doi:10.11959/j.issn.1000-436x.2017088

基于神经网络的文本表示模型新方法

曾诚飞[1]，张笑燕[1]，杜晓峰[2]，陆天波[1]
（1. 北京邮电大学软件学院，北京 100876；2. 北京邮电大学计算机学院，北京 100876）

摘 要：提出了一种改进的文本表示模型提取文本特征词向量方法。首先构建基于词典索引和所对应的词性索引的 double word-embedding 列表的 word-embedding 词向量，其次，利用在此基础上 Bi-LSTM 循环神经网络对生成后的词向量进一步进行特征提取，最后，通过 mean-pooling 层处理句子向量后且使用了 softmax 层进行文本分类。实验验证了 Bi-LSTM 和 double word-embedding 神经网络相结合的模型训练效果与提取情况。实验结果表明，该模型不但能较好地处理高质量的文本特征向量提取和表达序列，而且比 LSTM、LSTM+context window 和 Bi-LSTM 这 3 种神经网络有较明显的表达效果。

关键词：神经网络；词向量；Bi-LSTM；文本表示
中图分类号：TP183 **文献标识码**：A

New method of text representation model based on neural network

ZENG Shui-fei[1], ZHANG Xiao-yan[1], DU Xiao-feng[2], LU Tian-bo[1]
(1. School of Software Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China;
2. School of Computer, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: Method of text representation model was proposed to extract word-embedding from text feature. Firstly, the word-embedding of the dual word-embedding list based on dictionary index and the corresponding part of speech index was created. Then, feature vectors was obtained further from these extracted word-embeddings by using Bi-LSTM recurrent neural network. Finally, the sentence vectors were processed by mean-pooling layer and text categorization was classified by softmax layer. The training effects and extraction performance of the combination model of Bi-LSTM and double word-embedding neural network were verified. The experimental results show that this model not only performs well in dealing with the high-quality text feature vector and the expression sequence, but also significantly outperforms other three kinds of neural networks, which includes LSTM, LSTM+context window and Bi-LSTM.

Key words: neural network, word-embedding, Bi-LSTM, text representation

1 引言

伴随智能机器人的快速发展，自动问答系统已成为智能机器人最重要的核心技术之一，众多智能交互的业务都与自动问答系统紧密相关，如文本表示、文本分类和答案抽取等。文本表示和分类方法也是自动问答系统中的关键处理技术，其中，文本特征提取、文本分类和答案抽取是自动问答系统最基本的处理步骤。因此，自动问答系统的性能与文本表示和直接密切相关。

当面临和处理海量文本训练语料时，将文本表示和文本分类处理技术应用于智能机器人的自动问答系统，不管是文本分类准确率还是文本的表示方法，都存在文本信息表达和问句分类准确率偏低的问题，这些技术与人工智能、自然语言理解技术、机器学习甚至深度学习的产业化还存在一定差距。大多数研究者基于词袋、句子包、词向量等方面的单个字、词、短语、句子、段就文本表示、文本分类进行相关研究，这些研究成果存在一些缺陷和片面性；然而还有些学者分别从卷积神经网络、循环神经网络计算词向量和上下文信息预测等方面进行相应的研究，这些研究同样存在一些明显

收稿日期：2016-12-31；修回日期：2017-03-09

的不足之处，如 Xu 等[1]利用循环神经网络 RNN 结构进行文本标注的实验并取得了一些研究成果，但是文中对传统循环神经网络 RNN 结构中长期依赖关系这一特性，并没有对文本特征表达发挥较好的作用。

众所周知，最近深度学习相关研究成果表明神经网络的异军突起，人们发现神经网络、深度学习在自然语言处理 NLP 领域的应用潜能，同时，许多研究人员对人工神经网络与传统的统计学算法与方法进行综合比较，如朴素贝叶斯、支持向量机（SVM）、隐马尔可夫模型等有着明显的优势和科研价值。尤其是 Geoffrey[2]早期提出了 distributed representation 概念与思想对文本表示方法有重要作用。还有 Yao 等[3]深入分析了在中文分词中使用 Bi-LSTM 神经网络如何训练该模型，论证了带有词向量的 BLSTM 神经网络是一种有效的标注解决方案，值得进一步研究。考虑到文本的输入具有时序性，因 RNN 循环神经网络是一种时序神经网络，可以将当前时间节点的输入与前一时间节点的输出进行运算而得到当前时刻的输出结果，所以 RNN 循环神经网络解决了当前节点与之前节点之间的相关性，即文本特征向量将很好地被该方法有效表征。Yao 等[3]进一步利用 BLSTM 神经网络处理序列性数据，他们所采用的方法通过实验验证了比 LSTM 神经网络在处理中文分词更加有效。Chiu [4]在设计了一种 Bi-LSTM 和 CNN 结合的神经网络模型，实验证明了该模型获得较好的效果。

基于此，为了更有效地处理文本表示提取模型、问句分类的综合性能，本文提出一种改进的文本表示方法提取文本特征词向量模型。首先构建基于词典索引与对应词性索引的 double word-embedding 列表生成 word-embedding 词向量。其次，在此基础上结合 Bi-LSTM 循环神经网络对其进一步特征提取。最后，将通过 mean-pooling 层处理句子向量后使用 softmax 层进行文本分类，验证 Bi-LSTM 循环神经网络和特征词向量提取模型相结合的训练效果，使之能较好地处理高质量的文本特征向量提取方法和表达序列。

本文设计的方法和主要贡献包括以下 3 个方面。

1) 在 word-embedding 处理文本词向量的基础上，本文提出了增加单词索引和所对应的词性索引的“双索引”word-embedding 词向量构造方法，该方法能较好地将一个单词转化为词典索引和所对应的词性索引 word-embedding 列表的词向量，接着进行拼接生成的实数向量。实验表明，该方法较传统的文本词向量方法表现出了更好的文本表示能力，增加了文本特征提取模型方法。

2) 为了进一步研究文本特征词向量的提取与神经网络相结合模型的训练效率，本文设计了 Bi-LSTM+double word-embedding 架构的神经网络文本特征向量提取模型，该提取模型不仅完成了文本自身的特征表达信息，而且更容易体现和发现文本当前时刻与之后时刻之间关系的上下文特征信息，使特征向量具有高质量的特征表达效果，进一步提高了该模型的训练效率与性能。

3) 使用哈尔滨工业大学信息检索实验室提供问题集数据进行了验证，通过对 LSTM、LSTM+context window 、 Bi-LSTM 和 Bi-LSTM+double word-embedding 这 4 种深度学习模型进行实验数据对比与结果分析， Bi-LSTM 和 double word-embedding 的神经网络模型不但便捷地获取了最佳模型的训练参数，而且更表明了所采用的改进模型相比其他 3 种神经网络模型在文本特征词向量提取方面有较明显的优势。

2　主要方法与理论

这里先介绍本文应用的主要方法和理论，具体详细阐述如下。

2.1　word-embedding 词向量

自然语言理解 NLP 的首要问题就是如何转化为机器学习，接着是如何找到一种有效的方法将文本表示数字化，及文本表示的向量计算与向量之间的相关性。Geoffrey[2]在 1986 年提出了 distributed representation 用来表示词，现通常被称为“word representation”或“word-embedding”，中文称为“词向量”，本文用“word-embedding”表示词向量。word-embedding 词向量是自然语言处理 NLP 中的一种特征提取技术，它将从词典中获得的单词或短语映射成为与词典大小和索引相关的低纬度实数向量。word-embedding 词向量文本表示比传统的人工提取特征向量的方法具有更好的特征表达效果，并且能有效避免特征向量的维度灾难。构成 word-embedding 词向量计算如式(1)所示。

$$Vec_i = \boldsymbol{Emb}\left(IndexD\left(W_i\right)\right) \tag{1}$$

通过式(1)的计算，输入的单词序列将被构造成二维实数矩阵作为神经网络的输入。其中，$W_i \in Dict$，*Dict* 包含了所有单词的词典，*IndexD* 表示获取单词在 *Dict* 词典中所对应的索引，***Emb*** 表示 *n* 行 *d* 列的二维词向量列表，*n* 为 *Dict* 词典大小，*d* 为人工指定的词向量维度，***Emb***(*i*)表示获取 ***Emb*** 中第 *i* 行的行向量。在神经网络训练过程中，***Emb*** 中的词向量将被视为参数进行训练，训练完成的 ***Emb*** 参数将作为特征向量提取的基础和前提。

2.2 LSTM 循环神经网络

LSTM 全称为 long short term memory，是在循环神经网络基础上引入了 memory cell 单元。LSTM 神经网络单元具有学习长依赖性数据特征的能力和用途，本节分别从 LSTM 单元结构与算法思想、Bi-LSTM 神经网络结构模型进行描述，分析了 LSTM 神经网络在解决时序性建模问题方面表现尤为突出。

2.2.1 LSTM 单元结构与算法思想

借鉴文献[3~5]，LSTM 单元结构如图 1 所示。LSTM 单元结构包含 4 个元素：1 个输入门、1 个循环自连接的神经元、1 个遗忘门和 1 个输出门。LSTM 单元的增加可以对 RNN 循环神经网络的神经元状态传递进行更好的优化和补充。

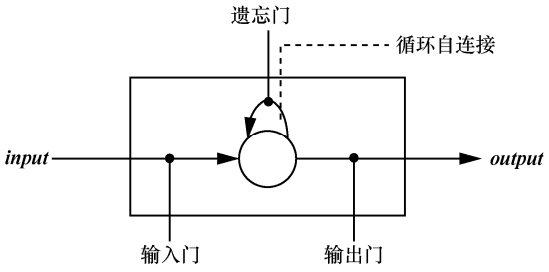

图 1 LSTM memory cell 单元结构

根据图 1 分析可知，LSTM memory cell 单元结构的作用是保存了 *t* 时刻的信息，该单元的状态传递关系又取决于 3 种门作用。输入向量 ***input*** 包含了 *t* 时刻的输入信息和上一时刻的自循环信息，也就是说输入门决定了哪些新信息被 memory cell 存储；遗忘门控制最新时刻哪些信息需要被抛弃；而输出门决定 cell 中哪些信息会被输出并进入下一时刻的自循环迭代。

本文采用文献[3~7]的 memory cell 结构模型和计算思路，LSTM 中的 memory cell 具体的算法定义与思想如式(2)~式(7)。

$$\boldsymbol{i}_t = \sigma\left(\boldsymbol{W}_i x_t + \boldsymbol{U}_i \boldsymbol{h}_{t-1} + \boldsymbol{b}_i\right) \tag{2}$$

$$\boldsymbol{C}'_t = \tan h\left(\boldsymbol{W}_c x_t + \boldsymbol{U}_c \boldsymbol{h}_{t-1} + \boldsymbol{b}_c\right) \tag{3}$$

$$\boldsymbol{f}_t = \sigma\left(\boldsymbol{W}_f x_t + \boldsymbol{U}_f \boldsymbol{h}_{t-1} + \boldsymbol{b}_f\right) \tag{4}$$

$$\boldsymbol{C}_t = i_t \boldsymbol{C}'_t + f_t \boldsymbol{C}_{t-1} \tag{5}$$

$$o_t = \sigma\left(\boldsymbol{W}_o x_t + \boldsymbol{U}_o \boldsymbol{h}_{t-1} + \boldsymbol{V}_o \boldsymbol{C}_t + \boldsymbol{b}_o\right) \tag{6}$$

$$\boldsymbol{h}_t = o_t \tan \boldsymbol{h}(\boldsymbol{C}_t) \tag{7}$$

式(2)~式(7)说明：$\boldsymbol{i}_t$ 和 $\boldsymbol{h}_t$ 分别是 *t* 时刻输入与输出向量，$\boldsymbol{f}_t$ 是遗忘门输出矩阵，***W*** 和 ***U*** 矩阵为输入与上一时刻输出向量的权重矩阵，***b*** 为偏置向量，函数 σ 为 sigmoid 非线性激活函数，***V*** 为权重矩阵。

2.2.2 Bi-LSTM 神经网络结构模型

Bi-LSTM 神经网络结构模型分为 2 个独立的 LSTM，输入序列分别以正序和逆序输入至 2 个 LSTM 神经网络进行特征提取，提取后的特征向量进行拼接形成最后的特征向量。在 *t* 时刻所获得的输出向量计算如式(8)所示。

$$\boldsymbol{h}_t = \boldsymbol{h}_t^f \,\|\, \boldsymbol{h}_{n-t}^r \tag{8}$$

Bi-LSTM 的模型设计理念是使 *t* 时刻所获得特征数据同时拥有过去和将来之间的相关性信息，实验证明，这种神经网络结构模型对文本特征提取效率和性能要优于单个 LSTM 结构模型，其中，$\boldsymbol{h}_t^f$ 和 $\boldsymbol{h}_{n-t}^r$ 分别表示正向输入序列在 *t* 时刻的输出向量和逆向输入序列在 *n*−*t* 时刻的输出向量，*n* 为输入序列的长度，$\boldsymbol{h}_t$ 为 *t* 时刻的最终输出向量，||表示将 2 个输出向量进行拼接后形成的词向量作为该词的最终特征表达。Bi-LSTM 中的 2 个 LSTM 神经网络参数是相互独立的，它们只共享 word-embedding 词向量列表。

3 基于神经网络的文本表示模型设计

本节重点描述了基于神经网络的文本表示提取模型设计这一主题，并且将结合实例说明文本特征向量提取模型的思路，主要内容有：double word-embedding 列表词向量、Bi-LSTM 和 double word-embedding 列表的神经网络相结合的模型架构、参数训练方法和结合实例浅析 4 个神经网络模型就文本特征向量提取思路(LSTM、LSTM+context window、Bi-LSTM 和 Bi-LSTM+double word-em-

bedding)，下面将对以上内容进行详实阐述。

3.1 double word-embedding 列表的词向量

在过去与传统构建词向量的方法中，通常的做法是将单词转化为词典所对应的索引，根据索引从词典大小长度的实数向量列表中获取对应位置的实数向量并构成该词的基础词向量用于神经网络的进一步处理，该方法存在特征词向量提取不足和固有的局限性。

本文在基于 word-embedding 词向量技术的理论和相关应用成果上，比如文献[5]中的基于 RNN 循环神经网络的 word-embedding 词向量，提出了一种改进和优化的词向量构建方法。普遍和传统的词向量构建方法是通过单个词典对应索引进行生成，这种方法对文本或词的特征表达效果不佳，存在一定局限性。为了进一步提升词向量的特征表达效果，本文采用了增加单词的词性和词典并构造了 double word-embedding 列表的词向量。当单词生成对应词向量时，不仅从该词的词典索引中获取实数向量，而且从该词所对应的词性词典中获取词性索引并构成实数向量，然后将两者 word-embedding 词向量进行拼接合成最终的基于词典索引和词性索引的 double word-embedding 词向量。double word-embedding 词向量生成方法如式(9)，其构建方法的比对说明如图 2 所示。

$$\begin{aligned} \boldsymbol{Vec}_i = \boldsymbol{Emb\_d}\left(IndexD\left(W_i\right)\right)\| \\ \boldsymbol{Emb\_p}\left(IndexP\left(Property(W_i)\right)\right) \end{aligned} \tag{9}$$

其中，W_i 为当前需要转化成词向量的词，*IndexD* 和 *IndexP* 分别表示从词典和词性词典中获取当前词的索引和对应词性的索引，*Property* 表示获得当前词的词性，***Emb_d*** 和 ***Emb_p*** 分别表示从词典 word-embedding 和词性 word-embedding 列表中获取对应索引的词向量，‖表示 2 个向量拼接后生成的最终词向量 $\boldsymbol{Vec}_i$。本文实验部分表明，增加了词性 word-embedding 后所提取的词向量具有更好的特征表达效果。

3.2 Bi-LSTM 和 double word-embedding 列表的神经网络模型

借鉴 Chiu[4]的模型启发该部分的设计思路，采用文献[5~7]关于 LSTM 和 Bi-LSTM 循环神经网络

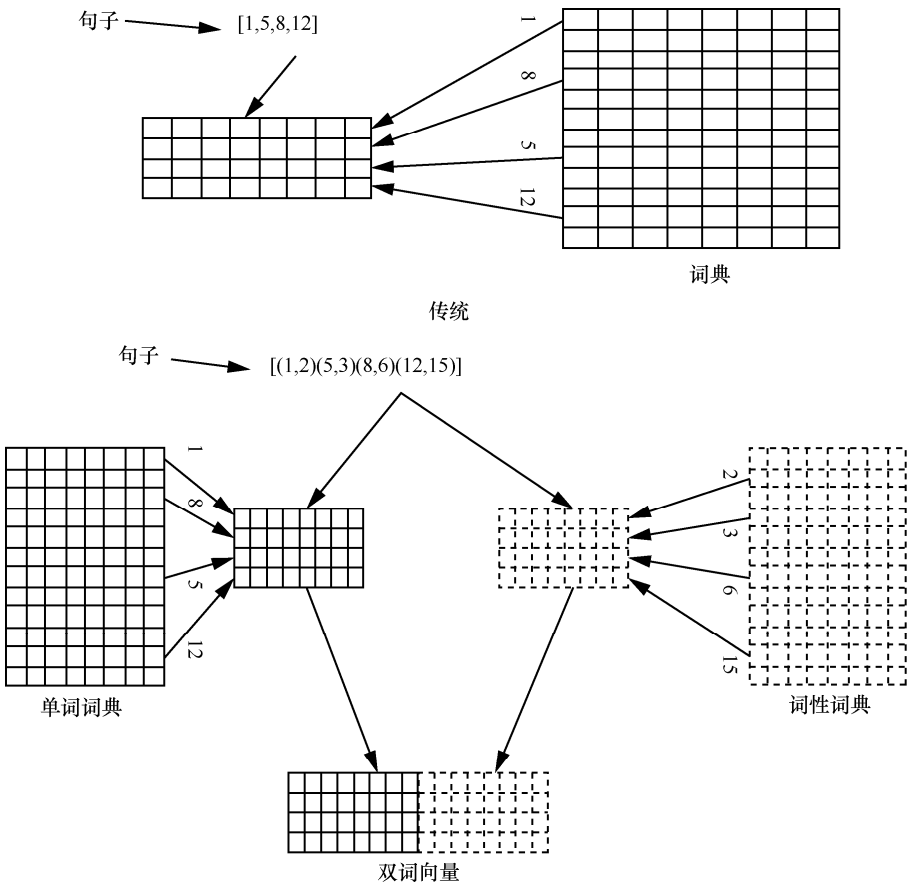

图 2　双词向量构建方法的比对

结构的理论基础和方法，尤其是 Tan 等[6]深度学习应用于答案选择研究，选择 LSTM 循环神经网络处理可变长的序列输入，本文提出了对 Bi-LSTM 和 double word-embedding 的神经网络模型相结合的改进文本特征词向量提取结构，具体如图 3 所示。

从图 3 可知，模型中的输入由句子转化而成的词向量矩阵，每一个单词的词向量由对应的词典索引和词性索引所获得的 word-embedding 向量拼接而成的。首先是词向量矩阵分别以正序和逆序输入至 Bi-LSTM 循环神经网络中，分别获得正向词向量序列 $\boldsymbol{h\_f}(0)$~$\boldsymbol{h_f}(n)$和逆向词向量序列 $\boldsymbol{h\_r}(0)$~$\boldsymbol{h\_r}(n)$。其次是模型将获得的词向量进行拼接，具体拼接方法是 $\boldsymbol{h_f}(t)$连接 $\boldsymbol{h\_r}(n-t)$，此时拼接得到的 $\boldsymbol{h}(0)$~$\boldsymbol{h}(n)$为所提取的文本特征向量，其中，$\boldsymbol{t}$ 表示在 t 时刻所生成的词向量。再次是模型将获得的词向量进行平均值池化处理并生成句子向量 $\boldsymbol{h\_mean}$，接着将 $\boldsymbol{h\_mean}$ 向量输入至 1 个 softmax 层进行分类，这里文本分类处理验证了特征向量对文本特征的表达效果情况，改进的模型结构计算如式(10)和式(11)所示。

$$P(y=i\,|\,\boldsymbol{X})=\frac{\mathrm{e}^{\boldsymbol{W}_i(\mathrm{Mean}(BL(\boldsymbol{X})))+\boldsymbol{b}_i}}{\sum_j \mathrm{e}^{\boldsymbol{W}_j(\mathrm{Mean}(BL(\boldsymbol{X})))+\boldsymbol{b}_j}} \tag{10}$$

$$Y_{\mathrm{pred}}=\arg\max_i(P(y=i\,|\,\boldsymbol{X})) \tag{11}$$

式(10)和式(11)中的 $\boldsymbol{X}$ 表示输入的词向量矩阵，$BL(\boldsymbol{X})$表示经过 Bi-LSTM 处理后生成的词向量矩阵，Mean 函数将词向量矩阵进行平均池化处理后得到向量 $\boldsymbol{h\_mean}$，$\boldsymbol{W}$ 为对应类别 $\boldsymbol{i}$ 的权重矩阵，$\boldsymbol{b}$ 为偏置向量，$\boldsymbol{h\_mean}$ 经过 softmax 层处理得到类别 $\boldsymbol{i}$ 的预测概率，最终通过 argmax 获得预测结果中概率最大的类别 Y_{pred} 为最终的计算结果。

3.3 LSTM、LSTM+context Window、Bi-LSTM 和 Bi-LSTM+double word-embedding 模型浅析

针对 LSTM、LSTM+context Window、Bi-LSTM 和 Bi-LSTM+double word-embedding 这 4 种神经网络模型的文本特征向量提取方法，并结合图 3 Bi-LSTM 和双 word-Embedding 神经网络模型，这里通过分析实例说明这 4 种神经网络模型之间的异同点，例如：“FBI 的全称是什么？”

1) LSTM 神经网络模型思路与具体实现过程

LSTM 神经网络模型实例

Step1 首先将句子进行分词获得单词列表为[FBI，的，全称，是，什么]这一表达形式，再将单词列表转化为单词对应的索引列表。假设索引列表值为[0, 2, 3, 6, 9]，则根据句子的索引列表从单词 word-embedding 列表中获取对应的词向量后得到词向量序列[$\boldsymbol{X}_0$, $\boldsymbol{X}_2$, $\boldsymbol{X}_3$, $\boldsymbol{X}_6$, $\boldsymbol{X}_9$]，其中，$\boldsymbol{X}_t$ 表示从 word-embedding 列表中所获取的第 t 个词向量，$\boldsymbol{X}_t$

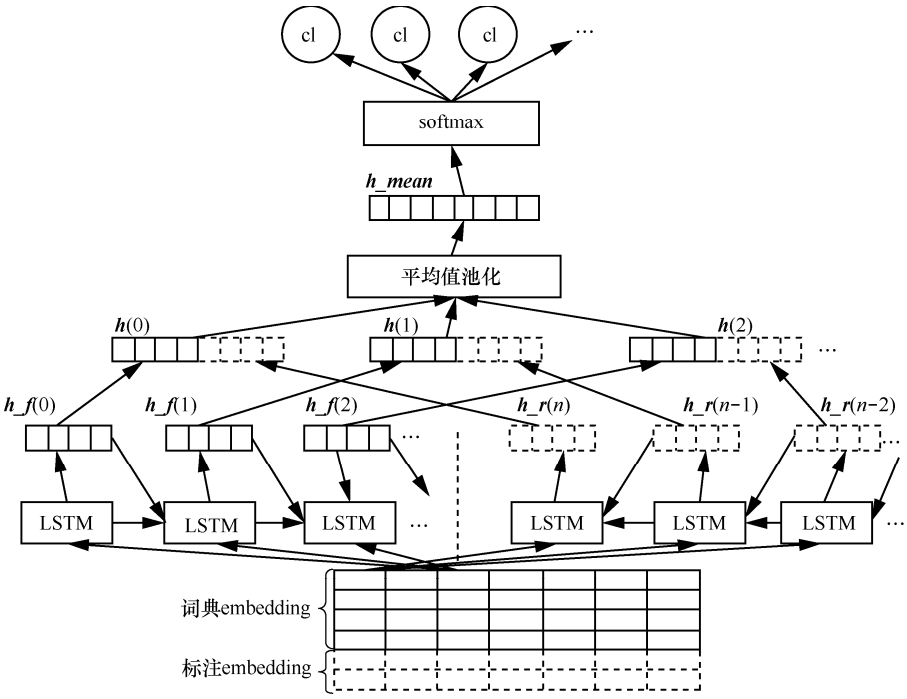

图 3 Bi-LSTM 和 double word-embedding 神经网络模型

为 128 维的实数向量。

Step2　将词向量序列输入 LSTM 神经网络中进行向量运算，其中，各 $\boldsymbol{W}$ 权值矩阵、$\boldsymbol{U}$ 权值矩阵均为 128×128 维方阵，偏置向量 $\boldsymbol{b}$ 为 128 维，运算完成后所得输出序列[$\boldsymbol{h}_0$，$\boldsymbol{h}_1$，$\boldsymbol{h}_2$，$\boldsymbol{h}_3$，$\boldsymbol{h}_4$]，其中，$\boldsymbol{h}_t$ 为第 t 个输入序列得到的输出特征向量，$\boldsymbol{h}_t$ 为 128 维实数向量。

Step3　将输出序列进行平均池化处理得到句子向量 ***h_mean*** 并输入 softmax 层进行分类。

Step4　文本特征词向量提取与分类处理完毕。

2) LSTM+context window 神经网络模型思路与具体实现过程

LSTM+context window 神经网络模型实例

Step1　首先将句子进行分词得到单词列表[FBI，的，全称，是，什么]，将单词列表转化成单词对应的索引列表。假设索引列表值为[0，2，3，6，9]，并为句子索引列表中每个索引添加上下文窗口（context window），本实验窗口尺寸设为 5。若窗口尺寸为 3，则所得二维索引列表[[−1, 0, 2], [0, 2, 3], [2, 3, 6], [3, 6, 9], [6, 9, −1]]，然后根据该索引列表从单词 word-embedding 列表中获取对应的词向量并与窗口中的词向量拼接（索引值为−1 表示对应 word-embedding 列表最后一条词向量），则得到的词向量序列为[$\boldsymbol{X}_{-1}$‖ $\boldsymbol{X}_0$‖ $\boldsymbol{X}_2$, $\boldsymbol{X}_0$‖ $\boldsymbol{X}_2$‖ $\boldsymbol{X}_3$, $\boldsymbol{X}_2$‖ $\boldsymbol{X}_3$‖ $\boldsymbol{X}_6$, $\boldsymbol{X}_3$‖$\boldsymbol{X}_6$‖$\boldsymbol{X}_9$, $\boldsymbol{X}_6$‖$\boldsymbol{X}_9$‖ $\boldsymbol{X}_{-1}$]，其中，‖表示词向量拼接。

Step2　将所得词向量序列输入至 LSTM 神经网络中，其中，$\boldsymbol{W}$、$\boldsymbol{U}$ 权值矩阵维度为(64×N)×(64×N)方阵，$\boldsymbol{b}$ 偏置向量维度为 64×N，N 表示窗口大小，输入后得到输出序列[$\boldsymbol{h}_0$，$\boldsymbol{h}_1$，$\boldsymbol{h}_2$，$\boldsymbol{h}_3$，$\boldsymbol{h}_4$]，$\boldsymbol{h}_t$ 维度为 64×N。

Step3、Step4　后续过程与 LSTM 神经网络模型是一致的。

3) Bi-LSTM 神经网络模型思路与具体实现过程

Bi-LSTM 神经网络模型实例

Step1　首先将句子进行分词得到单词列表并转化对应的索引列表，假设索引列表值为[0, 2, 3, 6, 9]，再根据该列表得到其逆序索引列表[9, 6, 3, 2, 0]，接着将 2 个索引列表转化成对应的词向量序列[$\boldsymbol{X}_0$, $\boldsymbol{X}_2$, $\boldsymbol{X}_3$, $\boldsymbol{X}_6$, $\boldsymbol{X}_9$]与[$\boldsymbol{X}_9$, $\boldsymbol{X}_6$, $\boldsymbol{X}_3$, $\boldsymbol{X}_2$, $\boldsymbol{X}_0$]，$\boldsymbol{X}_t$ 为 64 维实数向量。

Step2　将 2 个词向量序列分别输入至 2 个不同的 LSTM 神经网络模型中，每个 LSTM 神经网络中的 $\boldsymbol{W}$、$\boldsymbol{U}$ 权值矩阵为 64×64 方阵，偏置向量 $\boldsymbol{b}$ 为 64 维。输入后分别得到输出序列[***h_f***(0), ***h_f***(1), ***h_f***(2), ***h_f***(3), ***h_f***(4)] 与 [***h_r***(0), ***h_r***(1), ***h_r***(2), ***h_r***(3), ***h_r***(4)]，然后将 2 个序列进行首尾拼接，得到序列[***h_f***(0)‖ ***h_r***(4), ***h_f***(1)‖ ***h_r***(3), ***h_f***(2)‖ ***h_r***(2), ***h_f***(3)‖ ***h_r***(1), ***h_f***(4)‖ ***h_r***(0)]，拼接后词向量序列元素为 128 维。

Step3　将输出序列进行平均池化处理得到句子向量 ***h_mean*** 并输入 softmax 层进行分类。

Step4　文本特征词向量提取与分类处理完毕。

4) Bi-LSTM+double word-embedding 神经网络模型思路与具体实现过程。

Bi-LSTM+double word-embedding 神经网络模型实例

Step1　首先将句子进行分词和词性标注得到带有词性的二维分词列表[[FBI, eng]，[的，uj]，[全称，n]，[是，v]，[什么，r]]，根据单词词典和词性词典将分词列表转化成为对应的索引，假设索引列表值为[[0, 1], [2, 3], [3, 5], [6, 7], [9, 9]]，其中，列表元素的第一维对应单词词典索引，第二维对应词性词典索引，接着将该索引列表进行逆序处理得到第 2 个索引列表[[9, 9], [6, 7], [3, 5], [2, 3], [0, 1]]。现根据列表元素中的索引分别从词典 word-embedding 列表和词性 word-embedding 列表中获取对应的词向量并将它们进行各自拼接，所得词向量序列[$\boldsymbol{W}_0$‖ $\boldsymbol{P}_1$, $\boldsymbol{W}_2$‖ $\boldsymbol{P}_3$, $\boldsymbol{W}_3$‖ $\boldsymbol{P}_5$, $\boldsymbol{W}_6$‖ $\boldsymbol{P}_7$, $\boldsymbol{W}_9$‖ $\boldsymbol{P}_9$]和序列[$\boldsymbol{W}_9$‖ $\boldsymbol{P}_9$, $\boldsymbol{W}_6$‖ $\boldsymbol{P}_7$, $\boldsymbol{W}_3$‖ $\boldsymbol{P}_5$, $\boldsymbol{W}_2$‖ $\boldsymbol{P}_3$, $\boldsymbol{W}_0$‖ $\boldsymbol{P}_1$]，其中，$\boldsymbol{W}_t$ 表示词典 word-embedding 中的第 t 个词向量，维度为 64，$\boldsymbol{P}_t$ 表示词性 word-embedding 中的第 t 个词向量且维度为 32，‖表示词向量拼接，拼接后词向量维度为 96。

Step2　将拼接后 2 个词向量序列分别输入至 2 个不同的 LSTM 神经网络模型中，每个 LSTM 神经网络的 $\boldsymbol{W}$、$\boldsymbol{U}$ 权值矩阵为 96×96 的方阵，偏置向量 $\boldsymbol{b}$ 为 96 维。

Step3　输入后分别得到输出序列[***h_f***(0), ***h_f***(1), ***h_f***(2), ***h_f***(3), ***h_f***(4)] 和 [***h_r***(0), ***h_r***(1), ***h_r***(2), ***h_r***(3), ***h_r***(4)]，然后将 2 个序列进行首尾拼接，得到序列[***h_f***(0)‖ ***h_r***(4), ***h_f***(1)‖ ***h_r***(3), ***h_f***(2)‖ ***h_r***(2), ***h_f***(3)‖ ***h_r***(1), ***h_f***(4)‖ ***h_r***(0)]，拼接后词向量序列元素为 192 维。

Step4　将输出序列进行平均池化处理得到句子向量 ***h_mean*** 并输入 softmax 层进行分类。

Step5　文本特征词向量提取与分类处理完毕。

3.4 模型参数训练方法

本文综合算法与模型的时间复杂度和空间复杂度的性能考量参数训练方法，分别从所使用的误差函数和参数误差调整方法进行探索，保障了特征词向量模型训练的效果和优势。

1) 本文就神经网络模型训练时使用了误差函数为负似然对数函数。在进行实验验证时，发现训练集正确率越高，负似然对数误差函数则越接近 0，训练集正确率越低，负似然对数误差函数结果会越大于 0。负似然对数误差函数表达式为

$$L(\theta, D) = -\sum_{i=0}^{|D|} \mathrm{lb}(P(Y = y_i \mid x_i, \theta)) \tag{12}$$

2) 当模型训练时，本文所使用的参数误差调整方法为 Adadelta。Adadelta 与随机梯度下降方法相比较的优势在于自适应地调整误差学习率，Adadelta 方法是一种模拟牛顿法的学习方法，并且该方法选择每一次的梯度下降最佳方向，使误差函数可以在模型训练过程中更快更趋于收敛。虽然 Adadelta 方法通过一阶方法近似模拟二阶牛顿法，一方面有效地提高了误差训练速度，另一方面摆脱了学习率设定的困惑，但是该方法需要求解函数二阶导数的海瑟矩阵的逆矩阵，其时间复杂度为 $O(n^3)$，具体 Adadelta 式推导过程与详细的分析见文献[8]。因此，参数误差方法在模型实际训练过程中难以做到完美和利弊均衡的综合应用效果。

4 实验验证与结果分析

为了证明本文所采用的 Bi-LSTM 和 double word- embedding 神经网络结合的模型优越性，本文设计了同一组实验数据即训练集和测试集进行 2 组模型训练获得相关数据，然后从不同的角度将本文的方法进行数据分析。主要包括的内容有实验设置、实验数据和结果分析，突显改进的文本特征词向量提取模型优点。

4.1 实验设置

本文使用哈尔滨工业大学的问题分类语料进行实验验证，根据哈尔滨工业大学信息检索实验室提供的数据，该数据集包含了 7 个大类别和 85 个小类，大分类信息分别是描述类、人物类、地点类、实体类、时间类、数字类、未知类，总语料数达到 5 363 条问题数据。在本文进行实验过程中，将这 5 363 条问题数据的顺序进行随机打乱，再随机抽取其中 400 条问题数据作为实验的测试集。因此，本实验设置的最终训练集数据为 4 963 条训练语料，测试集数据为 400 条测试语料。

为了验证本文所设计的神经网络模型优点，通过 4 个神经网络模型的验证，实现了这 4 个神经网络模型文本分类正确率的结果对比与分析。这 4 个神经网络模型信息分别是词向量维度为 128 的 LSTM、增加了上下文窗口大小为 5 和词向量维度为 64 的 LSTM、词向量维度为 64 的 Bi-LSTM、双 word-embedding 列表的 Bi-LSTM（词向量维度为 64+32）。这 4 个神经网络模型所使用的 word-embedding 词向量维度均是让其模型训练出接近最优化参数并使测试集正确率最高的大小维度数值，从而排除了词向量维度大小对其实验的干扰和影响。而且设计了每个神经网络模型所提取的词向量均需要进行平均池化处理环节，接着输入至一个 softmax 层进行文本分类，进一步体现了本文所设计与改进的神经网络模型优越性。

本文实验中所验证的 4 种神经网络模型代码都是基于计算机语言 Python 的 Theano 框架实现的，实验所用计算机 CPU 均为英特尔酷睿 i3 处理器和主频 3.40 GHz，操作系统为微软 64 bit Windows 7，模型训练总用时大概为 1.5 h。

4.2 实验数据

1) **实验数据 1** 本文按照实验设置所述的方法和要求进行模型训练分类实验，首先是训练集和测试集的前期准备工作，经过随机打乱和任意抽取后获得的训练集和测试集的具体数据分布如表 1 所示。

表 1 实验训练集和测试集具体数据分布

类别	训练集	测试集
描述	785	58
人物	333	23
地点	936	73
数字	1 068	98
实体	1 242	104
时间	590	43
未知	9	1
总数	4 963	400

2) **实验数据 2** 神经网络模型中的具体参数设置如表 2 表示。4 种神经网络模型训练所获得最

优参数模型训练集和测试集的正确率结果如表 3 所示，根据表 3 数据可知，数据对比主要是根据各神经网络模型的测试集正确率结果进行分析，本文所设计和改进的模型在获得最优参数之后的测试集正确率提升至 90%的理想结果。

3) **实验数据 3**　4 种神经网络模型在模型训练迭代过程中，所获得的训练集和测试集分类错误分别如图 4 和图 5 所示，图 4 和图 5 中纵坐标轴数字分别表示训练集和测试集分类错误率，横坐标轴数字分别表示训练集和测试集正确率验证的轮次数量，根据图 5 折线变化情况的数据分析可知，Bi-LSTM+double word-embedding 神经网络模型的测试集分类错误百分比降至最低，到达本文改进的模型设计预期目标，该模型所使用的误差下降训练方法为 Adadelta 方法。

4) **实验数据 4**　测试集部分概率分布状况如表 4 所示。从表 4 中发现，测试集概率密度分布预测值基本上符合样本的正确标签。

表 2　　**神经网络模型参数设置**

参数名称	维度	层级	参数含义
W_i^0	96×96	正向 LSTM	输入门当前序列表征矩阵
W_f^0	96×96	正向 LSTM	遗忘门当前序列表征矩阵
W_o^0	96×96	正向 LSTM	输出门当前序列表征矩阵
W_c^0	96×96	正向 LSTM	当前序列候选表征矩阵
U_i^0	96×96	正向 LSTM	输入门前时刻序列表征矩阵
U_f^0	96×96	正向 LSTM	遗忘门前时刻序列表征矩阵
U_o^0	96×96	正向 LSTM	输出门前时刻序列表征矩阵
U_c^0	96×96	正向 LSTM	前时刻序列候选表征矩阵
b_i^0	1×96	正向 LSTM	输入门偏置向量
b_f^0	1×96	正向 LSTM	遗忘门偏置向量
b_o^0	1×96	正向 LSTM	输出门偏置向量
b_c^0	1×96	正向 LSTM	候选偏置向量
W_i^1	96×96	反向 LSTM	输入门当前序列表征矩阵
W_f^1	96×96	反向 LSTM	遗忘门当前序列表征矩阵
W_o^1	96×96	反向 LSTM	输出门当前序列表征矩阵
W_c^1	96×96	反向 LSTM	当前序列候选表征矩阵
U_i^1	96×96	反向 LSTM	输入门前时刻序列表征矩阵
U_f^1	96×96	反向 LSTM	遗忘门前时刻序列表征矩阵
U_o^1	96×96	反向 LSTM	输出门前时刻序列表征矩阵
U_c^1	96×96	反向 LSTM	前时刻序列候选表征矩阵
b_i^1	1×96	反向 LSTM	输入门偏置向量
b_f^1	1×96	反向 LSTM	遗忘门偏置向量
b_o^1	1×96	反向 LSTM	输出门偏置向量
b_c^1	1×96	反向 LSTM	候选偏置向量
W	7×192	SOFTMAX	分类层表征矩阵
b	1×7	SOFTMAX	分类层偏置向量

表 3　　4 种神经网络模型训练所获得最优参数模型训练集和测试集的正确率

模型	训练集正确率	测试集正确率	词向量维度
LSTM	99%	82%	128
LSTM+context window	99%	86%	5×64
Bi-LSTM	99%	87%	64
Bi-LSTM+double word-embedding	99%	90%	64+32

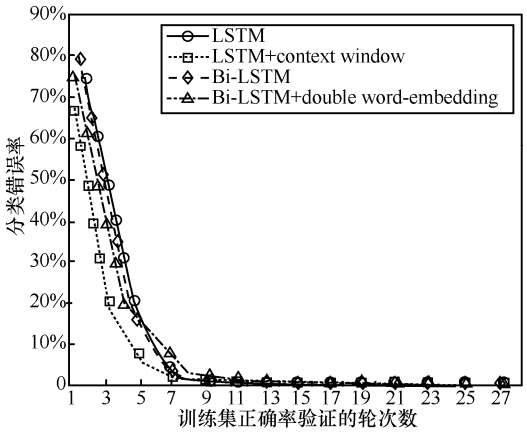

图 4　神经网络模型训练过程训练集分类错误率

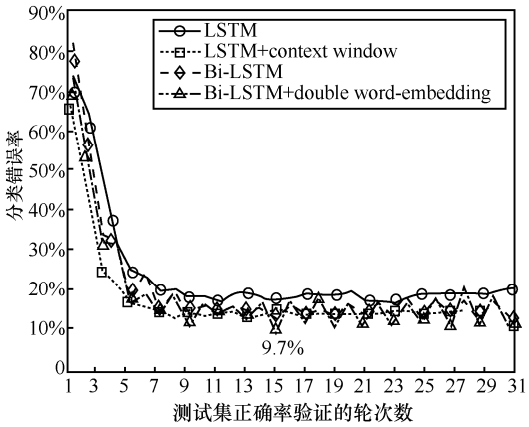

图 5　神经网络模型训练过程测试集分类错误率

表 4　　测试集部分概率分布状况

样本编号	类别 0	类别 1	类别 2	类别 3	类别 4	类别 5	类别 6	正确标签
24	0.000 5	0.001 748	0.874 618	0.100 758	0.000 031	0.009 069	0.013 275	2
25	0.439 797	0.093 689	0.024 485	0.000 94	0.317 784	0.115 709	0.007 597	4
26	0.000 013	0.054 079	0.879 863	0.012 894	0.050 611	0.000 197	0.002 342	3
27	0.611 823	0.160 361	0.135 009	0.000 045	0.068 51	0.000 001	0.024 251	0
28	0.000 007	0	0.000 001	0.999 979	0.000 001	0	0.000 012	3
29	0.112 567	0.040 439	0.035 534	0.000 341	0.769 876	0.030 551	0.010 691	4
30	0.000 973	0.170 807	0.441 168	0.002 186	0.192 433	0.190 177	0.002 256	2
31	0.000 488	0.000 057	0.002 153	0.995 496	0.001 251	0.000 191	0.000 364	3
32	0.067 126	0.019 039	0.004 159	0.001 812	0.899 413	0.004 31	0.004 14	4
33	0.000 2	0.000 205	0.000 045	0.000 046	0.999 353	0.000 041	0.000 11	4
34	0.000 102	0.001 531	0.978 825	0.015 233	0.000 323	0.003 705	0.000 28	2
35	0	1	0	0	0	0	0	1
36	0.000 024	0.000 004	0.000 009	0.999 919	0.000 031	0.000 001	0.000 013	3
37	0.024 869	0.541 476	0.070 165	0.015 507	0.066 96	0.279 107	0.001 916	4
38	0.331 426	0.000 09	0.520 277	0.004 777	0.127 866	0.000 296	0.015 269	3
39	0.000 106	0.000 001	0.000 042	0.999 098	0.000 045	0	0.000 708	3
40	0.000 415	0.000 068	0.000 829	0.001 67	0.993 545	0.002 899	0.000 573	4

4.3　实验结果分析

首先，从选用的训练集和测试集语料角度进行分析。每个神经网络模型分类实验所使用的哈尔滨工业大学问题分类语料数据集来看每个类别都具有一定数量的问题集语料，且任意随机抽取的测试集语料，其每一个类别的数量是按照训练集中语料数量的比例所获得的。总体上来讲，本文实验中所使用的训练集与测试集的数据质量是比较优质的，从而保证了其神经网络模型训练所得到的最终实验数据结果具有较高的可信度和准确率。

其次，从评价神经网络模型训练的分类正确率进行浅析。评价任何一个神经网络模型在文本和问句分类准确率高低，主要取决于其测试集的准确率大小。也就是说无论训练集的正确率是否趋近完美，甚至达到 100%，该训练集也无法代表全部的输入可能性，而只有未训练的测试集数据进行测试并使其正确率达到一定的高度，如 80%~90%，这样的具备一定的实用性和可靠性。对于任何一个神经网络模型就文本特征向量的提取质量好坏，主要根据测试集最终分类效果进行评估。同理可得，本文中的实验数据结果主要针对模型训练中的测试集进行探讨。

这里就神经网络模型训练中的测试集准确率进行详实阐释，本节实验部分所选取的 3 个对比模型分别为 LSTM 循环神经网络、LSTM+context window 循环神经网络、Bi-LSTM 循环神经网络，这 3 个神经网络模型是目前最为常用也是在自然语言处理 NLP 领域中获得最佳效果的实用模型，本文提出的改进模型是在这 3 个模型基础之上的一个延伸和扩充。从神经网络模型分类正确率结果表 2 的数据中可以发现：1) 改进的文本特征词向量提取模型在文本分类正确率上取得了一定程度上的进步；2) 当 4 种神经网络模型在训练集正确率上都达到了最大化时，本文所构建的模型在测试集文本分类正确率与其他 3 种神经网络模型表现较好的分类效果。因此，本文进一步说明模型中的训练集正确率数据并不能作为可比性的指标，而是把各神经网络模型的测试集分类正确率作为实验验证的评价指标。

再次，从测试集分类正确率和选择神经网络模型特点这两者相结合进行探析。从神经网络模型分类正确率结果（表 2）的数据显示，LSTM 的测试集分类正确率为 82%，LSTM+context window 的测试集分类正确率为 86%，Bi-LSTM 的测试集分类正确率为 87%，而改进的 Bi-LSTM+double word-embedding 神经网络模型分类正确率则达到 90%，可知这 4 个模型的测试集正确率呈现逐渐递增的趋势，表现较为明显的增幅对比性。虽然这 4 种神经网络的基础模型结构都是源自循环神经网络 RNN，但是单纯的 RNN 结构的神经网络由于在权值连接和相互作用方面显得简单，使文本特征向量的有效维度并不能较好地发挥其特征表达作用。然而，循环神经网络 RNN 模型结构即使为处理时序性数据提供了思路，但是其结构本身在自然语言处理 NLP 中并没有完全发挥提升效果的作用。因此，为了改进循环神经网络 RNN，一些研究人员一方面提出了 LSTM 循环神经网络模型，该网络模型在处理数据的输入、输出和自循环过程中加入了“门”思想进行特征的放大与限制，从而明显改善了有效特征的表达效果，并且在一定程度上限制了无效特征维度的影响，在本文的 LSTM 神经网络模型测试集分类效果达到 82%的高正确率也验证了这个特点。另一方面，虽然 LSTM 模型是循环神经网络 RNN 架构的创新之处，但是基于循环神经网络 RNN 结构在处理时序性数据时往往仅考虑了当前时刻数据与其之前时刻数据的作用关系，而忽略了当前数据与未来或之后时刻数据之间的相关性。基于上述 2 方面因素，本文提出了如下改进方法与措施。1）为了解决该缺陷和进一步完善措施，采用数据增加上下文窗口方法即一个将当前数据与未来数据建立关系的有效手段之一，本文中使用 LSTM+context window 模型取得了 86%的测试集正确率，也是统筹考虑到当前数据和未来数据之间关系的重要性。2）LSTM+context window 模型虽然在一定程度上建立了当前数据与未来数据之间的关系，但是无法构建当前数据与未来数据之间的全面关系信息，并且利用增加上下文窗口方法容易引起成倍词向量维度的猛增，从而导致了模型训练的难度，因而需要更好的神经网络结构改进和解决该模型的缺点，这里采用了 Bi-LSTM 神经网络模型弥补与解决上述存在的问题。该模型训练所获得的词向量不仅完全包含了当前时刻分别与过去、未来的关系信息，而且词向量的维度并没有增加，实验中 Bi-LSTM 模型对测试集的分类正确率达到 87%，也是对特征向量提取模型改进的结果体现。3）本文提出了基于以上模型，从 word-embedding 词向量着手，目的是提

供更多的有效特征信息。大部分文献里都是采用传统的词向量构成方法即是基于该单词的词典索引，考虑到单词的词性也是该单词特征的一个重要表达，所以在word-embedding基础上增加了词性词向量列表。本文中词典词向量维度设置为64，词性词向量维度设置为32，模型实验中的测试集分类正确率为90%，该结果表明了在word-embedding词向量基础上增加词性维度确实增强了文本特征向量的表达效果。

最后，从文本特征向量提取效果进行简析。由神经网络模型训练过程测试集分类错误率（图3）数据表明，该数据在一定程度上论证了改进的文本特征向量提取模型的优点，测试集的分类错误率降至最低，也就是说可以方便地获取最佳模型训练参数，因而所采用的模型相比其他3种神经网络模型在文本特征提取方面有较凸显的优势。

综合上述实验结果与分析可得，文中所使用Bi-LSTM+double word-embedding的神经网络架构在文本特征向量提取模型表现为效果最佳，比其他3种神经网络模型在文本特征词向量提取的效果更好，使文本特征词向量对本身应有的文本表示起到了高质量的特征表达效果，也更有利于应用到未来和后续的自然语言处理NLP各项核心技术中。

5 相关研究

伴随机器学习特别是神经网络技术的快速崛起，产业界与工业界对人工智能的持续升温，为了提高自动问答系统文本特征表达的效率，词向量、神经网络、深度学习相关算法吸引了大量科研工作者的关注和研究，如Mikolov等[9]、Maas等[10]、Cho等[11]词向量方面的研究为文本特征词向量的应用和推广提供了基础理论，并产生了一批与词向量和神经网络相结合的研究文献，如Santos[12]等把词向量与神经网络进行组合在一起的研究思路与方法，并取得了预期的研究成果。Severyn[13]提议用卷积神经网络学习一种问句和答案表示优化方法。Cross[14]描述了对于增加依赖和增加连续分析的减少特征，他们文中呈现了一个最简单的Bi-LSTAM句子表示模型。

在词向量算法方面，Geoffrey等[2]在1986年利用了distributed representation来表示词，他们率先提出了分布表示的思想，为词向量应用和研究发展作出重要贡献。Turian[15]在2010年继续沿用与深化word representations概念，他们所采用的word representations思路与传统的统计学方法进行了全面的比较，结果证明，word representations方法不但在很大程度上更容易减少词向量的维度，而且对特征向量有更明显的表现效果。Sugawara等[16]通过开发神经语言模型的word-embedding词向量捕获了语义和语法的行为，并可能找到这些词之间的关系，他们也开发了一种有监督的词义消歧，接着词义消歧利用word-embedding词向量作为局部上下文特征。因此，词向量算法往往可以提取高质量的文本特征向量表达序列。

在词向量与Bi-LSTM神经网络相结合模型设计方面。近来，大量学者不但关注对文本特征向量表示方法的相关研究，而且对词向量与神经网络如何相结合进行文本表示和分类一直是研究的热点。Xu等[1]和Chiu等[4]认为，即使RNN循环神经网络作为一种特殊的变种，但是当LSTM循环神经网络在处理序列性数据比如文本时，他们证明了比普通RNN循环神经网络更加便捷和有效。随着研究人员对神经网络的进一步研究，Chiu等[4]在LSTM的基础上又提出了Bi-LSTM神经网络，他们不仅将输入数据以正序和逆序的方式分别输入2个LSTM神经网络中，而且将所得到的结果进行拼接。这样的模型与单一的LSTM神经网络相比较，其在处理结果上有着较好的表现和优势。Sundermeyer[17]构建了英语和法语方面的LSTM神经网络结构，他们从多个层面分析了模型的优缺点并总结了LSTM神经网络的处理效果状况，实现了在语音识别系统顶层WER的大量进展或改进。Wang等[18]探析了Bi-LSTM神经网络在词性标注、分词、命名实体识别等领域的应用，通过实验的方法取得了不错的效果。Huang[19]结合了LSTM神经网络和CRF这两者方法，借助实验的形式，验证了其模型在序列标注中的有效性和健壮性。Wang等[20]介绍了使用Bi-LSTM神经网构建一种由字符组成的词表示向量模型，特别与传统的基准相比较的益处是支持各种丰富的语言。

但是通过分析以上相关研究结果发现，这些方法都存在一些共同的弊端：1) 仅侧重于特征词向量表示自己本身的某个方面研究，并没有考虑或结合对词典索引和所对应的词性索引即“双索引”选择

方法和效果的研究，也就是他们研究的侧重点与角度不同，同样，学者们对神经网络也是从某个方面或某个层面的单一研究对象内容；2) 他们在对词向量与神经网络相结合进行理论应用和实验验证时，绝大多数仅是从文本分类的准确率角度进行论证和比较，并没有从文本表示效果和文本当前时刻与之前时刻之间的关联性进行深入挖掘或进一步的深度分析。总而言之，本文对文本特征词向量提取方法及 Bi-LSTM 和 double word-embedding 的神经网络框架模型等内容进行了重点的描述与分析，这些研究对文本表示和文本分类技术应用在自然语言处理 NLP 领域中起到有益补充作用，对将来进一步相关研究有一定的启发和承上启下作用。

6　结束语

从文本特征词向量提取模型出发，提出了基于 word-embedding 词向量的词索引和词性索引的 double word-embedding 词向量生成方法，接着在 RNN 循环神经网络和 word-embedding 词向量基础上，设计了一种基于 Bi-LSTM 循环神经网络加词典和词性 double word-embedding 词向量模型。通过实验验证和数据分析，目的是改进和优化文本特征词向量的提取效果。本文首先对改进的神经网络模型的相关理论进行深入研究与应用。一方面在理论与方法上进行阐释和设计，即在此基础上，本文对模型的总体架构进行了详细设计并与 LSTM 神经网络、LSTM+context window 神经网络、Bi-LSTM 神经网络模型通过实验就文本分类的正确率进行结果对比与分析。另一方面通过实验验证本文应用的理论和达到预期的目标，即本文中实验部分所使用的训练集和测试集数据来自哈尔滨工业大学问题分类语料库，通过实验中对测试集文本的分类正确率对比可知，改进的文本特征词向量提取模型获得了 90%的测试集分类正确率，高于其他 3 种神经网络模型的分类正确率。由此可见，本文所提出的改进模型提取的文本特征向量相比传统的神经网络拥有更好的特征表达性，该模型为基于神经网络的文本特征向量提取方法研究提供了一种新思路和新架构，延伸了神经网络和词向量相结合的相关理论、方法的有益研究参考价值，也极大地推动了该模型和方法在自然语言理解 NLP 领域的应用。

虽然 Bi-LSTM 与 double word-embedding 神经网络相结合的模型取得一些成果，但是对于文本表示在词、短语、句子、段等方面，及与神经网络相结合的模型尚有研究空间，包括基于卷积神经网络的词和句两者相结合的文本特征提取模型，及 LSTM 和 word-embedding 相结合的循环神经网络 RNN 行业应用推广，如 Rao[21]应用了 LSTM 神经网络和 word-embedding 相结合的方案对于社会媒体信息进行文本分类，并且该技术推广于服务商提供客户服务的投诉与选民投票倾向分析的领域，这些研究主题与方向都有待进一步的探讨，也是下一步研究的方向之一，这些研究将夯实和拓展文本特征提取与分类方法的理论基础与实用价值。

参考文献：

[1] XU W D, AULI M, CLARK S. CCG supertagging with a recurrent neural network[C]//The 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers). 2015: 250-255.

[2] GEOFFREY E, HINTON. Learning distributed representations of concepts[C]//The Eighth Annual Conference of the Cognitive Science Society. Amherst, Mass, 1986: 1-12.

[3] YAO Y S, HUANG Z. Bi-directional LSTM recurrent neural network for Chinese word segmentation[J]. arXiv:1602.04874v1[cs.LG], 2016.

[4] CHIU J P C, NICHOLS E. Named entity recognition with bidirectional LSTM-CNNs[J]. Transactions of the Association for Computational Linguistics,2016,4: 357-370.

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

[6] TAN M, XIANG B, ZHOU B W. LSTM-based deep learning models for non-factoid answer selection[C]//ICLR 2016.

[7] KIPERWASSER E, GOLDBERG Y. Simple and accurate dependency parsing using Bidirectional LSTM feature representations[J]. Transactions of the Association for Computational Linguistics, 2016, 4: 313-327.

[8] ZEILER M D. ADADELTA: an adaptive learning rate method[J]. arXir:1212.5701v1[cs.LG], 2012.

[9] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality[J].Advances in Neural Information Processing Systems, 2013,26:3111-3119.

[10] MAAS A L, DALY R E,PHAM P T, et al. Learning word vectors for sentiment analysis[C]//The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. 2011: 142-150.

[11] CHO K. Natural language understanding with distributed representation[J]. Nato Asi, 2015, 147: 139-155.

[12] SANTOS C D, TAN M,XIANG B. Attentive pooling networks[J].

arXir: 1602.03609v1[cs.CL], 2016.

[13] SEVERYN A, MOSCHITTI A. Modeling relational information in question-answer pairs with convolutional neural networks[J]. arXiv: 1604.01178v1[cs.CL], 2016.

[14] CROSS J, HUANG L. Incremental parsing with minimal features using bi-directional LSTM[C]//The 54th Annual Meeting of the Association for Computational Linguistics. 2016: 32-37.

[15] TURIAN J, RATINOV L, BENGIO Y. Word representations: a simple and general method for semi-supervised learning[C]//The 48th Annual Meeting of the Association for Computational Linguistics. 2010: 384-394.

[16] SUGAWARA H, TAKAMURA H, SASANO R, et al. Context representation with word embeddings for WSD[M]. Computational Linguistics. Springer. Singapore, 2015:108-119.

[17] SUNDERMEYER M, SCHLÜTER R, NEY H. LSTM neural networks for language modeling[J]. Interspeech,2012,31(43):601-608.

[18] WANG P L, QIAN Y, SOONG F K, et al. A unified tagging solution: bidirectional LSTM recurrent neural network with word embedding[J]. arXiv: 1511.00215v1[cs.CL], 2015.

[19] HUANG Z H, XU W, YU K. Bidirectional LSTM-CRF models for sequence tagging[J]. arXiv:1508.01991v1[cs.CL], 2015.

[20] WANG L, LUÍS T, MARUJO L, et al. Finding function in form: compositional character models for open vocabulary word representation[C]//The 2015 Conference on Empirical Methods in Natural Language Processing. 2016: 1520-1530.

[21] RAO A, SPASOJEVIC N. Actionable and political text classification using word embeddings and LSTM[J]. arXiv:1607.02501v1[cs.CL], 2016.

作者简介：

曾谁飞（1978-），男，江西广昌人，北京邮电大学博士生，主要研究方向为智能信息处理、机器学习、深度学习和神经网络等。

张笑燕（1973-），女，山东烟台人，博士，北京邮电大学教授，主要研究方向为软件工程理论、移动互联网软件、ad hoc 和无线传感器网络。

杜晓峰（1973-），男，陕西韩城人，北京邮电大学讲师，主要研究方向为云计算与大数据分析。

陆天波（1977-），男，贵州毕节人，博士，北京邮电大学副教授，主要研究方向为网络与信息安全、安全软件工程、P2P 计算。