第47卷 第1期 计 算 机 工 程 2021年1月
Vol.47 No.1 Computer Engineering January 2021

·人工智能与模式识别· 文章编号：1000-3428(2021)01-0079-08 文献标志码：A 中图分类号：TP391

基于BERT模型的中文短文本分类算法

段丹丹[1]，唐加山[1]，温 勇[1]，袁克海[1,2]

（1.南京邮电大学 理学院，南京 210023；2.圣母大学 心理学系，美国 南本德 46556）

摘 要：针对现有中文短文本分类算法通常存在特征稀疏、用词不规范和数据海量等问题，提出一种基于Transformer的双向编码器表示(BERT)的中文短文本分类算法，使用BERT预训练语言模型对短文本进行句子层面的特征向量表示，并将获得的特征向量输入Softmax回归模型进行训练与分类。实验结果表明，随着搜狐新闻文本数据量的增加，该算法在测试集上的整体F1值最高达到93%，相比基于TextCNN模型的短文本分类算法提升6个百分点，说明其能有效表示句子层面的语义信息，具有更好的中文短文本分类效果。

关键词：中文短文本分类；基于Transformer的双向编码器表示；Softmax回归模型；TextCNN模型；word2vec模型

开放科学(资源服务)标志码(OSID)：

中文引用格式：段丹丹，唐加山，温勇，等.基于BERT模型的中文短文本分类算法[J].计算机工程，2021，47(1)：79-86.

英文引用格式：DUAN Dandan，TANG Jiashan，WEN Yong，et al.Chinese short text classification algorithm based on BERT model[J].Computer Engineering，2021，47(1)：79-86.

Chinese Short Text Classification Algorithm Based on BERT Model

DUAN Dandan[1]，TANG Jiashan[1]，WEN Yong[1]，YUAN Kehai[1,2]

（1.College of Science，Nanjing University of Posts and Telecommunications，Nanjing 210023，China；
2.Department of Psychology，University of Notre Dame，South Bend 46556，USA）

【Abstract】The existing Chinese short text classification algorithms are faced with sparse features，informal words and massive data.To address the problems，this paper proposes a Chinese short text classification algorithm based on the Bidirectional Encoder Representation from Transformer（BERT）model.The algorithm uses BERT pre-training language model to perform eigenvector representation of short text on the sentence level，and then the obtained eigenvector is input into the Softmax regression model for training and classification.Experimental results show that with the growth of data from Sohu news，the overall F1 value of the proposed algorithm on the test dataset is up to 93%，which is 6 percentage points higher than that of the TextCNN-based short text classification algorithm.The result demonstrates that the proposed algorithm performs better in semantic information representation at the sentence level，and in the classification of Chinese short texts.

【Key words】Chinese short text classification；Bidirectional Encoder Representation from Transformer（BERT）；Softmax regression model；TextCNN model；word2vec model

DOI：10. 19678/j. issn. 1000-3428. 0056222

0 概述

根据中国互联网络信息中心于2019年2月28日发布的第43次《中国互联网络发展状况统计报告》[1]，截至2018年12月我国网民规模达8.29亿，互联网普及率达到59.6%，其中网民通过手机接入互联网的比例高达98.6%，即时通信、搜索引擎和网络新闻是手机网民使用率最高的应用，这3类手机应用包含聊天记录、搜索日志、新闻标题、手机短信等大量短文本[2]，携带了丰富的数据信息，其已成为人类社会的重要信息资源，如何高效管理这些海量的短文本并从中快速获取有效信息受到越来越多学者的关注，并且对于短文本分类技术的需求日益突显。

国内学者针对中文短文本的分类研究主要包括中文短文本的特征表示与分类算法的选择与改进。文献[3]提出一种基于word2vec的中文短文本分类算法，使用

基金项目：南京邮电大学横向科研项目(2018外095)。

作者简介：段丹丹(1994—)，女，硕士研究生，主研方向为自然语言处理、数据分析；唐加山(通信作者)、温 勇、袁克海，教授。

收稿日期：2019-10-09 **修回日期：**2019-11-27 **E-mail：**tangjs@njupt.edu.cn

word2vec词嵌入技术对短文本的分词结果进行词向量表示，并使用TF-IDF对每个词向量进行加权，最终使用LIBSVM分类算法进行文本分类。实验结果表明，该算法可以有效提高短文本的分类效果。文献[4]提出一种全新的文本表示方法（N-of-DOC），即通过运用基尼不纯度、信息增益和卡方检验从短语特征中提取整个训练集的高质量特征。每篇文档提取的短语特征必须由这些高质量特征线性表示，再经word2vec词向量表示后，使用卷积神经网络（Convolutional Neural Network，CNN）的卷积层和池化层提取高层特征，最终利用Softmax分类器进行分类。实验结果表明，该方法在分类精度上相比传统方法提高了4.23%。文献[5]以微博为例，设置词和字两种特征粒度，选择信息增益率、信息增益、word2vec和特征频度来降低特征维度，重点探讨两种特征粒度在口语化短文本分类中的特点和作用，并得出在口语化短文本分类中选择字特征效果更好。文献[6]提出一种基于混合神经网络的中文短文本分类方法，先使用自定义特征词筛选机制将文档基于短语和字符两个层面进行特征词筛选，运用CNN结合循环神经网络（Recurrent Neural Network，RNN）提取文档的高阶向量特征，并引入注意力机制优化高阶向量特征。实验结果表明，在二分类及多分类数据集上，该方法能够在分类精度上比单模型取得更好的效果。文献[7]提出一种融合词频-逆文本频率（Term Frequency-Inverse Document Frequency，TF-IDF）和隐含狄利克雷分布（Latent Dirichlet Allocation，LDA）的中文FastText短文本分类方法。该方法在模型输入阶段对经过n元语法模型处理后的词典进行TF-IDF筛选，之后使用LDA模型对语料库进行主题分析以补充特征词典，使得模型在计算输入词序列向量均值时会偏向高区分度的词条。实验结果表明，该方法在中文短文本分类方面具有更高的精确率。

文献[8]采用正则化权值的方式对K近邻（K-Nearest Neighbor，KNN）算法进行改进，并结合粒子群优化（Particle Swarm Optimization，PSO）算法提高文本分类效果。文献[9]采用组合模型的方式，提出一种基于集成神经网络的短文本分类模型C-RNN。该方法使用CNN构造扩展词向量，再利用RNN捕获短文本内部结构的依赖关系，然后使用正则化项选取出模型复杂度和经验风险均最小的模型。实验结果表明，该方法对短文本分类具有较好的分类效果和鲁棒性。

上述针对短文本的特征表示算法均是将短文本进行分词或者分字，处理对象为字符或者词语层面的特征，而由于短文本具有特征稀疏的特性，字符或者词语不能表示短文本的完整语义，因此导致短文本的特征表示向量不能较好地代表短文本语义。文献[8-9]虽然对分类算法进行了改进，但是分类算法的输入仍是短文本的特征表示向量，特征表示向量的误差会向下传至分类器，因此，短文本的特征表示是提高短文本分类性能的关键步骤。基于以上研究，本文将对短文本的特征表示进行改进，提出一种基于Transformer的双向编码器表示（Bidirectional Encoder Representation from Transformer，BERT）[10]的中文短文本分类算法。

1 基于BERT的中文短文本分类

本文基于BERT的中文短文本分类算法主要由短文本预处理、短文本向量化以及短文本分类三部分构成，短文本预处理的目的是将输入的短文本整理成分类所需的文本，降低其他符号对分类效果的影响，然后对预处理后的短文本进行向量化表示并形成特征向量，最终将特征向量输入搭建好的分类器以实现短文本分类。

1.1 短文本预处理

中文短文本有多种预处理方式，而本文对于短文本的预处理过程具体如下[11]：

1）文本清洗。文本清洗主要包括去除特殊符号、去除多余空白及文本繁体转简体3个步骤。去除特殊符号以及多余空白是使短文本的特征表示尽可能地只关注短文本自身词汇的特征和语义本身，降低其他符号对分类准确率的影响。文本繁体转简体是为了方便后续的文本向量化表示，因为本文使用的文本向量化表示方法是调用外部的词向量模型，如果文本中使用的词汇不在词汇表中，则会使当前词汇使用初始化的向量表示方法，改变词汇本身的语义，而多数繁体文本较为复杂，通常都会超出词汇表的范围，而将其转为简体既不会改变其本身的语义，又方便向量化表示，所以文本繁体转简体的步骤十分必要。

2）去除停用词。因为中文短文本中通常存在“的”“吧”“啊”“呃”等高频且无实际意义的词，所以本文将这类词语加入停用词库进行过滤，这样可在一定程度上降低输入文本的特征维度，提高文本分类处理的效率和效果。

3）类别匹配。将原始文本与其对应类别一一匹配，因为本文使用有监督的文本分类算法，所以需要知道每一个样本的特定类别。

4）文本过滤。文本过滤主要包括文本过滤和类别过滤。文本长度过滤是因为本文研究对象为短文本，而短文本通常为不超过200个字符的文本形式，若文本过长，则会超出本文研究范围，所以将此类文本进行过滤。类别过滤是因为有的类别所包含的文本样本过少，不具有研究参考价值，所以将此类别的文本进行整体过滤。

1.2 BERT模型

预处理后的短文本只有再经过一次向量化表示，才能作为分类模型的输入。通常地，短文本向量

化表示是将短文本进行分词，之后针对分词后的短文本进行特征词提取，选取最能代表短文本语义的特征词进行词向量表示，一般使用word2vec模型[12]作为词向量模型，能够将每个特征词都转化为相同形状的多个$1\times k$维的向量，其中k为词向量维数，最后经过拼接的方式将特征词的词向量整合成一个$n\times k$维的向量，其中n为短文本特征词个数。由于word2vec模型进行词向量表示时不能通过上下文语义进行特征词区分，例如“苹果”这个词存在多种语义，如果是“院子里的苹果熟了”，则此时“苹果”表示水果，如果是“苹果公司发布新产品”，则此时“苹果”表示公司名，因此word2vec词向量模型会将这两个短文本中的“苹果”都表示成相同的向量，然而对于分类器而言，这两个词表示相同含义。为解决该问题，本文使用BERT模型替代word2vec模型进行文本语义表示。

1.2.1　BERT模型结构

BERT模型结构[10]如图1所示，其中$E_1, E_2, \cdots, E_N$表示字的文本输入，其经过双向Transformer编码器（Trm模块）得到文本的向量化表示，即文本的向量化表示主要通过Transformer编码器实现。

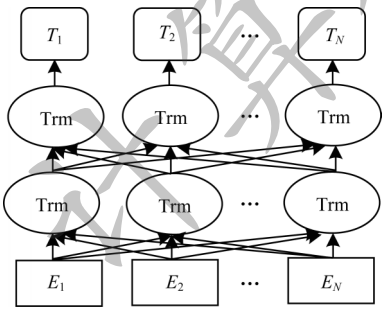

图1　BERT模型结构

Fig.1　Structure of BERT model

Transformer[13]是一个基于Self-attention的Seq2seq模型。Seq2seq是一个Encoder-Deocder结构的模型，即输入和输出均是一个序列，其中，Encoder将一个可变长度的输入序列变为固定长度的向量，Decoder将该固定长度的向量解码为可变长度的输出序列，Seq2seq模型结构如图2所示。

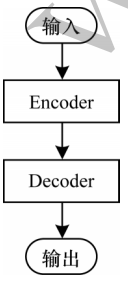

图2　Seq2seq模型结构

Fig.2　Structure of Seq2seq model

通常解决序列问题的Encoder-Decoder结构的核心模块基于RNN实现，但是RNN不能进行并行实现且运行速度慢。为此，Transformer使用Self-attention替代RNN。BERT模型中主要使用Transformer的Encoder部分，具体结构如图3所示。

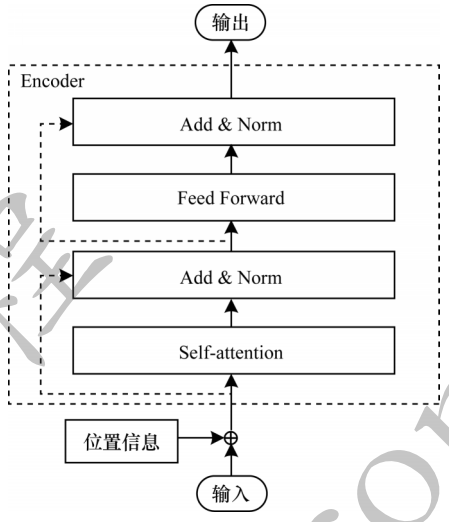

图3　Transformer Encoder结构

Fig.3　Structure of Transformer Encoder

从图3可以看出，Encoder的输入是一句句子的字嵌入表示，然后加上该句子中每个字的位置信息，之后经过Self-attention层，使得Encoder在对每个字进行编码时可以查看该字的前后信息。Encoder的输出会再经过一层Add & Norm层，Add表示将Self-attention层的输入和输出进行相加，Norm表示将相加过的输出进行归一化处理，使得Self-attention层的输出有固定的均值和标准差，其中，均值为0，标准差为1，归一化后的向量列表会再传入一层全连接的前馈神经网络。同样地，Feed Forward层也会经过相应的Add & Norm层处理，之后输出归一化后的词向量列表。Encoder部分中最主要的模块为Self-attention，其核心思想是计算一句句子中每个词与该句子中所有词的相互关系，再利用这些相互关系来调整每个词的权重以获得每个词新的表达方式，该表达方式不但蕴含词本身的语义，还蕴含其与其他词的关系，因此通过该方法获得的向量相比传统词向量具有更加全局的表达方式[14]。

1.2.2　Self-attention计算

假设输入句子$\boldsymbol{X}$，将其按照字粒度进行分字后表示为$\boldsymbol{X}=(\boldsymbol{x}^1, \boldsymbol{x}^2, \cdots, \boldsymbol{x}^N)^{\mathrm{T}}$，$N$表示输入句子中字的个数，将每个字采用One-hot向量[15]表示，设维数为k，则$\boldsymbol{X}$对应的字嵌入矩阵为$\boldsymbol{A}=(\boldsymbol{a}^1, \boldsymbol{a}^2, \cdots, \boldsymbol{a}^N)^{\mathrm{T}}$，其中$\boldsymbol{a}^i$是对应$\boldsymbol{x}^i$的向量表示，是一个$k$维向量，$\boldsymbol{A}$是一个$N\times k$维的矩阵，每一行对应该输入句子中一个字的向量表示。Self-attention计算步骤具体如下：

1）计算Query、Key、Value矩阵[13]，通过模型训练得到：

$\boldsymbol{Q}=\boldsymbol{A}\boldsymbol{W}^{\boldsymbol{Q}},\boldsymbol{K}=\boldsymbol{A}\boldsymbol{W}^{\boldsymbol{K}},\boldsymbol{V}=\boldsymbol{A}\boldsymbol{W}^{\boldsymbol{V}}$

其中：$\boldsymbol{Q}$、$\boldsymbol{K}$、$\boldsymbol{V}$分别为$N\times d_k$、$N\times d_k$、$N\times d_v$维的矩阵，它们的每一行分别对应输入句子中一个字的Query、Key、Value向量，且每个Query和Key向量的维度均为d_k，Value向量的维度为d_v；权重矩阵$\boldsymbol{W}^{\boldsymbol{Q}}$和$\boldsymbol{W}^{\boldsymbol{K}}$的维度均为$k\times d_k$，权重矩阵$\boldsymbol{W}^{\boldsymbol{V}}$的维度为$k\times d_v$。

2）计算Attention[13]：

$$\text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\text{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V}$$

其中，Softmax(·)为归一化指数函数，当其作用于矩阵时，表示对矩阵中的每一个行向量都进行以下运算[16]：

$$\text{Softmax}(z_1,z_2,\cdots,z_N)=\frac{1}{\sum_i^N \mathrm{e}^{z_i}}(\mathrm{e}^{z_1},\mathrm{e}^{z_2},\cdots,\mathrm{e}^{z_N})$$

其中，$(z_1,z_2,\cdots,z_N)$为一个N维行向量，经Softmax(·)函数作用后的行向量元素被等比例压缩至[0,1]，并且压缩后的向量元素和为1。最终得到的Attention值是一个$N\times d_v$维的矩阵，每一行代表输入句子中相应字的Attention向量，该向量已融合其他位置字的信息，是一个全新的向量表示。

由上文计算公式可以看出，整个Self-attention计算过程是一系列矩阵乘法，且可以实现并行运算，运行速度优于RNN。在实际应用过程中，Transformer使用Multi-head Self-attention，即多头Self-attention，多头模式可以增强模型关注能力，head个数即超参数个数[13]，在实际训练模型中可以人为设置。假如本文设置head=2，那么其中一个Self-attention可以更多地关注每个字相邻单词的信息，另一个Self-attention可以更多地关注每个字更远位置的单词信息，然后将这两个Self-attention矩阵进行横向拼接，最后使用一个附加的权重矩阵与该矩阵相乘使其压缩成一个矩阵，计算公式[13]如下：

$$\text{MultiHead}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=$$
$$\text{Concat}(\text{head}_1,\text{head}_2,\cdots,\text{head}_k)\boldsymbol{W}^{O}$$
$$\text{head}_i=\text{Attention}(\boldsymbol{Q}\boldsymbol{W}_i^{\boldsymbol{Q}},\boldsymbol{K}\boldsymbol{W}_i^{\boldsymbol{K}},\boldsymbol{V}\boldsymbol{W}_i^{\boldsymbol{V}})$$

其中：$\boldsymbol{W}_i^{\boldsymbol{Q}}$、$\boldsymbol{W}_i^{\boldsymbol{K}}$、$\boldsymbol{W}_i^{\boldsymbol{V}}$表示第$i$个head的$\boldsymbol{W}^{\boldsymbol{Q}}$、$\boldsymbol{W}^{\boldsymbol{K}}$、$\boldsymbol{W}^{\boldsymbol{V}}$权重矩阵，维度设置与上文相同；$\boldsymbol{W}^{O}$表示附加的权重矩阵，维度为$hd_v\times N$；Concat(·)表示拼接函数。

在上文Self-attention计算过程中没有考虑输入序列中各个单词的顺序，但在自然语言处理中文本的单词顺序是非常重要的信息，例如，“他打了我”和“我打了他”，对应单词完全一样，但是由于单词顺序不同，却表达出完全相反的语义，因此在实际应用中，Transformer将输入字的位置信息加在输入层的字嵌入表示上，即在进入Self-attention层之前，字嵌入表示矩阵已经融合了位置信息。综上所述，BERT模型使用双向Transformer的Encoder可以学习每个单词的前后信息，获得更好的词向量表示。

1.2.3 预训练任务

为增强语义表示能力，BERT模型创新性地提出MLM（Masked LM）和NSP（Next Sentence Prediction）两个预训练任务。

1）MLM任务。给定一句句子，随机掩盖其中的一个或者几个词，用剩余的词去预测掩盖的词。该任务是为了使BERT模型能够实现深度的双向表示，具体做法为：针对训练样本中的每个句子随机掩盖其中15%的词用于预测，例如，“大都好物不坚牢”，被掩盖的词是“坚”，对于被掩盖的词，进一步采取以下策略：

（1）80%的概率真的用[MASK]替代被掩盖的词：“大都好物不坚牢”→“大都好物不[MASK]牢”。

（2）10%的概率用一个随机词去替代它：“大都好物不坚牢”→“大都好物不好牢”。

（3）10%的概率保持不变：“大都好物不坚牢”→“大都好物不坚牢”。

经过上述操作，在后续微调任务的语句中不会出现[MASK]标记，若总使用[MASK]替代被掩盖的词，则会导致模型预训练与后续微调过程不一致。另外，由于当预测一个词汇时，模型并不知道输入的词汇是否为正确的词汇，这使得模型更多地依赖上下文信息预测词汇，因此上述操作赋予模型一定的纠错能力。本文只随机替换1.5%的词为其他词，整体上不会影响模型的语言理解能力。

2）NSP任务。给定一篇文章中的两句句子，判断第二句句子在文章中是否紧跟在第一句句子之后。问答（Question Answering，QA）和自然语言推理（Natural Language Inference，NLI）等重要的自然语言处理下游任务多数是基于理解两个句子之间的关系，因此该任务是为了使BERT模型学习到两个句子之间的关系。具体做法为：从文本语料库中随机选择50%正确语句对和50%错误语句对，若选择A和B作为训练样本时，则B有50%的概率是A的下一个句子，也有50%的概率来自语料库中随机选择的句子，本质上是在训练一个二分类模型，判断句子之间的正确关系。在实际训练过程中，结合NSP任务与MLM任务能够使模型更准确地刻画语句甚至篇章层面的语义信息。

1.3 短文本表示

本文使用BERT模型进行短文本的向量表示，一般的短文本表示流程如图4所示。

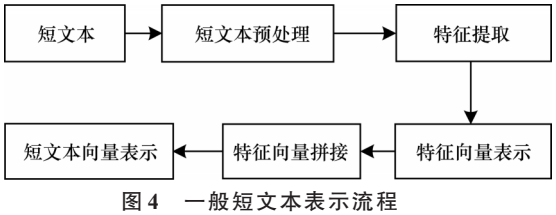

图4 一般短文本表示流程

Fig.4 Procedure of general short text representation

BERT模型的输出有两种形式：一种是字符级别的向量，即输入短文本的每个字符对应的向量表示；另一种是句子级别的向量，即BERT模型输出最左边[CLS]特殊符号的向量，BERT模型认为该向量可以代表整个句子的语义，如图5所示。

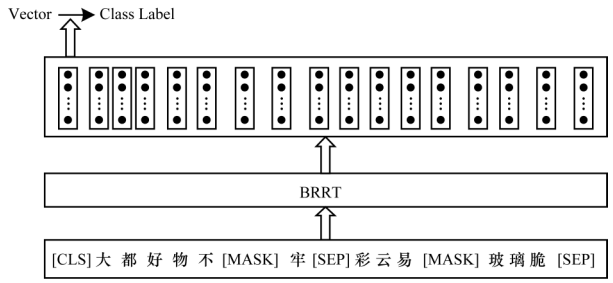

图5　BERT模型输出

Fig.5　Output of BERT model

在图5中，最底端的[CLS]和[SEP]是BERT模型自动添加的句子开头和结尾的表示符号，可以看出输入字符串中每个字符经过BERT模型处理后都有相应的向量表示。当需要得到一个句子的向量表示时，BERT模型输出最左边[CLS]特殊符号的向量，由于本文使用BERT模型的输出，因此相比一般短文本表示流程，无需进行特征提取、特征向量表示及特征向量拼接，具体流程如图6所示。

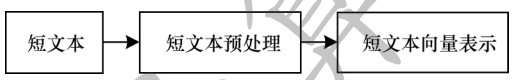

图6　本文短文本表示流程

Fig.6　Procedure of the proposed short text representation

1.4　Softmax回归模型

本文引入Softmax回归模型进行短文本分类。Softmax回归模型是Logistic回归模型在多分类问题中的扩展，属于广义线性模型。假设有训练样本集$\{(\boldsymbol{x}^1,y^1),(\boldsymbol{x}^2,y^2),\cdots,(\boldsymbol{x}^m,y^m)\}$，其中：$\boldsymbol{x}^i\in\mathbb{R}^n$表示第$i$个训练样本对应的短文本向量，维度为$n$，共$m$个训练样本；$y^i\in\{1,2,\cdots,k\}$表示第$i$个训练样本对应的类别，$k$为类别个数，由于本文研究短文本多分类问题，因此$k\geqslant 2$。给定测试输入样本$x$，Softmax回归模型的分布函数为条件概率$p(y=j|x)$，即计算给定样本$x$属于第$j$个类别的概率，其中出现概率最大的类别即为当前样本$x$所属的类别，因此最终分布函数会输出一个$k$维向量，每一维表示当前样本属于当前类别的概率，并且模型将k维向量的和做归一化操作，即向量元素的和为1。因此，Softmax回归模型的判别函数$h_\theta(\boldsymbol{x}^i)$为[17]：

$$h_\theta(\boldsymbol{x}^i)=\begin{bmatrix}p(y^i=1|\boldsymbol{x}^i;\ \boldsymbol{\theta})\\p(y^i=2|\boldsymbol{x}^i;\ \boldsymbol{\theta})\\\vdots\\p(y^i=k|\boldsymbol{x}^i;\ \boldsymbol{\theta})\end{bmatrix}=\frac{1}{\sum_{j=1}^{k}\mathrm{e}^{\boldsymbol{\theta}_j^{\mathrm{T}}\boldsymbol{x}^i}}\begin{bmatrix}\mathrm{e}^{\boldsymbol{\theta}_1^{\mathrm{T}}\boldsymbol{x}^i}\\\mathrm{e}^{\boldsymbol{\theta}_2^{\mathrm{T}}\boldsymbol{x}^i}\\\vdots\\\mathrm{e}^{\boldsymbol{\theta}_k^{\mathrm{T}}\boldsymbol{x}^i}\end{bmatrix}$$

其中：$h_\theta(\boldsymbol{x}^i)$中任一元素$p(y^i=k|\boldsymbol{x}^i;\ \boldsymbol{\theta})$是当前输入样本$\boldsymbol{x}^i$属于当前类别$k$的概率，并且向量中各个元素之和等于1；$\boldsymbol{\theta}$为模型的总参数，$\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_k\in\mathbb{R}^n$为各个类别对应的分类器参数，具体关系为$\boldsymbol{\theta}=[\boldsymbol{\theta}_1^{\mathrm{T}},\boldsymbol{\theta}_2^{\mathrm{T}},\cdots,\boldsymbol{\theta}_k^{\mathrm{T}}]^{\mathrm{T}}$。

Softmax回归模型的参数估计可用极大似然法进行求解，似然函数和对数似然函数分别为：

$$L(\boldsymbol{\theta})=\prod_{i=1}^{m}\prod_{j=1}^{k}\left(\frac{\mathrm{e}^{\boldsymbol{\theta}_j^{\mathrm{T}}\boldsymbol{x}^i}}{\sum_{j=1}^{k}\mathrm{e}^{\boldsymbol{\theta}_j^{\mathrm{T}}\boldsymbol{x}^i}}\right)^{I\{y^i=j\}}$$

$$I(\boldsymbol{\theta})=\ln(L(\boldsymbol{\theta}))=\sum_{i=1}^{m}\sum_{j=1}^{k}I\{y^i=j\}\cdot\ln\frac{\mathrm{e}^{\boldsymbol{\theta}_j^{\mathrm{T}}\boldsymbol{x}^i}}{\sum_{j=1}^{k}\mathrm{e}^{\boldsymbol{\theta}_j^{\mathrm{T}}\boldsymbol{x}^i}}$$

其中，$I\{\cdot\}$为示性函数，$I\{y^i=j\}=\begin{cases}1,y^i=j\\0,y^i\neq j\end{cases}$。

在一般情况下，Softmax回归模型通过最小化损失函数求得$\boldsymbol{\theta}$，从而预测一个新样本的类别。定义Softmax回归模型的损失函数[18]为：

$$J(\boldsymbol{\theta})=-\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{j=1}^{k}I\{y^i=j\}\ln\frac{\mathrm{e}^{\boldsymbol{\theta}_j^{\mathrm{T}}\boldsymbol{x}^i}}{\sum_{l=1}^{k}\mathrm{e}^{\boldsymbol{\theta}_l^{\mathrm{T}}\boldsymbol{x}^i}}\right]$$

其中，m为样本个数，k为类别个数，i表示某个样本，$\boldsymbol{x}^i$是第i个样本x的向量表示，j表示某个类别。

本文使用随机梯度下降法优化上述损失函数，由于在Softmax回归模型中，样本x属于类别j的概率为$p(y^i=j|\boldsymbol{x}^i;\ \boldsymbol{\theta})=\frac{\mathrm{e}^{\boldsymbol{\theta}_j^{\mathrm{T}}\boldsymbol{x}^i}}{\sum_{l=1}^{k}\mathrm{e}^{\boldsymbol{\theta}_l^{\mathrm{T}}\boldsymbol{x}^i}}$，因此损失函数的梯度为

$$\nabla_{\boldsymbol{\theta}_j}J(\boldsymbol{\theta})=-\frac{1}{m}\left(\sum_{i=1}^{m}[\boldsymbol{x}^i(I\{y^i=j\}-p(y^i=j|\boldsymbol{x}^i;\ \boldsymbol{\theta}))]\right)。$$

在梯度下降法的实现过程中，每一次迭代均需要按照$\boldsymbol{\theta}_j=\boldsymbol{\theta}_j-\alpha\nabla_{\boldsymbol{\theta}_j}J(\boldsymbol{\theta})$，$j=1,2,\cdots,k$更新参数。通过上述方法求出$\boldsymbol{\theta}$得到判别函数$\boldsymbol{h}_\theta(\boldsymbol{x}^i)$，即可对输入数据实现预测与分类。

1.5　短文本分类算法

本文提出基于BERT的中文短文本分类算法，具体步骤如下：

算法1　基于BERT的中文短文本分类算法

输入　初始中文短文本训练集$T=\{(\boldsymbol{x}^1,y^1),(\boldsymbol{x}^2,y^2),\cdots,(\boldsymbol{x}^N,y^N)\}$，$i=1,2,\cdots,N$，其中，$\boldsymbol{x}^i$为第$i$个训练样本对应的中文短文本向量，$y^i$为第$i$个训练样本对应的类别

输出　中文短文本分类模型M

步骤1　对训练集T进行预处理得到训练集$T'=\{(\boldsymbol{x}^{1'},y^{1'}),(\boldsymbol{x}^{2'},y^{2'}),\cdots,(\boldsymbol{x}^{N'},y^{N'})\}$，$i=1,2,\cdots,N'$，其中，$\boldsymbol{x}^{i'}$为预处理后的第$i'$个训练样本对应的中文短文本向量，$y^{i'}$为预处理后第$i'$个训练样本对应的类别。

步骤2　使用BERT预处理语言模型在训练集

T'上进行微调，采用BERT模型输出得到训练集T'对应的特征表示$\boldsymbol{V}=(\boldsymbol{v}^1,\boldsymbol{v}^2,\cdots,\boldsymbol{v}^{N'})$，$i=1,2,\cdots,N'$，其中，$\boldsymbol{v}^i$是每条短文本$x^{i'}$对应句子级别的特征向量。

步骤3 将步骤2中得到的特征表示$\boldsymbol{V}$输入Softmax回归模型进行训练，输出中文短文本分类模型M。

2 实验结果与分析

2.1 实验数据

本文实验使用的语料库来自搜狗实验室提供的2012年6月—2012年7月国内、社会、体育、娱乐等18个频道的搜狐新闻数据[19]，选取其中体育、财经、娱乐、IT、汽车和教育6个类别。根据6 000、18 000和30 000的文本数据量设计A、B、C 3组实验，按照8∶1∶1的比例划分训练集、验证集及测试集，并且每次从各类别中随机选择等量的文本数据，如表1所示。

表1 实验数据设置

Table 1 Setting of experimental data

类别	文本数据量		
	A组	B组	C组
体育	1 000	3 000	5 000
财经	1 000	3 000	5 000
娱乐	1 000	3 000	5 000
IT	1 000	3 000	5 000
汽车	1 000	3 000	5 000
教育	1 000	3 000	5 000

2.2 评价指标

本文研究问题属于分类问题，分类问题常用的评价指标为精确率(P)、召回率(R)以及F1值，分类结果的混淆矩阵[20]如表2所示。

表2 分类结果的混淆矩阵

Table 2 Confusion matrix of classification results

真实情况	预测情况	
	正例	反例
正例	真正例(TP)	假反例(FN)
反例	假正例(FP)	真反例(TN)

1)P是指分类器预测为正且预测正确的样本占所有预测为正的样本的比例，计算公式如下：

$$P=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$$

2)R是指分类器预测为正且预测正确的样本占所有真实为正的样本的比例，计算公式如下：

$$R=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$$

3)F1值是P和R的综合指标，一般计算公式[21]如下：

$$F_\beta=\frac{(\beta^2+1)\times P\times R}{\beta^2\times P+R}$$

其中，F_β是基于P和R的加权调和平均，$\beta>0$时表示R对P的相对重要性[21]，通常取$\beta=1$，此时F_β为标准F1值，计算公式如下：

$$\mathrm{F1}=\frac{2\times P\times R}{P+R}$$

其中，$0\leqslant \mathrm{F1}\leqslant 1$，当$P=R=1$时，F1值达到最大值1，此时$P$和$R$均达到100%的理想情况，而由文献[21]可知该情况在实际应用中很难实现，当P高时R通常会偏低，当R高时P通常会偏低，因此在使用F1值评估分类器性能时，其值越接近1，说明分类器性能越好。可见，F1值可以更加全面地反映分类性能，因此本文将其作为衡量分类效果的主要评价指标。

2.3 实验过程

本文选择TextCNN模型[22]作为对照模型进行实验，其利用卷积神经网络对文本进行分类，执行效率高且分类效果较好。TextCNN模型在短文本分类训练过程中使用的分词工具为jieba分词，词嵌入技术为word2vec，训练参数设置如表3所示。

表3 TextCNN模型训练参数设置

Table 3 Training parameter setting of TextCNN model

参数名	参数值
词嵌入维度	100
CNN卷积核个数	100
CNN卷积核大小	5
Dropout随机失活率	0.1
模型迭代次数	30
学习率	0.001
每批训练集的数据量	64

本文使用Google提供的BERT-Base预训练模型。该模型具有12层网络结构，其中隐藏层有768维，采用Multi-head Self-attention(head=12)，并且共有1.1×10^8个参数，训练参数设置如表4所示。

表4 BERT模型训练参数设置

Table 4 Training parameter setting of BERT model

参数名	参数值
Dropout随机失活率	0.1
模型迭代次数	5
学习率	5e-5
每批训练集的数据量	24

2.4 实验结果

本文做了A、B和C 3组实验，数据量逐渐增加，每组实验均使用TextCNN模型作为对比，测试集保持不变，验证集与训练集中的模型参数保持一致，评价指标主要采用F1值，对比结果如图7所示。可以看出，在3组实验中BERT模型与TextCNN模型在6个类别上的F1值均有所差异，说明两个模型在6个类别上的分类性能不同，而且两个模型均在体育、娱乐、汽车和教育类别上表现出优于财经和IT类别的

分类性能，主要原因为这4类新闻数据具有更多的类别区分词，有利于模型学习到更优的类别特征，提高预测能力，这也从侧面反映出文本分类模型的分类性能与文本数据质量具有一定的关系。另外，BERT模型在6个类别上的分类性能均比TextCNN模型效果好，其中BERT模型在A组实验财经类别上的F1值相比TextCNN模型最高提升14个百分点，在教育类别上最低提升3个百分点，而且BERT模型在C组实验的体育类别上的F1值最高可达97%，在B组实验财经类别上的F1值最低为85%，但也高出TextCNN模型5个百分点，验证了本文基于BERT模型的中文短文分类算法的可行性。

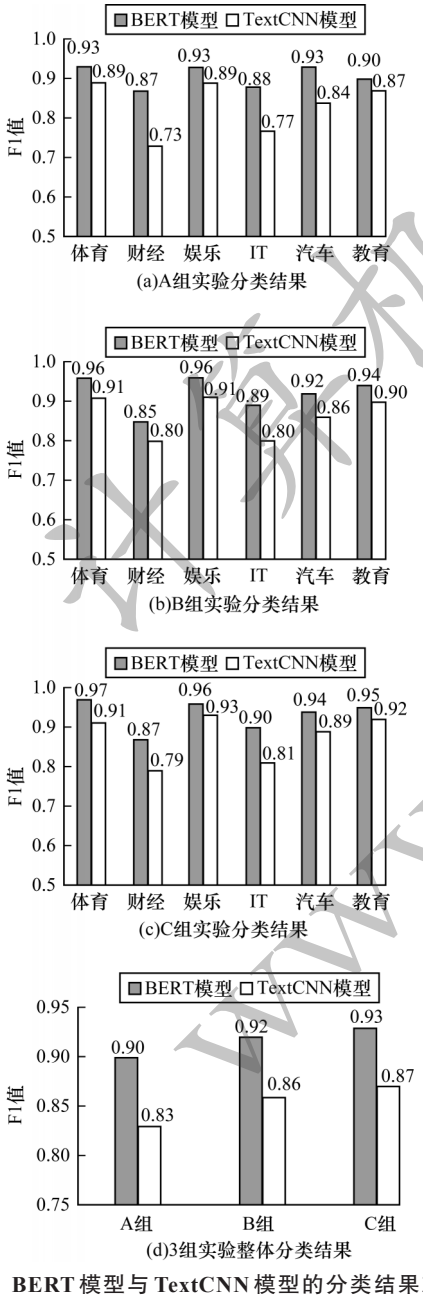

图7　BERT模型与TextCNN模型的分类结果对比

Fig.7　Comparison of classification results between BERT model and TextCNN model

从3组实验整体分类结果可以看出，随着实验数据量的增加，两个模型F1值均有所提高，但总体而言，BERT模型的F1值一直优于TextCNN模型，最高F1值达到93%，相比TextCNN模型提升6个百分点，即使在数据量较少的情况下，BERT模型的F1值也达到90%，说明其相比TextCNN模型能更好地表示短文本层面的语义信息，具有更好的中文短文本分类效果。

3　结束语

本文在解决中文短文本分类的问题时，使用BERT模型替代常用的word2vec模型进行短文本的向量表示，提出一种基于BERT模型的中文短文本分类算法。实验结果表明，该算法在搜狐新闻数据的多个类别上具有较好的分类效果，在体育类别上的F1值最高达到97%，并且随着数据量的增加，在测试集上的整体F1值最高达到93%，相比基于TextCNN模型的中文短文本分类算法提升6个百分点，说明BERT模型具有更好的中文短文本分类效果，对其他处理对象为句子级别的自然语言处理下游任务具有一定的参考价值。后续将在本文算法的句子表征上融入表情、标点符号等位置信息来丰富短文本的句子向量特征表示，进一步提高中文短文本的分类效果。

参考文献

[1] China Internet Information Center. The 43rd statistical report on Internet development in China [EB/OL]. [2019-09-16]. http://www.cnnic.net.cn/hlwfzyj/hlwxzbg/hlwtjbg/201902/t20190228_70645.html. (in Chinese)
中国互联网信息中心. 第43次《中国互联网发展状况统计报告》[EB/OL]. [2019-09-16]. http://www.cnnic.net.cn/ hlwfzyj/hlwxzbg/hlwtjbg/201902/t20190228_70645.html.

[2] WU Yanwen, HUANG Kai, WANG Xinyue, et al. Method of emotional classification in short texts combined with LDA models[J]. Journal of Chinese Computer Systems 2019, 40(10): 2082-2086. (in Chinese)
吴彦文，黄凯，王馨悦，等. 一种融合主题模型的短文本情感分类方法[J]. 小型微型计算机系统，2019，40(10)：2082-2086.

[3] YANG Zhitong, ZHENG Jun. Research on Chinese short text classification based on word2vec [C]// Proceedings of the 2nd IEEE International Conference on Computer and Communications. Washington D. C., USA: IEEE Press, 2019: 90-96.

[4] CHEN Qiaohong, WANG Lei, SUN Qi, et al. Short text classification method of convolutional neural network[J]. Application of Computer Systems, 2019, 28(5): 137-142. (in Chinese)
陈巧红，王磊，孙麒，等. 卷积神经网络的短文本分类方法[J]. 计算机系统应用，2019，28(5)：137-142.

[5] LIU Xiaomin, WANG Hao, LI Xinlei, et al. A comparative study on the role of different feature granularity in short text classification of Weibo[J]. Information Science, 2018, 36(12): 126-133. (in Chinese)

刘小敏,王昊,李心蕾,等. 不同特征粒度在微博短文本分类中作用的比较研究[J]. 情报科学,2018,36(12):126-133.

[6] WANG Lei. Research on Chinese short text classification method based on hybrid neural network[D]. Hangzhou: Zhejiang Sci-Tech University,2019. (in Chinese)
王磊. 基于混合神经网络的中文短文本分类方法研究[D]. 杭州:浙江理工大学,2019.

[7] FENG Yong, QU Bohao, XU Hongyan, et al. Chinese FastText short text classification method based on TF-IDF and LDA [J]. Journal of Applied Sciences, 2019,37(3):378-388. (in Chinese)
冯勇,屈渤浩,徐红艳,等. 融合TF-IDF和LDA的中文FastText短文本分类方法[J]. 应用科学学报,2019,37(3):378-388.

[8] WU Fenlin. Adaptive normalized weighted KNN text classification based on PSO [J]. Scientific Bulletin of National Mining University,2016(1):109-115.

[9] GAO Yunlong,ZUO Wanli,WANG Ying,et al. Short text classification model based on integrated neural network[J]. Journal of Jilin University(Science Edition),2018,56(4):933-938. (in Chinese)
高云龙,左万利,王英,等. 基于集成神经网络的短文本分类模型[J]. 吉林大学学报(理学版),2018,56(4):933-938.

[10] DEVLIN J, CHANG M W, LEE K, et al. BERT: pre-training of deep bidirectional transformers for language understanding [EB/OL]. [2019-09-16]. https://arxiv.org/abs/1810.04805.

[11] SUN Zhaoying,LIU Gongshen. Research on neural network clustering algorithm for short texts[J]. Computer Science, 2018,45(S1):392-395. (in Chinese)
孙昭颖,刘功申. 面向短文本的神经网络聚类算法研究[J]. 计算机科学,2018,45(S1):392-395.

[12] ZHANG Dongwen, XU Hua, SU Zengcai, et al. Chinese comments sentiment classification based on word2vec and SVMperf [J]. Expert Systems with Applications, 2015,42(4):1857-1863.

[13] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need [EB/OL]. [2019-09-16]. https://arXiv:1706.03762v5.

[14] YANG Piao, DONG Wenyong. Chinese named entity recognition method based on BERT embedding [J]. Computer Engineering,2020,46(4):40-45,52. (in Chinese)
杨飘,董文永. 基于BERT嵌入的中文命名实体识别方法[J]. 计算机工程,2020,46(4):40-45,52.

[15] BRAUD C, DENIS P. Comparing word representations for implicit discourse relation classification[C]//Proceedings of 2015 Conference on Empirical Methods in Natural Language Processing. Lisbon, Portugal: Association for Computational Linguistics,2015:1-8.

[16] LI Li, YING Sancong. Implementation of Softmax layer of FPGA-based convolutional neural networks[J]. Modern Computer(Professional Edition), 2017(26):21-24. (in Chinese)
李理,应三丛. 基于FPGA的卷积神经网络Softmax层实现[J]. 现代计算机(专业版),2017(26):21-24.

[17] YANG Sen. Application research of credit scoring model for small and micro enterprises based on Softmax regression[D]. Suzhou: Soochow University, 2017. (in Chinese)
杨森. 基于Softmax回归的小微企业信用评分模型应用研究[D]. 苏州:苏州大学,2017.

[18] LI Ran. Research on short text emotional tendency based on deep learning [D]. Beijing: Beijing Institute of Technology,2015. (in Chinese)
李然. 基于深度学习的短文本情感倾向性研究[D]. 北京:北京理工大学,2015.

[19] Sogou Lab Data. Sohu news data(SogouCS)[EB/OL]. [2019-09-16]. http://www.sogou.coms/labs/resource/cs.php. (in Chinese)
搜狗实验室数据. 搜狐新闻数据(SogouCS)[EB/OL]. [2019-09-16]. http://www.sogou.coms/labs/resource/cs.php.

[20] LI Y X, TAN C L, DING X Q, et al. Contextual post-processing based on the confusion matrix in offline handwritten Chinese script recognition [J]. Pattern Recognition,2004,37(9):1901-1912.

[21] ZHOU Zhihua. Machine learning[M]. Beijing: Tsinghua University Press,2016. (in Chinese)
周志华. 机器学习[M]. 北京:清华大学出版社,2016.

[22] KIM Y. Convolutional neural networks for sentence classification[EB/OL]. [2019-09-16]. https://arxiv.org/abs/1408.5882.

编辑 陆燕菲