文章编号：2095-2163(2019)02-0021-07 中图分类号：TP391.1 文献标志码：A

基于 BiLSTM 神经网络的特征融合短文本分类算法

和志强，杨 建，罗长玲
（河北经贸大学 信息技术学院，石家庄 050061）

摘 要：由于短文本自身具有词汇个数少且格式不规范的特点，造成神经网络输入矩阵存在特征稀疏、维度过高以及语义特征提取不充分等问题。为解决上述问题，提出一种基于双向长短时记忆神经网络的短文本分类算法（WTL-BiLSTM），该算法融合 Word2vec、TF-IDF 和 LDA 主题模型实现文本向量化，在获取短文本词义特征的同时，加入词汇重要程度特征和文本主题特征。并利用 BiLSTM 从前、后两个方向全面捕捉短文本语义特征，有效避免了 RNN 模型梯度爆炸和梯度消失问题。经实验验证，该算法能够有效解决短文本分类过程中出现的问题，相比于传统的短文本分类算法，分类准确率得到一定程度的提升。

关键词：BiLSTM；Word2vec 模型；短文本分类

Combination characteristics based on BiLSTM for short text classification

HE Zhiqiang, YANG Jian, LUO Changling
（Information Technology College, Hebei University of Economics & Business, Shijiazhuang 050061, China）

【Abstract】Due to the characteristics that the short text itself has a small number of vocabulary and format is not standardized, the input matrix of neural network has problems such as sparse features, high dimension and insufficient extraction of semantic feature. In order to solve the above problems, this paper proposes a short text classification algorithm (WTL-BiLSTM) based on bidirectional long short-term memory cyclic neural network. This algorithm combines Word2vec model, TF-IDF model and LDA theme model to implement text vectorization. At the same time, it acquires the importance of vocabulary features and text topic features while acquiring short text semantic features. The algorithm uses BiLSTM to capture the short text semantic features from the front and back directions, effectively avoiding the gradient explosion and gradient disappearance of the RNN model. The experimental results show that the designed classification algorithm can effectively solve the problems in the short text classification process. Compared to the traditional short text classification algorithm, the classification accuracy rate is improved to a certain degree.

【Key words】BiLSTM; Word2vec model; short text classification

0 引 言

随着互联网技术的快速发展，信息以文本、图片、视频、音频等多种方式进行广泛传播，在众多传播方式中文本仍旧是人们获取信息的主要方式[1]。目前，文本信息的表现方式已呈现出多样化，迄今为止则衍生出包括微博、新闻简讯、标题摘要、事件评论等在内的多种类型，与之前长文本相比，具有词数少、数量大、实时性强等特点。网络上海量短文本信息与当前社会的热点事件紧密相关，对这些信息进行有效分类管理，有助于政府、企业了解政治、经济、文化等领域发生的最新变化，更好地做到舆情疏导、危机公关、产品营销等[2]。因此，这些海量短文本数据极具研究价值，如何实现对海量短文本高效、准确分类并获得有价值信息，已成为学界亟待解决的研究课题。

目前，针对短文本分类问题的研究，主要可分为 2 个方面，即：文本向量化表示和分类模型。其中，文本向量化表示重点包括向量空间模型（Vector Space Model, VSM）[3]和基于文本分布式表示两种方法。前者是基于文本表层信息的提取，只限于对词频的简单统计、计算，常存在特征矩阵稀疏、维度高、词汇鸿沟等问题[4]。而后者是对文本深层信息的提取，根据词汇的概率分布获取词汇语义信息。文献[5]利用大规模语料训练 Word2vec 模型，学习文本中词汇间潜在的语义关联，在词汇粒度层面提取特征，得到词向量，解决矩阵稀疏、无语义特征问题。文献[6-8]利用 LDA 主题模型提取文章主题，在文本粒度层面扩展文本特征。文献[9-10]在 Word2vec 向量的基础上，结合 TFIDF 算法对词向量进行加权，以考虑词汇在不同类别中的重要程度。

作者简介：和志强（1972-），男，博士，教授，主要研究方向：数据挖掘、高速数据采集；杨 建（1992-），男，硕士研究生，主要研究方向：数据挖掘、高速数据采集；罗长玲（1993-），男，硕士研究生，主要研究方向：数据挖掘、高速数据采集。

收稿日期：2018-12-20

文本分类模型中,深度学习模型的分类效果一般要优于传统的机器学习算法。总地来说,卷积神经网络(Convolutional Neural Network, CNN)就常常用来获取邻近词汇间的关联,注重捕捉文本局部特征[11]。循环神经网络(Recurrent Neural Network, RNN)作为一种序列模型,可以由前向后读取文档词汇,并可以对获取的语义信息进行记忆,因此 RNN 模型可以在获取文本局部特征的基础上捕捉到更广范围内的特征信息。相比于 CNN 模型,RNN 模型更适合处理短文本这种序列化数据,然而 RNN 模型在实际处理文本数据过程中,常出现梯度爆炸和梯度消失的问题。为解决梯度消失问题,文献[12]对 RNN 模型做出改进,提出了长短时记忆模型(Long and short time memory model, LSTM)。类似 RNN,LSTM 模型也按照文档输入顺序学习词汇间的特征信息。文档中词汇的正确理解要依赖于上下文信息,因此文献[13]提出双向循环神经网络处理短文本数据能够更充分抓取上下文关联信息。

在文本向量化过程中通常只单独考虑词汇的词义特征或者文本的语义特征,然而词义和语义概念不同,词义代表单个词汇的意思,而语义是多个词汇按照一定顺序综合表达的意思[14],因此为了丰富文本特征信息,文章对短文本向量化表示方法加以改进,结合 Word2vec、LDA 与 TF-IDF 三种模型,分别在词汇粒度和文本粒度两个层面提取文本特征,并计算词汇 TF-IDF 值,对多维词汇向量进行加权,以表现词汇的重要程度。将 3 种算法获取的文本特征组合成为向量矩阵。另外,利用双向 LSTM 作为分类模型,既能解决 RNN 梯度消失问题,又能从 2 个方向充分捕捉上下文特征信息。

1 改进的短文本分类算法 WTL-BiLSTM

1.1 算法流程

文章设计的短文本分类模型将 Word2vec、TF-IDF 与 LDA 三种模型提取的特征向量组合成输入矩阵,输入到 BiLSTM 中利用隐藏层节点获取短文本内部的关联关系,最后利用 softmax 分类器对神经网络层的输出进行类别判定。主要包括文本预处理、向量化表示、神经网络层及分类与评估四部分,算法设计流程如图 1 所示。

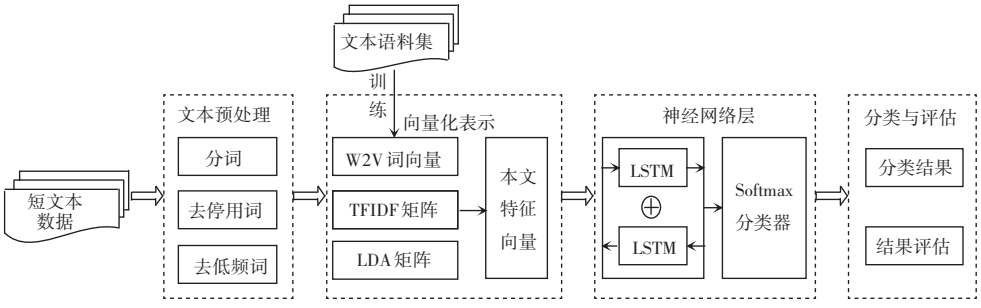

图 1 算法流程图

Fig. 1 Algorithm flow chart

1.2 文本预处理

文本预处理是文本分类过程中基础、且重要的一项操作,训练数据的质量对分类准确程度有很大影响。本次设计的文本预处理主要包括:分词、去停用词与去低频词。

研究可知,中文分词不同于英文分词,词与词之间没有明显的界限,本文利用 jieba 分词软件将文本切割为若干个有意义的词语,并使用含有 36 000 余个常用词条的分词字典以提高分词的正确率。文本分词得到的词组中通常包括一些高频但无意义的词语、标点符号,如“一会儿”、“的”、“了”、“。”等,这些被统称为停用词。为保证文本特征的有效性,利用停用词表过滤掉这些干扰项,保留有价值的特征项;词组还包括部分出现频率低、对分类结果影响小的低频次分项,考虑到处理的数据为短文本,所以去掉出现次数为 1 的词汇,在一定程度上降低了特征维度。

1.3 文本向量化

文本向量化指的是将一篇文档表示为一个向量或矩阵的形式,主要是基于词的向量化[15]。利用大规模语料训练 Word2vec 模型,将每个单词表示为 n 维向量,再结合 TF-IDF 算法、LDA 主题模型提取的词汇权重向量和文档主题向量,实现文本的向量化表示。对此可展开研究论述如下。

1.3.1 Word2vec 模型

Word2vec 是一种轻量级神经网络,能够简单、高效地将词语映射为一个低维、稠密的向量。语料集则构成一个巨大的词向量空间,每个词向量映射

为其中的一个点，此后则可通过计算向量点间的距离来判断词汇间的相似性。该模型包括：CBOM（Continuous Bag－of－Word Model）和 Skip－gram（Continuous Skip-gram Model）两种训练方式，两者都包括输入层、投影层和输出层三层网络，主要区别在于：前者通过上下文信息预测当前词汇，而后者是通过当前词汇信息预测上下文，两者网络结构分别如图 2、图 3 所示。

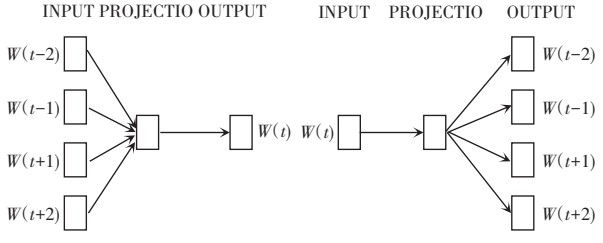

图 2　CBOM 网络结构

图 3　Skip-gram 网络结构

Fig. 2　The network structure of CBOM

Fig. 3　The network structure of Skip-gram

由于短文本中词汇数量较少，上下文信息缺失，故本次设计使用 Skip-gram 方式来获取包含上下文词义信息的词向量，训练参数见表 1。设置窗口大小为 5，语料库中包括 27 243 978 个词汇，经训练得到 167 299 个 100 维的词向量。以“伊拉克”为例，查看与其词义相关联的词汇集合如图 4 所示，可以看出训练结果良好。

表 1　Word2vec 训练参数

Tab. 1　The training parameters of Word2vec

训练方式	窗口大小	向量维度	语料词汇个数	词向量个数	训练时间/s
Skip-gram	5	100	27 243 978	167 299	641

图 4　Word2vec 模型词义相关度

Fig. 4　Word relevance of Word2vec model

Skip-gram 预测上文词汇概率的表示公式为：

$$P(w_{n-c}, w_{n-c+1}, \cdots, w_{n+c-1}, w_{n+c} \mid w_n), \quad (1)$$

其中，w_n 表示语料集中的一个词汇；c 为滑动窗口的大小；$w_{n-c}, w_{n-c+1}, \cdots, w_{n+c-1}, w_{n+c}$ 为根据 w_n 预测的前后 $2*c$ 个词汇。

假设文档 d_m 为文档集 $D = \{d_1, d_2, \cdots, d_m, \cdots, d_M\}$ 中的任意一篇文档，d_m 有词汇集合 $W = \{w_1, w_2, \cdots, w_n, \cdots, w_N\}$，利用 Word2vec 训练得到的词向量对文档 $\boldsymbol{d}_m$ 进行文本向量化，文档 $\boldsymbol{d}_m$ 转化为一个 $N*V$ 的二维矩阵，文档集 $\boldsymbol{D}$ 则表示为一个 $M*N*V$ 的三维矩阵。由此可得对应的数学表述如下：

$$\boldsymbol{d}_m = \begin{bmatrix} w_{11}, w_{21}, \cdots w_{n1}, \cdots, w_{N1} \\ w_{12}, w_{22}, \cdots w_{n2}, \cdots, w_{N2} \\ \vdots \quad \vdots \\ w_{1V}, w_{2V}, \cdots w_{nV}, \cdots, w_{NV} \end{bmatrix}, \quad (2)$$

$$\boldsymbol{D} = [d_1, d_2, \cdots, d_m, \cdots, d_M], \quad (3)$$

其中，M 表示文档集中文档总数；N 表示每篇文档的词汇数量；V 表示 Word2vec 模型训练得到的词向量的维度。

Word2vec 模型将词汇映射为包含上下文词义信息的 V 维词向量，很好地解决了 one－hot 编码方式出现的词汇鸿沟问题，而且有效降低了特征维度，避免了维度灾难问题 。然而 Word2vec 模型无法区分词汇在文档中的重要程度，所以接下来研究将利用 TF-IDF 算法计算词汇的权重，为词向量进行加权。

1.3.2　TF-IDF 算法

TF-IDF 模型广泛应用于信息检索、搜索引擎中，其主要思想为：若某一词汇 w_n 在一类文档 $D_i (D_i \subseteq D)$ 有很高的出现频率，而在其它类文档中很少出现，则认为该词汇能够代表该类文章，具有良好的类别区分能力。该模型主要包括：词频（Term frequency，*TF*）和逆文档频率（Inverse document frequency，*IDF*）两部分。这里，*TF* 表示某个词 w_n 在文档 $\boldsymbol{d}_m$ 中出现的频率，*IDF* 则代表该词的类别区分度，其数学公式可表示为：

$$TF(w_n, \boldsymbol{d}_m) = \frac{f_{n,m}}{\sum_{i=1}^{N} f_{i,m}}, \quad (4)$$

$$IDF(w_n, \boldsymbol{d}_m) = \log\left(\frac{|\boldsymbol{D}|}{|\boldsymbol{D}_{w_n}| + 0.01}\right), \quad (5)$$

其中，$f_{n,m}$ 表示词 w_n 在文档 $\boldsymbol{d}_m$ 中出现的次数；$\sum_{i=1}^{N} f_{i,m}$ 表示文档 $\boldsymbol{d}_m$ 中出现的词汇总数；$|\boldsymbol{D}|$ 为文

档集包含的文档数量；$|\boldsymbol{D}_{w_n}|$ 表示出现词 w_n 的文档数量，为避免因语料集中不包括词 w_n，出现分母为零的情况，为分母添加一个常量，同时减小常量的影响，故分母设置为 $|\boldsymbol{D}_{w_n}|+0.01$。

TFIDF 权重即为 *TF* 与 *IDF* 的乘积。总结来说，词 w_n 的重要性与其在文档中的出现频率成正比增加，同时也与其在语料集中的出现频率成反比下降。经计算可得文档集的权重矩阵的数学公式如下：

$$\boldsymbol{TFIDF}_{M\times N}=\begin{bmatrix} tf_{11},tf_{12},\cdots,tf_{1N} \\ tf_{21},tf_{22},\cdots,tf_{2N} \\ \vdots \quad\quad \vdots \\ tf_{M1},tf_{M2},..,tf_{MN} \end{bmatrix}. \tag{6}$$

1.3.3 LDA 主题模型

LDA 是一种无监督学习的主题概率生成模型，其核心思想为：每篇文档含有若干个隐含主题，每个隐含主题又包含一个与该主题相关的词汇集合。LDA 主题模型主要包括 2 个多项式分布：文档-主题分布 θ 和主题-词汇分布 φ。前者表示各个主题在文档中出现的概率分布；后者表示每个词汇在主题中出现的概率分布。该模型的设计结构如图 5 所示，其中 α、β、K 分别表示：文档中主题分布的先验分布（Dirichlet 分布）的参数、主题中词汇分布的先验分布（Dirichlet 分布）的参数、主题个数。这 3 个参数均需手动设置。

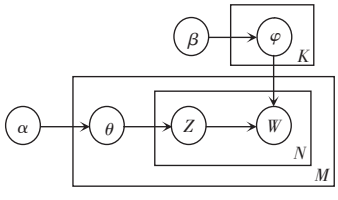

图 5 LDA 模型结构

Fig. 5 Structure of LDA model

短文本集经 LDA 主题模型训练后，输出文档-主题矩阵、主题-词汇矩阵[6]，以概率形式体现文档隐含的语义特征，是对文档深层特征的直接提取。经实验检验，本次设计 LDA 主题模型的参数设置见表 2。

表 2 LDA 主题模型参数

Tab. 2 Parameters of LDA topic model

β	α	K	*iter*
0.01	$1/K$	100	1 000

其中，*iter* 为迭代次数，主题数量 K 设置与 Word2vec 的维度相同，以便进行模型组合，最后得到文档 - 主题矩阵向量 $\boldsymbol{L}$，用此表示语料库中每篇文档 K 个主题的概率分布。研究推得其数学运算公式如下：

$$\boldsymbol{L}_{M\times K}=\begin{bmatrix} lf_{11},lf_{12},..,lf_{1K} \\ lf_{21},lf_{22},..,lf_{2K} \\ \vdots \quad\quad \vdots \\ lf_{M1},lf_{M2},..,lf_{MK} \end{bmatrix}. \tag{7}$$

1.3.4 模型组合

改进的短文本向量化表示方法主要对 3 种模型提取的文本特征进行组合，首先利用计算的 *TFIDF* 值对 Word2vec 词向量进行加权，再将 Word2vec 词向量与 LDA 主题向量进行拼接构成新的向量矩阵，使其包含词汇语义信息的同时又包含文档的主题信息，则文档集中文档 m 的向量化表示则可写作如下形式：

$$\boldsymbol{WTL}_m=\boldsymbol{d}_m\times\boldsymbol{TFIDF}_m\oplus\boldsymbol{L}_m, \tag{8}$$

$$\boldsymbol{WTL}_m=\begin{bmatrix} w_{11}*tf_{11},w_{21}*tf_{12},..,w_{N1}*tf_{1N} \\ w_{12}*tf_{11},w_{22}*tf_{12},..,w_{N2}*tf_{1N} \\ \vdots \quad\quad\quad \vdots \\ w_{1V}*tf_{11},w_{2V}*tf_{12},\cdots,w_{NV}*tf_{1N} \\ lf_{11},\quad lf_{12},\quad \cdots,\quad lf_{1K} \end{bmatrix}. \tag{9}$$

1.4 双向 LSTM 神经网络

自然语言是典型的序列数据，RNN 目前已经常应用于序列数据的处理和预测。RNN 能够记忆之前学习的信息，并利用之前的信息对当前输出施加影响，刻画一个当前输出与之前信息的关系。典型的 RNN 结构及其时序展开结构如图 6 所示。

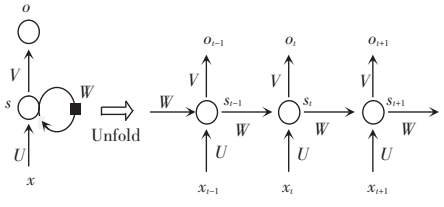

图 6 RNN 结构图

Fig. 6 Structure of RNN

由图 6 可以看出，当前 t 时刻 RNN 的输入除了来自输入层的 x_t，还包括之前时刻隐藏层的状态 s_{t-1}，而当前时刻隐藏层的状态 s_t，不仅作为当前时刻输出层的输入，还被 RNN 记忆下来作为下一时刻隐藏层 s_{t+1} 的输入，RNN 正是基于以上原理对序列数据实现处理。理论上该神经网络模型可以处理无限长的序列，然而在实际应用中训练 RNN 时容易出现梯度爆炸和梯度消失的问题，使得 RNN 无法应对长距离的影响，导致其处理长序列的结果并不理想。针对研究中可能面临的梯度问题，一般通过设置梯

度阈值来解决梯度爆炸问题，当梯度值超过该阈值时直接进行截取；而对于梯度消失问题，则流行使用对 RNN 做出改进的 LSTM。

LSTM 在 RNN 的基础上新增一种自我连接的 CEC（Constant Error Carrousel）单元来记忆长距离信息，并引入门（gate）机制来对输入、输出信息进行限制管理。门实际上是一层全连接层，其输入为一个向量，输出一个取值范围在[0,1]之间的实数向量，LSTM 的设计结构如图 7 所示。

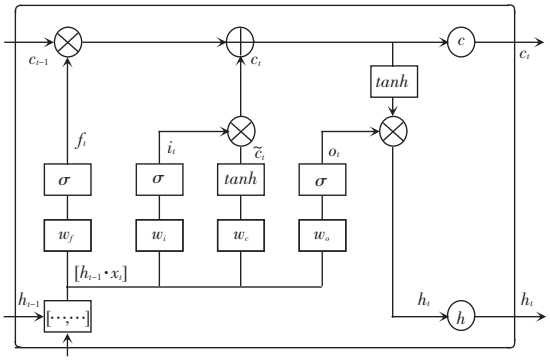

图 7　LSTM 结构图

Fig. 7　Structure of LSTM

由图 7 可知，LSTM 主要包括记忆单元 c、输入门 i、遗忘门 f 以及输出门 o。其中，遗忘门 f 和输入门 i 用来控制记忆单元的输入信息，决定之前时刻单元状态 C_{t-1} 和当前时刻网络输入 x_t 的保留比例。输出门 o 用来控制单元状态 C_t 输入到当前时刻输出值 h_t 的信息量。LSTM 工作过程主要涉及各符号的阐释解读见表 3。其中将用到如下数学公式：

表 3　符号解释说明表

Tab. 3　Symbolic interpretation

符号	说明
c_{t-1}	$t-1$ 时刻记忆单元状态
h_{t-1}	$t-1$ 时刻网络输出值
c_t	t 时刻记忆单元状态
h_t	t 时刻网络输出值
$\tilde{c}_t$	t 时刻记忆单元的候选值
x_t	t 时刻网络输入值
$[\cdot,\cdot]$	向量拼接符
$\boldsymbol{W}_f$, $\boldsymbol{W}_i$, $\boldsymbol{W}_0$	遗忘门、输入门和输出门的权重矩阵
$\boldsymbol{W}_c$	候选值的权重矩阵
b_f, b_i, b_0	遗忘门、输入门和输出门的偏置项
σ	sigmoid 激活函数
$tanh$	反正切激活函数
$\circ$	按元素乘

$$f_t = \sigma(\boldsymbol{W}_f \cdot [h_{t-1}, x_t] + b_f)\ ; \tag{10}$$

$$i_t = \sigma(\boldsymbol{W}_i \cdot [h_{t-1} \cdot x_t] + b_i)\ ; \tag{11}$$

$$\tilde{c}_t = tanh(\boldsymbol{W}_c \cdot [h_{t-1} \cdot x_t] + b_c)\ ; \tag{12}$$

$$C_t = f_t \circ C_{t-1} + i_t \circ \tilde{C}_t ; \tag{13}$$

$$o_t = \sigma(\boldsymbol{W}_o \cdot [h_{t-1} \cdot x_t] + b_o)\ ; \tag{14}$$

$$h_t = o_t \circ tanh(c_t)\ . \tag{15}$$

通常情况下，文档中每个词汇语义的正确理解不但依赖于之前的元素，而且还与之后的元素密切相关。比如下面这句话："我的硬盘坏了，我想____一个新硬盘。"，要补全这句话，若只考虑前面的元素信息，则横线处可以填："修一下"、"买一个"、"扔掉"等，具有很大的不确定性。若同时考虑横线前后的元素信息，则选择填"买"的概率较大。LSTM 为单向神经网络只能从前往后传输状态信息，而不能获取后文对当前词汇的影响，因此，本次设计使用 2 个方向相反的双向 LSTM 来充分捕捉词汇的上下文信息，最大限度理解当前词汇的语义信息，其网络结构如图 8 所示。

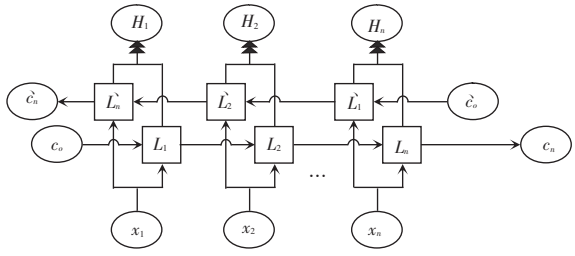

图 8　BiLSTM 结构图

Fig. 8　Structure of BiLSTM

可以看出，BiLSTM 神经网络最后输出 H_n 由 2 个单向、反向的 LSTM 输出结果拼接得到，其拼接公式为：

$$H_n = \grave{h}_n \oplus h_n . \tag{16}$$

2　实验结果分析

2.1　实验数据

本文利用 22 万余条新闻简讯作为实验数据，数据集中涉及 13 个栏目的分类新闻，从中选择 affairs、economic、education、game 等 8 类新闻，并将其以 8:2 的比例划分为训练集和测试集。经统计部分文档，发现分词后大部分文档由 100~200 个词汇组成，文档词汇数目分布如图 9 所示，在完成预处理操作后文档保持在 100 词左右，故本次设计对文档长度进行标准化，保证每篇文档词汇数目为 100，超出部分进行截断，不足则补 0。

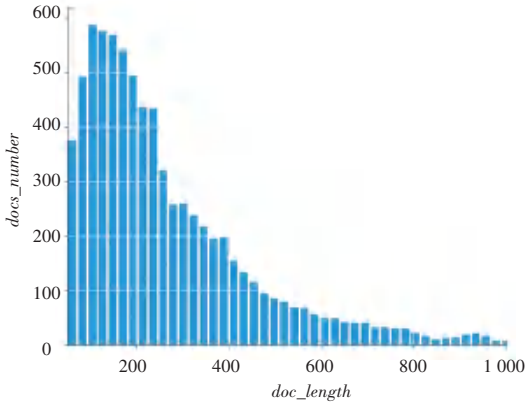

图 9 文档词汇数目分布图

Fig. 9 Document vocabulary number distribution

2.2 实验参数

实验参数的合理设置对实验结果有直接影响，Word2vec 模型与 LDA 主题模型的训练参数已在上文列出，神经网络部分的参数设置见表 4。参数调整过程中使用固定参数的方法，分别对 LSTM 层单元数、Dense 层单元数、丢弃率等参数进行了对比实验，并利用 EarlyStopping 机制对 *val_acc* 值进行监控，当准确率不再上升时则停止训练，以避免过拟合、不收敛等问题，同时可以加快学习速度，提高调参效率。

表 4 神经网络参数设置

Tab. 4 Neural network parameter setting

参数	参数值
LSTM 层单元数	64
Dense 层单元数	40
丢弃率（*dropout*）	0.5
批尺寸（*batch_size*）	500
训练次数（*epoch_size*）	35
优化器（optimizer）	adam

2.3 实验结果及分析

为验证 WTL-BiLSTM 模型的有效性，分别与基于词向量的 LSTM 模型（Word2vec LSTM，W-LSTM）、双向 LSTM 模型（Word2vec BiLSTM，W-BiLSTM）及传统机器学习算法 SVM 进行如下的对比实验研究。

首先与基于神经网络的分类算法进行对比，实验中各模型的参数设置保持一致。图 10～图 15 分别为 W-LSTM、W-BiLSTM、WTL-BiLSTM 的损失函数变化趋势和分类准确率，利用 Python 绘图模块 matplotlib 绘制得到。在图 10～图 15 中，train_～和 val_～分别表示训练集和测试集。对比三者损失函数变化图发现，基于词向量的双向 LSTM 模型（W-BiLSTM）下降到稳定值的速度最快，文章设计的模型（WTL-BiLSTM）下降到稳定值的速度稍慢，但相对其它 2 种模型最后训练得到的损失函数值最小，即说明分类效果最优。通过对比分类准确率图也可以发现，3 种模型分类准确率从高到低依次为：WTL-BiLSTM 模型、W-BiLSTM 模型和 W-LSTM 模型。

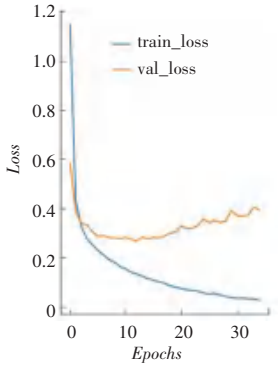

图 10 W-LSTM 损失函数变化图

Fig. 10 Change graph of W-LSTM loss function

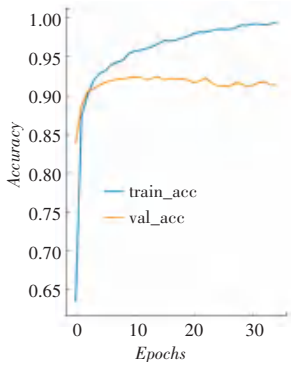

图 11 W-LSTM 分类准确率

Fig. 11 Classification accuracy of W-LSTM

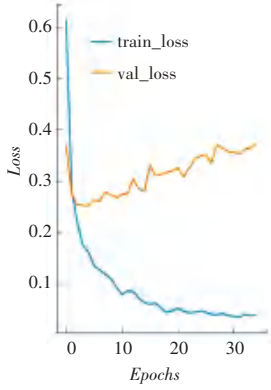

图 12 W-BiLSTM 损失函数变化图

Fig. 12 Change graph of W-BiLSTM loss function

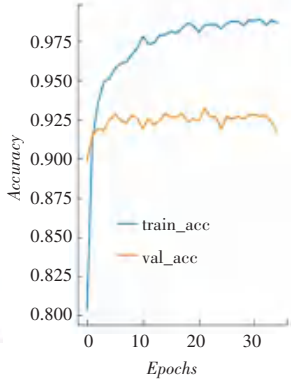

图 13 W-BiLSTM 分类准确率

Fig. 13 Classification accuracy of W-BiLSTM

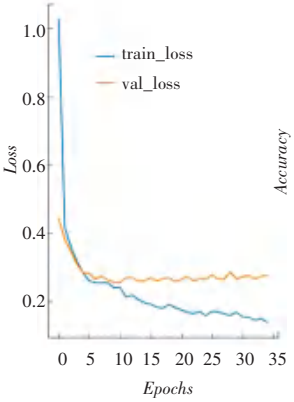

图 14 WTL-BiLSTM 损失函数变化图

Fig. 14 Change graph of WTL-BiLSTM loss function

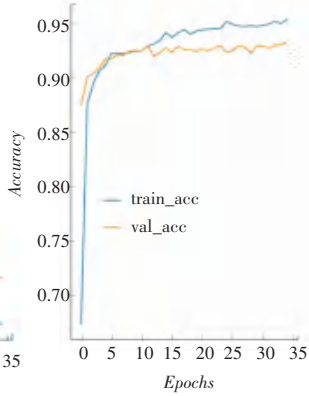

图 15 WTL-BiLSTM 分类准确率

Fig. 15 Classification accuracy of WTL-BiLSTM

本次实验还与 SVM 算法进行对比，实验思路按照文献[9]的处理方法，利用 TFIDF 权重对 Word2vec 模型训练的词向量进行加权，然后将文档中的加权词向量累加后作为文档向量，最后输入到 SVM 中实现分类，算法运行后的分类准确率对比见表 5。根据分类准确率对比发现，本次设计模型的分类效果要远远优于传统机器学习算法。

表 5　各分类模型分类准确率

Tab. 5　Classification accuracy of each classification model

分类模型	分类准确率/%
SVM	81.62
W-LSTM	89.94
W-BiLSTM	91.69
WTL-BiLSTM	93.63

3　结束语

文章提出一种基于双向 LSTM 神经网络的短文本分类算法，从文本向量化表示和分类模型两方面分别提供了改进。文本向量化表示融合 Word2vec 模型、TFIDF 模型与 LDA 主题模型，丰富了文本特征信息。分类模型则采用双向的 LSTM 从词汇前后两个方向充分提取上下文信息。通过实验对比发现，文章提出算法的分类效果要优于只利用 Word2vec 模型进行文本向量化的 LSTM 模型、BiLSTM 模型。另外还与传统机器学习算法 SVM 进行了比较，本次设计的算法同样优于 SVM 模型，进一步验证了本次设计改进文本向量化表示和分类模型的有效性。

参考文献

[1] 张敬谊，张亚红，李静. 基于词向量特征的文本分类模型研究[J]. 信息技术与标准化，2017(5)：71-75.

[2] 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(6A)：392-395.

[3] ZHANG X，ZHU S，LIANG W. Detecting spam and promoting campaigns in the twitter social network[C] // 2012 IEEE 12th International Conference on Data Mining (ICDM). Brussels，Belgium：IEEE，2012：1194-1199.

[4] 刘金岭. 基于主题的中文短信文本分类研究[J]. 计算机工程，2010，36(4)：30-32.

[5] BOJANOWSKI P，GRAVE E，JOULIN A，et al. Enriching word vectors with subword information [C]// EMNLP 2016. Austin，Texas，USA：ACL，2016：26-27.

[6] BLEI D M，NG A Y，JORDAN M I. Latent Dirichlet Allocation [J]. the Journal of Machine Learning Research，2003(3)：993-1022.

[7] PHAN X H，NGUYEN L M，HORIGUCHI S. Learning to classify short and sparse text & Web with hidden topics from large-scale data collections[C] // The 17th International Conference on World Wide Web. Beijing，China：国际万维网会议委员会，北京航空航天大学，2008：91-100.

[8] CHEN Mengen，JIN Xiaoming，SHEN Dou. Short text classification improved by learning multi-granularity topics[C] // Proceedings of the 22nd International Joint Conference on Artificial Intelligence. Barcelona，Catalonia，Spain：AAAI Press，2011：1776-1781.

[9] 李锐，张谦，刘嘉勇. 基于加权 Word2vec 的微博情感分析[J]. 通信技术，2017，50(3)：502-506.

[10] 张谦，高章敏，刘嘉勇. 基于 Word2vec 的微博短文本分类研究[J]. 信息网络安全，2017(1)：57-62.

[11] 曾蒸，李莉，陈晶. 用于情感分类的双向深度 LSTM[J]. 计算机科学，2018，45(8)：213-217，252.

[12] HOCHREITER S，SCHMIDHUBER J. Long short-term memory [J]. Neural computation，1997，9(8)：1735-1780.

[13] SCHUSTER M，PALIWAL K K. Bidirectional recurrent neural networks[J]. IEEE Transactions on Signal Processing，1997，45(11)：2673-2681.

[14] 张小川，余林峰，桑瑞婷. 融合 CNN 和 LDA 的短文本分类研究[J]. 软件工程，2018，21(6)：17-21.

[15] 唐明，朱磊，邹显春. 基于 Word2Vec 的一种文档向量表示[J]. 计算机科学，2016，43(6)：214-217，269.

(上接第 20 页)

来，最大程度地利用了车间加工过程中产生的数据资源。在此模型基础上，又利用模糊网络分析法寻找该模型的关键节点，用实例证明该方法比传统的 AHP 层次分析法找到的关键节点更为准确，识别度更高，并且能成功应用于基于数据的复杂网络模型。

参考文献

[1] 陶雪娇，胡晓峰，刘洋. 大数据研究综述[J]. 系统仿真学报，2013，25(S1)：142-146.

[2] APPLEGATE D，COOK W. A computational study of the job-shop scheduling problem[J]. Orsa Journal on Computing，1991，3(2)：149-156.

[3] 张纪会，徐军芹. 适应性供应链的复杂网络模型研究[J]. 中国管理科学，2009，17(2)：76-79.

[4] 李春光. 复杂网络建模及其动力学性质的若干研究[D]. 成都：电子科技大学，2004.

[5] 李晓娟. 复杂产品制造过程加权演化模型与节点重要度分析[D]. 乌鲁木齐：新疆大学，2012.

[6] YU J R，CHENG S J. An integrated approach for deriving priorities in analytic network process [J]. European Journal of Operational Research，2007，180(3)：1427-1432.