型的不变

2.3. BERT 模型的分类任务

BERT 模型不同于卷积或循环等传统分类模型，该模型会在分类之前，先在大语料环境进行预训练。通过预训练后的 BERT 模型虽然可以直接用来分类，但是往往准确率较差，需要在特定任务的数据集上进行进一步的预训练，在该阶段，模型会在原先预训练的基础上使用目标数据再次进行如遮蔽语言模型或下一句预测任务，使自身的参数适用于目标数据，这相当于将模型从一般领域到目标领域进行了一次迁移学习。最后通过对两次预训练后的模型进行分类效果的精加工(Fine-tuning)，也就是微调训练后输出分类结果。BERT 模型分类流程如图 5 所示：

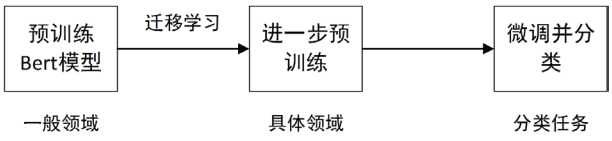

Figure 5. BERT model training process
图 5. BERT 模型训练流程

得益于预训练阶段后优秀的词向量表达与模型参数，模型能轻易地捕捉语句的深层抽象特征，所以模型微调阶段大多只是在训练输出层的参数和略微调整预训练模型而已，这保证了微调阶段的收敛速度和分类准确率。

3. 基于 BERT 模型的子词级文本分类方法

3.1. 中文子词级文本表示

本文使用的子词级粒度的文本表示最早应用于英文文本翻译任务中。通过词切片[18] (Word Piece)的方法将英文单词再次切分，如“The highest mountain”这句话进行子词级表示后，结果如图 6 所示。虽然子词级的粒度小于词级，但是仍大于字符级，这就保证了子词级不会像字符级一样丢失文本语义信息，同时也解决了词级所需词汇库过大的问题。

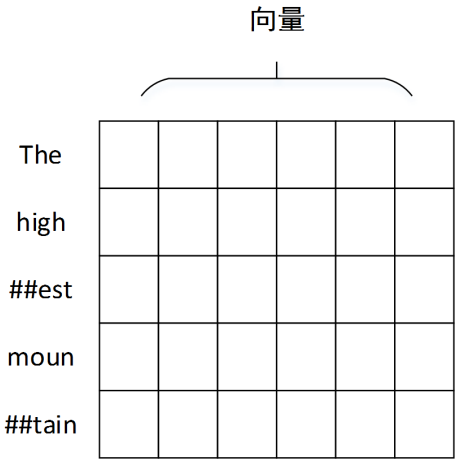

Figure 6. English subword level representation
图 6. 英文子词级表示

而在中文使用子词级时，与英文的操作不同。在对文本进行分词处理后，会先区分单词中的词首和非词首字符，对所有非词首字符添加“##”号，并生成一行标记序列 seg_labels，使用 0 来标记所有没添加“##”号的字符，使用 1 来标记添加了“##”号的字符，使用−1 来标记其余标点符号与占位符等。该标记序列不仅区分了单词的界线，同时也标明了不需要处理的标点符号等，其具体操作如图 7 所示：

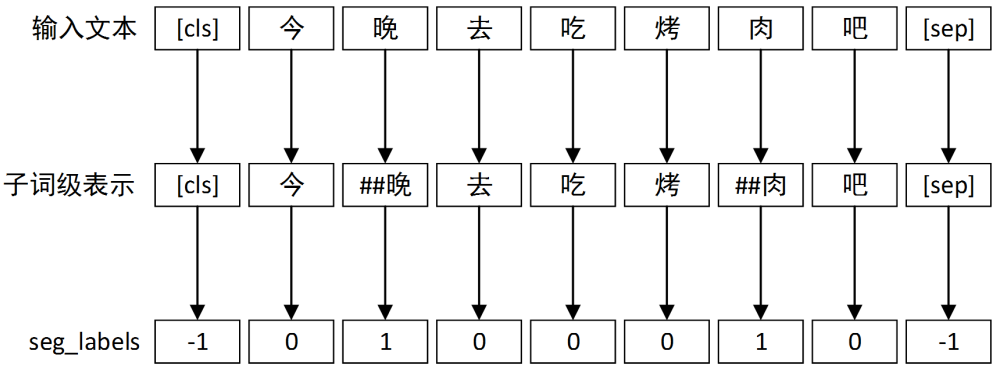

Figure 7. BERT model subword level text representation
图 7. BERT 模型子词级文本表示

添加了“##”号的字符拥有与原字符不同的数字或矢量，这样在仅增加一倍的词汇库大小的情况下，能够保留中文单词的语义信息。同时由于中文子词级表示中，词首字符就像是单词的主键一样，通过词首能有效减少对中文单词的操作流程和难度，对于需要提出遮蔽完整中文单词，以及中文位置的标注方法的本文来说是必不可少的。所以本文模型选择子词级的文本表示粒度。

3.2. 子词级遮蔽方法

在 BERT 模型中，由于英文与中文语言结构的差异，遮蔽语言模型在中文中选择遮蔽区域时，只会选择中文的字符，而不是完整的中文单词，为了解决这个问题，本文提出一种使用子词级遮蔽方法的遮蔽语言模型。对章节 2.2 中提到的遮蔽语言模型两个步骤做出以下修改。

第一步遮蔽区域的选择中，在模型随机选用 15%的遮蔽区域前，使用章节 3.1 中的得到的标记序列 seg_labels，按照词首决定整个单词是否遮蔽的原则，重新整合中文字符关系，使待遮蔽区域可以覆盖整个中文单词，同时排除占位符与标点符号。该方法步骤的伪代码如表 1 所示：

Table 1. Subword level masking area selection code
表 1. 子词级遮蔽区域选择代码

输入：文本序列，标记序列 seg_labels。 输出：整合遮蔽区域的文本序列。	
1. for sent_index, token in enumerate(sent):	\\遍历文本序列
2. seg_label = seg_labels[sent_index]	\\将文本对应标记序列
3. if seg_label == -1:	\\如果为标点符号与占位符，进入判断
4. if pos == -1：continue	\\如果位置信息为空，跳过
5. else：遮蔽区域 = pos + num; pos = -1; num = 0	\\如果位置信息不为空，将位置信息加计数的区域设为待遮蔽区域。清空位置信息，清空计数
6. if seg_label == 0:	\\如果为词首，进入判断
7. if pos == -1：pos = sent_index	\\如果位置信息为空，记录词首位置
8. else：遮蔽区域 = pos + num; pos = sent_index; num = 0	\\如果位置信息不为空，将位置信息加计数的区域设为待遮蔽区域。更新位置信息，清空计数
9. if seg_label == 1: num += 1; continue	\\如果为非词首，计数加一，跳向下一字符

在第二步遮蔽方式的选择中，按照非词首的遮蔽方式是否同词首一致的区别，子词级遮蔽方法可以继续细分为两种：

第一种子词级随机遮蔽方法中，一个单词中每个字符是否会被遮蔽服从词首，但是遮蔽方式都是单独判断。在某一个时刻可能都不相同，如同随机一样，如图 8 所示：

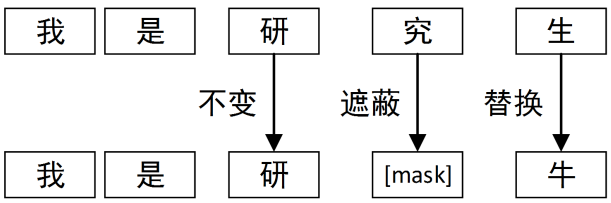

Figure 8. Subword level random masking method
图 8. 子词级随机遮蔽方法

在第二种子词级遵从词首的遮蔽方法中，一个单词中非词首字符不再单独进行遮蔽方式的判断，而是同判断是否遮蔽时一样与词首保持一致。如图 9 所示：

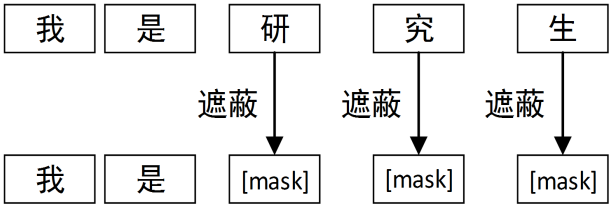

Figure 9. The method of concealing the first part of a subword
图 9. 子词级遵从词首遮蔽方法

这两种子词级遮蔽方法都可以实现对整个单词的遮蔽操作，但哪个方法在真实数据中词向量表达效果更好，本文将在下文进行对比实验。

3.3. 中文单词位置嵌入

使在文本分类中大多数都是单句分类任务，不需要考虑句子位置的信息。所以本文模型的输入处理放弃使用句子位置嵌入(Segment Embedding)。同时由于 BERT 中原本用来标记单词位置的位置嵌入(Position Embedding)在面对中文时，标记的只是中文字符的位置，缺失了单词的位置信息。所以本文提出一种子词级的中文单词位置嵌入(Word Position Embeddings)，使用章节 3.1 中得到的 seg_labels 序列标记的词首和非词首信息，对于不同字符：

a) 遇到于词首与占位符，位置计数加一。

b) 遇到非词首时，位置计数与其对应词首一致。

如图 10 所示：

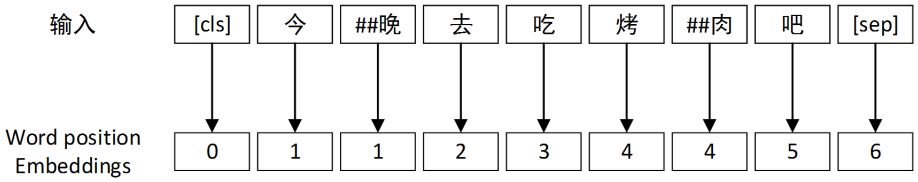

Figure 10. Chinese word position count
图 10. 中文单词位置计数

并将模型的输入处理改为词向量嵌入(Token Embedding)，位置嵌入(Position Embedding)和单词嵌入(Word Position Embeddings)的总和，如图 11 所示：

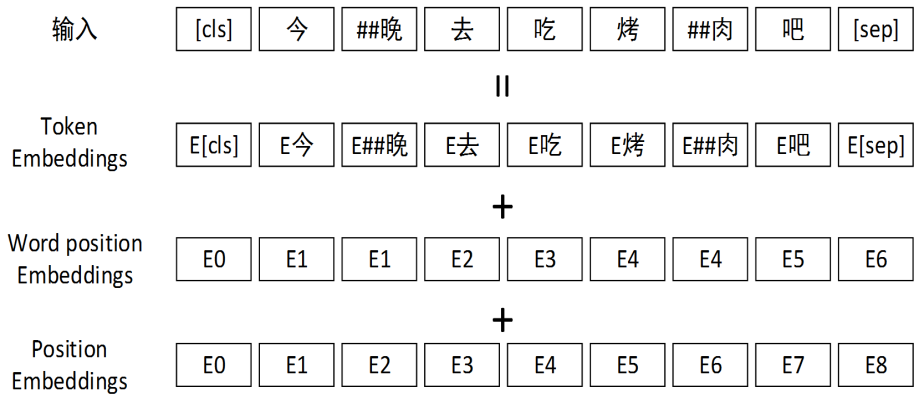

Figure 11. Chinese word position embedding
图 11. 中文单词位置嵌入

4. 实验与分析

本文所有实验都基于 NVIDIA tesla V100 gpu 与深度学习框架 PaddelPaddel 来实现，所使用的语言为 python，并开启 gpu 加速，加速版本为 CUDA9.0、cudnn7.3。

4.1. 实验数据集

本文使用来自网络的梨视频视频标题数据集、网易网络新闻数据集、京东商品评论数据集。三个数据集包含网络中常见文本类型，都由本文使用爬虫程序从网络获取。数据集具体情况如下所示：

a) 梨视频视频标题数据集：该数据集由本文收集整理，数据由爬虫软件从短视频网站获取，包含音乐、二次元、搞笑等共 9 个分区的共 2 万条文本数据。

b) 网易网络新闻数据集：该数据集也是由本文收集整理，数据来源自网易新闻频道，包含娱乐、体育、健康等 8 个频道的共 1 万 5 千条文本数据。

c) 京东商品评论数据集：该数据集由本文收集整理，数据来源于京东手机、笔记本电脑、图书、食品、家居 5 个分类下的商品评论信息，按好评、中评、差评三个评价等级收集，共 7 万 7 千条文本数据。

4.2. 本文模型参数介绍

该模型的部分参数如表 2 所示：

Table 2. Model parameters and super parameters
表 2. 模型参数与超参

名称	含义	取值
hidden_size	模型向量维度	768
num_attention_heads	模型注意力头数	12
batch_size (预训练)	模型每批处理大小	4096
learning_rate (预训练)	学习速率	1e−4
batch_size (微调)	模型每批处理大小	24
learning_rate (微调)	学习速率	5e−5

在预训练阶段，训练将最多进行 3 万个 step，每 10 个 step 输出一次训练结果，每 100 个 step 在验证集上进行一次验证，每 1 万个 step 保存一次模型。

在模型微调阶段，为了取得更好的训练效果，将学习速率减为 5e−5，同时为了减少计算资源压力，将模型每批处理大小减少为 24。在该阶段，训练最大进行 10 个 epoch，每 10 个 step 将输出一次训练结果，每 100 个 step 在验证集与测试集上进行一次验证，每 1000 个 step 保存一次模型。为了保证模型训练时每个 epoch 的模型保持不变，固定模型的随机种子(random_seed)。模型最后分类标签数(num_labels)，与模型输入的数据集相关。

4.3. 评价指标

在衡量不同方法的词向量表达能力的实验中，本文采用文本置信度(Perplexity，ppl)来衡量好坏，在语言模型中 ppl 值可以认为是平均分支系数(Average branch factor)，即预测下一个词时可以有多少种选择，如 ppl 值为 60，可以直观地理解为，在模型生成一句话时下一个词有 60 个合理选择，可选词数越少，一般认为模型越准确。所以 ppl 值越小，词向量模型越好。其中 ppl 值的计算方法如公式 1

所示：

$$ppl = 10^{-\frac{1}{N}\log p(S)} \tag{1}$$

在衡量文本分类准确率的对比试验中，本文采用文本分类的准确率(accuracy)来衡量好坏，准确率是所有样本中分类正确样本所占的比例。

而在衡量文本预测效果的好坏方面，由于缺少相关评价标准，所以本文直接列出原始文本与预测文本，通过对比文本流畅度与预测词的准确性来衡量预测的效果。

最后在对比处理速度时，采用更为直观的每秒处理 step 数来表示，单位为 step/s。

4.4. 评实验结果对比

4.4.1. 文本词向量表达能力对比实验

本章将从文本 ppl 值、真实数据的预测效果，和模型处理速度，三个方面来进行文本词向量表达能力的对比实验。在文本 ppl 值的对比实验中，为了使 ppl 值的对比更为明显，本文使用文本长度较长的网易网络新闻数据集作为实验数据集。同时为了充分对比本文方法的有效性，分别实现了以下 4 种 BERT 模型，如表 3 所示：

Table 3. Experimental model
表 3. 实验模型

名称	介绍
char_BERT	原始 BERT 模型，未加入中文词位置嵌入
word_char_BERT	原始 BERT 模型，加入中文词位置嵌入
word_random_BERT	使用子词级随机遮蔽方法的 BERT 模型，并加入中文词位置嵌入
word_allmask_BERT	使用了子词级遵从词首方法的 BERT 模型，并加入中文词位置嵌入

每个模型将在预训练阶段对大小为 10,000 与 1500 的数据进行训练，每个数据集训练三次，共 9 万个 step，结果取训练完成后 ppl 的平均值。实验结果如表 4 所示：

Table 4. Comparison of ppl values
表 4. ppl 值对比实验结果

遮蔽方法	10,000 条数据	1500 条数据
char_BERT	10.124173	16.399334
word_char_BERT	8.989015	15.476432
word_random_BERT	3.518629	13.895789
word_allmask_BERT	2.152880	12.000850

由上表结果可以看出，在加入中文单词位置后，word_char_BERT 的 ppl 值较未使用的 char_BERT 有了明显的下降。而使用了本文提出的子词级遮蔽方法的 word_random_BERT 和 word_allmask_BERT 的 ppl 值相较于未使用该方法的 word_char_BERT 有大幅的下降。最后在两种子词级遮蔽方法中，使用子词级遵从词首的遮蔽方法的 word_allmask_BERT 的 ppl 值，相较于使用子词级随机遮蔽方法的 word_random_BERT 更低。由上述结果可以得到，子词级遮蔽方法和中文单词位置嵌入都能有效降低文本 ppl，提高词向量表达。

但是单独对比 ppl 值无法体现本文提出方法在具体任务中的具体提升效果，所以本文在 ppl 值对比实验的基础上进行了真实数据上的遮蔽预测效果的对比实验，在该实验中使用 word_char_BERT、word_random_BERT、word_allmask_BERT 三个模型，在京东评论和网易新闻两个数据集中分别随机提取一句话，作为实验的预测文本。实验的方法为先遮蔽预测文本中的部分单词，然后观察遮蔽词的预测效果。实验数据与结果如图 12 和表 5 所示：

预测文本：1.成都欢迎你！成都是天府之国，也是美食之都。
2.算了吧，我觉得这电脑不差。

遮蔽后文本：1.[mask][mask]欢迎你！[mask][mask]是天府之国，也是美食之都。
2. 算了吧，我觉得这电脑[mask][mask]。

Figure 12. BERT model input processing
图 12. BERT 模型输入处理

Table 5. Experimental data
表 5. 实验数据

遮蔽方法	预测结果	预测词
word_char_BERT	1.太成欢迎你 太成是天府之国 也是美食之都 2.算了吧 我觉得这电脑太太	1.成都：太成 2.不差：太太
word_random_BERT	1.成都欢迎你 成都是天府之国 也是美食之都 2.算了吧 我觉得这电脑太错	1.成都：成都 2.不差：太错
word_allmask_BERT	1.成都欢迎你 成都是天府之国 也是美食之都 2.算了吧 我觉得这电脑不错	1.成都：成都 2.不差：不错

由表 4 可以看到，没使用子词级遮蔽方法的 word_char_BERT 模型的预测效果较差，不仅没有正确预测成都这个地名，同时也未正确预测评论文本。在使用子词级随机遮蔽方法的 word_random_BERT 中，对地名成都的预测准确，但在对评论文本的预测中，预测词与原词语义相反，由此构建的词向量在情感分析任务中会产生一定的不良影响。最后在使用了子词级遵从词首遮蔽方法的 word_allmask_BERT 中，不仅地名成都准确预测，同时对评论遮蔽词的预测结果也与原语义相同，在三种遮蔽方法中拥有最好的预测结果。

而在处理速度方面，由于本文提出的方法修改了 BERT 模型处理中文的粒度，同时额外添加了中文单词的位置信息，所以必然会对处理速度造成一定的影响。为了验证对速度的影响，本文将对比使用额外单词位置和子词级遮蔽方法前后的模型处理速度，使用的实验模型为 char_BERT、word_char_BERT 和 word_allmask_BERT，使用的评价标准为章节 4.3 中提到的每秒处理 step 数，实验结果如图 13 所示。

由图 13 可以看出，虽然使用子词级遮蔽方法和中文单词位置后模型处理速度有略微的下降，但是综合前面两个词向量对比实验可以看到，加入了单词位置嵌入的模型拥有更好的词向量表达能力，使用子词级遮蔽方法后模型拥有更好的词向量表达能力与预测能力，相较于使用这两种方法带来的词向量表达效果提升，处理速度的略微下降在可以接受的范围内。而在两种子词级遮蔽方法的比较中，子词级遵从词首的方法拥有更好的词向量表达与预测的能力，所以在后续分类效果的对比中，本文将使用该方法。

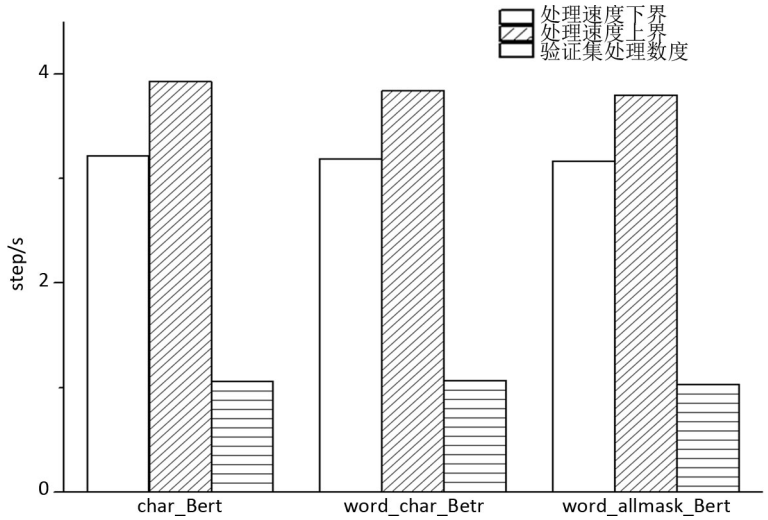

Figure 13. Processing speed comparison
图 13. 处理速度对比

4.4.2. 文本分类效果对比实验

本文主要的研究目标是中文网络文本的分类，所以为了验证本文提出的文本分类方法在文本分类任务中的有效性，将上文中拥有最好实验结果的 BERT 模型 word_allmask_BERT，同基础 BERT 模型、CNN 模型、RNN 模型、Fasttest 模型以及 Inception-CharFcn [19]模型，在梨视频视频标题数据集、网易网络新闻数据集、京东商品评论数据集，三个数据集上进行分类效果的对比实验，评价指标选用章节 4.3 中提到的准确率，结果将按百分比(%)表示。实验结果如表 6 所示：

Table 6. Classification accuracy comparison
表 6. 分类准确率对比

模型方法	梨视频视频标题数据集	网易网络新闻数据集	京东商品评论数据集
Cnn	83.33	92.36	79.43
Rnn	82.27	89.26	80.16
Fasttest	81.25	92.51	79.70
Inception-CharFcn	87.98	95.36	83.34
BERT	92.62	95.96	87.50
word_allmask_BERT	93.75	96.46	89.52

由上列实验结果可以看到，使用了中文单词位置嵌入和子词级遮蔽方法的 word_allmask_BERT 模型在三个数据集上都拥有最高的分类准确率，并且分类效果明显优于原始 BERT 模型，证明了本文提出的子词级文本分类方法对 BERT 模型中文文本分类准确率的提升效果。

5. 结束语

本文就 BERT 模型对中文的文本分类任务进行了研究，提出的子词级文本分类方法能有效解决 BERT 模型面对中文时的不足，同时增加中文文本分类的准确率。但是在面对越来越多的网络文本时，BERT 模型无法处理这类文本中容易出现的错别字问题，导致分类准确率下降。所以在下一步中，将尝试在微调阶段使用遮蔽语言模型中的遮蔽预测方法进行文本纠错。

基金项目

四川省科技厅重点项目(2017GZ0331)。

参考文献

[1] Hinton, G.E. and Salakhutdinov, R.R. (2006) Reducing the Dimensionality of Data with Neural Networks. *Science*, **313**, 504-507. https://doi.org/10.1126/science.1127647

[2] Kim, Y., Jernite, Y., Sontag, D., *et al*. (2016) Character-Aware Neural Language Models. *Thirtieth AAAI Conference on Artificial Intelligence*, North America, March 2016. https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12489

[3] Liu, P., Qiu, X. and Huang, X. (2016) Recurrent Neural Network for Text Classification with Multi-Task Learning.

[4] Joulin, A., Grave, E., Bojanowski, P., *et al*. (2016) Bag of Tricks for Efficient Text Classification.

[5] Szegedy, C., Liu, W., Jia, Y., *et al*. (2014) Going Deeper with Convolutions. *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Boston, 7-12 June 2015, 1-9. https://doi.org/10.1109/CVPR.2015.7298594

[6] Peters, M.E., Ammar, W., Bhagavatula, C., *et al*. (2017) Semi-Supervised Sequence Tagging with Bidirectional Language Models. *Proceedings of the* 55*th Annual Meeting of the Association for Computational Linguistics*, Volume 1, 1756-1765. https://doi.org/10.18653/v1/P17-1161

[7] Peters, M.E., Neumann, M., Iyyer, M., *et al*. (2018) Deep Contextualized Word Representations.

[8] Radford, A., Narasimhan, K., Salimans, T., *et al*. (2018) Improving Language Understanding by Generative Pre-Training. https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf

[9] Vaswani, A., Shazeer, N., Parmar, N., *et al*. (2017) Attention Is All You Need. *Advances in Neural Information Processing Systems*, Long Beach, CA, 2017, 5998-6008.

[10] Williams, A., Nangia, N. and Bowman, S.R. (2017) A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. *Proceedings of the* 2018 *Conference of the North American Chapter of the Association for Computational Linguistics*: *Human Language Technologies*, Volume 1, 1112-1122. https://doi.org/10.18653/v1/N18-1101

[11] Rajpurkar, P., Zhang, J., Lopyrev, K., *et al*. (2016) SQuAD: 100,000+ Questions for Machine Comprehension of Text. *Proceedings of the* 2016 *Conference on Empirical Methods in Natural Language Processing*, Austin, November 2016, 2383-2392. https://doi.org/10.18653/v1/D16-1264

[12] Devlin, J., Chang, M.W., Lee, K., *et al*. (2018) BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding.

[13] Taylor, W.L. (1953) "Cloze Procedure": A New Tool for Measuring Readability. *Journalism Quarterly*, **30**, 415-433. https://doi.org/10.1177/107769905303000401

[14] Xie, Z., Wang, S.I., Li, J., *et al*. (2017) Data Noising as Smoothing in Neural Network Language Models.

[15] Liu, X., He, P., Chen, W., *et al*. (2019) Multi-Task Deep Neural Networks for Natural Language Understanding. *Proceedings of the* 57*th Annual Meeting of the Association for Computational Linguistics*, Florence, July 2019, 4487-4496. https://doi.org/10.18653/v1/P19-1441

[16] Sun, C., Huang, L. and Qiu, X. (2019) Utilizing BERT for Aspect-Based Sentiment Analysis via Constructing Auxiliary Sentence.

[17] Sun, C., Qiu, X., Xu, Y., *et al*. (2019) How to Fine-Tune BERT for Text Classification? In: *China National Conference on Chinese Computational Linguistics*, Springer, Cham, 194-206. https://doi.org/10.1007/978-3-030-32381-3_16

[18] Wu, Y., Schuster, M., Chen, Z., *et al*. (2016) Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation.

[19] 李思锐. 字符级全卷积神经网络的文本分类方法[J]. 计算机科学与应用, 2020, 10(2): Paper ID 34199.