文章编号：1003-0077(2017)03-0062-07

用于文本分类的局部化双向长短时记忆

万圣贤[1,2]，兰艳艳[1]，郭嘉丰[1]，徐 君[1]，庞 亮[1,2]，程学旗[1]

(1. 中国科学院 计算技术研究所，北京 100190；
2. 中国科学院大学，北京 100190)

摘 要：近年来，深度学习越来越广泛地应用于自然语言处理领域，人们提出了诸如循环神经网络(RNN)等模型来构建文本表达并解决文本分类等任务。长短时记忆(long short term memory，LSTM)是一种具有特别神经元结构的 RNN。LSTM 的输入是句子的单词序列，模型对单词序列进行扫描并最终得到整个句子的表达。然而，常用的做法是只把 LSTM 在扫描完整个句子时得到的表达输入到分类器中，而忽略了扫描过程中生成的中间表达。这种做法不能高效地提取一些局部的文本特征，而这些特征往往对决定文档的类别非常重要。为了解决这个问题，该文提出局部化双向 LSTM 模型，包括 MaxBiLSTM 和 ConvBiLSTM。MaxBiLSTM 直接对双向 LSTM 的中间表达进行 max pooling。ConvBiLSTM 对双向 LSTM 的中间表达先卷积再进行 max pooling。在两个公开的文本分类数据集上进行了实验。结果表明，局部化双向 LSTM 尤其是 ConvBiLSTM 相对于 LSTM 有明显的效果提升，并取得了目前的最优结果。

关键词：文本分类；深度学习；长短时记忆；卷积

中图分类号：TP391　　**文献标识码**：A

Local Bidirectional Long Short Term Memory for Text Classification

WAN Shengxian[1,2], LAN Yanyan[1], GUO Jiafeng[1], XU Jun[1], PANG Liang[1,2], CHENG Xueqi[1]

(1. Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China;
2. University of Chinese Academy of Sciences, Beijing 100190, China)

Abstract: Deep learning has shown great benefits for natural language processing in recent years. Models such as Recurrent Neural Networks (RNNs) have been proposed to extract text representation, which can be applied for text classification. Long short term memory (LSTM) is an advanced kind of RNN with special neural cells. LSTM accepts a sequence of words from a sentence scans over the whole sequence and outputs the representation of the sentence. However, customary practices use only the last representation LSTM produced for classification, ignoring all other intermediate representations. A clear drawback is that it could not capture efficiently local features that are very important for determining the sentence's class label. In this paper, we propose the local bidirectional long short term memory to deal with this problem, including MaxBiLSTM and ConvBiLSTM. MaxBiLSTM conducts a max pooling operation and ConvBiLSTM conducts a convolution operation followed with a max pooling operation on all intermediate representations generated by bidirectional LSTM. Experimental results on two public datasets for text classification show that local bidirectional LSTM, especially ConvBiLSTM, outperforms bidirectional LSTM consistently and reaches the state-of-the-art performances.

Key words: text classification; deep learning; long short term memory; convolution

收稿日期：2015-09-20　**定稿日期**：2015-12-30

基金项目：973 基金项目(2014CB340401，2012CB316303)；国家自然科学基金(6122010，61472401，61433014，61425016，61203298)；中国科学院青年创新促进会(2014310，2016102)

1　引言

近年来，深度学习越来越广泛地应用于自然语言处理领域。相对于传统的依赖于特征的机器学习方法，深度学习的目标是自动从原始数据中构建出更好的表达，从而降低人工提取特征的代价。目前深度学习已经在文本分类、语言模型及机器翻译等应用中取得了很好的效果[1-3]。

循环神经网络（recurrent neural network，RNN）是目前文本分类的一种常用深度学习模型。RNN 的特点是可以建模文本中包含的单词序列信息，并且支持变长的输入。然而，传统的 RNN 在使用后向传播进行训练的时候面临着梯度衰减和梯度爆炸等问题，这些问题导致传统的 RNN 捕获不到远距离的依赖，也就是说在某个位置学习到的表达只能捕获距离当前位置较近的信息。长短时记忆（long short term memory，LSTM）[4]是一种更高级的 RNN，通过在神经元上面添加门（gate）的方式更好地控制神经元中信息的读取和写入。由于每个神经元上的门打开时刻可以不同，所以 LSTM 既可以捕获远距离的依赖，同时也可以捕获近距离的依赖，此外，这种方式还可以更有效地过滤掉一些不重要的词，起到去噪的作用。基本的 LSTM 只沿着序列的一个方向扫描，为了更充分地捕获序列的模式信息，常用的一种方式是采用双向 LSTM（BiLSTM）[5]。BiLSTM 就是同时用两个 LSTM 沿着序列的两个方向进行扫描。目前 LSTM 在机器翻译、信息检索等任务中都优于传统的 RNN[3, 6-7]。

用于文本分类时，常规做法是使用 BiLSTM 扫描完整个句子后得到的最终表达来作为整个句子的表达，然后把表达交给分类器进行分类。这样做是因为期望 BiLSTM 在扫描完整个句子后能够在神经元中捕获文档包含的所有重要信息。但是，尽管门的机制让 BiLSTM 能够捕获到更长距离的依赖，BiLSTM 的这种贪婪（greedy）的合并方式从根本上说依然不可避免地导致一定程度的信息衰减。因此，BiLSTM 在捕获文本的局部特征上仍然不够高效，而这些局部特征往往对文本的类别起到至关重要的作用。以情感分类为例，如果一个句子的中间部分出现了“great”这个词，这是很强的情感信号，可是 BiLSTM 在读取到“great”单词之后依然要继续扫描剩下的单词，而在这个扫描的过程中，“great”的信息仍然不可避免地被弱化。

为了更好地提取文本分类需要的局部特征，本文提出局部化双向 LSTM（LBiLSTM）模型。LBiLSTM 借鉴了卷积神经网络（convolutional neural network，CNN）[2, 8]的优点来改进 BiLSTM。CNN 通过一组固定窗口的特征提取器对整个句子进行卷积操作，在句子的每个位置上提取该位置上的局部特征，这些局部特征最后通过 pooling（一般为 max pooling）的方式构建成整个句子的表达。Max pooling 的方式使得一些特别强的局部特征可以直接进入最终句子的表达。因此，相对于 RNN，CNN 的表达学习方式更侧重于提取局部的特征。本文提出的 LBiLSTM 具体采取两种做法，分别为 MaxBiLSTM 和 ConvBiLSTM。MaxBiLSTM 的做法是对 BiLSTM 得到所有位置上的表达直接进行 max pooling 操作，从而得到最终表达。这是一种较为直接的做法，但存在的问题是，每个位置上两个方向得到的表达是独立做 pooling 的，因此两个方向的 LSTM 不能有效地进行交互。而 ConvBiLSTM 先使用卷积操作把每个位置上两个方向的 LSTM 得到的表达综合起来构成该位置新的局部表达，然后再进行 max pooling。总的来说，LBiLSTM 的好处是，既结合了 BiLSTM 的优点，可以捕获较远距离的依赖关系，又结合了 CNN 的有效提取局部性特征的优点。

我们在两个公开的文本分类数据集上进行了实验，并和多种基准模型进行了对比。实验表明，LBiLSTM 尤其是 ConvBiLSTM 的一致性优于 BiLSTM，并且在这两个数据集上取得了当前的最好效果。

2　相关工作

深度学习被广泛地应用于文本表达学习并用于解决语言模型、文本分类、信息检索等任务，代表性的神经网络模型包括 RNN、递归神经网络（RvNN）和 CNN 等。

Socher 等人[1]首先将 RvNN 用于解决情感分析等任务，发现 RvNN 可以建模单词语义的组合关系（composition），通过对单词和短语表达的递归组合可以构建出整个句子的表达。Li 等人[7]也在多个文本分类数据集上对 RNN 和 RvNN 的效果进行了对比，发现在大部分数据集上，直接基于序列结构

的 RNN 也能表现出和 RvNN 类似的效果。此外，他们也发现通过在 RNN 中采用类似门的方式可以取得比传统 RNN 更好的效果[9]。

基本的 RNN 模型是单向的，为了同时得到某个位置左右两边的信息，Schuster 等人[5]首先提出使用双向 RNN。进一步的，双向 RNN 被用于解决文本分类任务及机器翻译，并通常能取得比传统的 RNN 更好的效果[7,10]。

传统 RNN 面临的优化问题很早就被人们所认识[11]，并提出了 LSTM[12]。近来，LSTM 被成功地应用于机器翻译[3]及信息检索中[13]。Palangi 等人[6]使用搜索引擎点击数据训练用于匹配查询和文档的 LSTM，并和相应的 RNN、CNN 版本做了对比分析，发现 LSTM 通过门的作用能够比传统的 RNN 更好地控制语义信息的读取与合并，最后效果也得到明显提升。Tai 等人[13]也在 RvNN 上采用了类似的神经元，从而得到基于树结构的 LSTM，并在情感分类数据集上取得了提升。

此外，为了解决 CNN 的固定窗口问题，Lai 等人[14]提出循环卷积神经网络(RCNN)，然而，和本文工作的区别是，这个工作并不是基于 LSTM，因此在捕获长距离依赖的时候不够高效。此外，RCNN 的动机主要是在进行卷积时为每个单词提供更多的上下文信息。

3 用于文本分类的循环神经网络

本节主要介绍与本文直接相关的几个用于文本分类的循环神经网络模型，包括 RNN、BiRNN、LSTM 及 BiLSTM。

3.1 RNN

RNN 在用于文本分类的时候，输入通常是一个单词序列$[w_1,w_2,w_3,\cdots,w_T]$，其中，假设 V 为整个词表空间，那么$w_i\in\mathrm{R}^{|V|}$是第 i 个位置上的单词的 one-hot 表达，也就是该单词在词表中对应的编号的那一维度为 1，其他维度全为 0。模型首先将单词通过分布式表达的思想表示为一个 d 维向量，也就是词嵌入(word embedding)，

$$[x_1,x_2,x_3,\cdots,x_T]=\boldsymbol{W}_e[w_1,w_2,w_3,\cdots,w_T]$$

其中，x_t 为 t 位置上单词的分布式表达，$\boldsymbol{W}_e\in\mathrm{R}^{d\cdot|V|}$词表中所有单词的表达向量构成单词表达矩阵。然后 RNN 沿着一个方向依次读入每个位置上的单词表达，并构建出该位置对应的中间表达。当两个方向的 RNN 都读取完所有单词后，把 T 位置(另一个方向的 RNN 把 1 位置)的表达输入分类器。分类器通常是一个 softmax 层或者一个多层感知机(MLP)。

3.2 RNN 与 BiRNN

传统的 RNN 的计算如式(1)所示。

$$h_t=g(Wx_t+Uh_{t-1}+b) \tag{1}$$

其中，$W\in\mathrm{R}^{|h_t|\cdot|x_t|}$，$U\in\mathrm{R}^{|h_t|\cdot|h_t|}$，$b\in\mathrm{R}^{|h_t|}$，为模型的参数。$h_t$ 为得到的 t 位置上的中间表达。g 为非线性函数，本文中我们采用 tanh 函数。这种 RNN 只沿着序列的一个方向进行扫描，并把 T 位置的表达作为整个句子的表达。为了能够更加充分地捕获单词序列的模式信息，BiRNN 同时使用两个 RNN 沿着单词序列的两个方向进行扫描，并把最终得到的两个表达拼接到一起作为整个句子的表达，双向 RNN 的计算如式(2)、式(3)所示。

$$\overrightarrow{h}_t=g(\overrightarrow{W}x_t+\overrightarrow{U}\overrightarrow{h}_{t-1}+\overrightarrow{b}) \tag{2}$$

$$\overleftarrow{h}_t=g(\overleftarrow{W}x_t+\overleftarrow{U}\overleftarrow{h}_{t+1}+\overleftarrow{b}) \tag{3}$$

其中，字母上面的箭头表示 RNN 的扫描方向。

3.3 LSTM 与 BiLSTM

LSTM 是一种递归神经网络(RNN)，传统的 RNN 非常难以训练，面临梯度爆炸或者梯度衰减问题，为了解决这个问题，LSTM 在传统 RNN 神经元的基础上增加了若干个门(gate)，用于控制每个神经元信息的读取和写入。RNN 加门的方式有很多变种，本文使用的 LSTM 如图 1 所示，具体的计算如式(4)～式(9)所示。

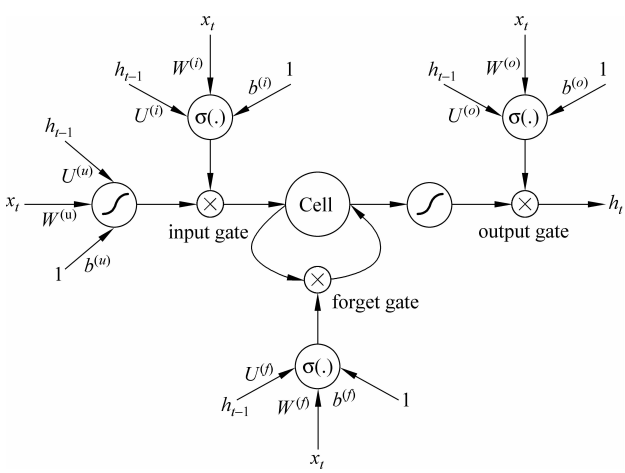

图 1 本文所使用的 LSTM 的神经元结构图

$$i_t=\sigma(W^{(i)}x_t+U^{(i)}h_{t-1}+b^{(i)}) \tag{4}$$

$$f_t=\sigma(W^{(f)}x_t+U^{(f)}h_{t-1}+b^{(f)}) \tag{5}$$

$$o_t = \sigma(W^{(o)} x_t + U^{(o)} h_{t-1} + b^{(o)}) \quad (6)$$

$$u_t = \tanh(W^{(u)} x_t + U^{(u)} h_{t-1} + b^{(u)}) \quad (7)$$

$$c_t = i_t \odot u_t + f_t \odot c_{t-1} \quad (8)$$

$$h_t = o_t \odot \tanh(c_t) \quad (9)$$

其中，所有的 $W \in \mathrm{R}^{k \cdot |x_t|}$，$U \in \mathrm{R}^{k \cdot k}$ 和 $b \in \mathrm{R}^{k \cdot 1}$ 为模型的参数，k 为 LSTM 神经元的个数。$i \in \mathrm{R}^{k \cdot 1}$ 是输入门，$f \in \mathrm{R}^{k \cdot 1}$ 是遗忘门，$o \in \mathrm{R}^{k \cdot 1}$ 是输出门，σ 为 sigmoid 函数，$\odot$ 表示向量对应元素相乘的操作(element-wise product)；$u \in \mathrm{R}^{k \cdot 1}$ 为上一位置的表达和当前读入的表达组合后形成的新表达。$c \in \mathrm{R}^{k \cdot 1}$ 为内部存储信息的神经元。具体的工作流程是这样的，LSTM 根据当前位置读取到的信息 x_t 和上一位置得到的表达 h_t 决定这几个门的打开和关闭情况，并计算出当前位置形成的新表达 u_t。如果输入门打开，则新表达会写入内部神经元，如果遗忘门打开，则神经元会对过去的信息进行衰减。更新完内部神经元之后，如果输出门打开，则从神经元中读取出信息并输出。由于每个神经元都有自己的门，而且不同神经元之间门打开的方式是不同的，所以一部分神经元可以捕获长距离的依赖，另一部分可以捕获短距离的依赖，LSTM 也因此得名。LSTM 通过对这些门的控制，有效地降低了 RNN 的梯度衰减问题，从而捕获远距离的依赖。

和 BiRNN 的方式类似，LSTM 也可以有对应的双向版本，即 BiLSTM，如图 2(a)所示。

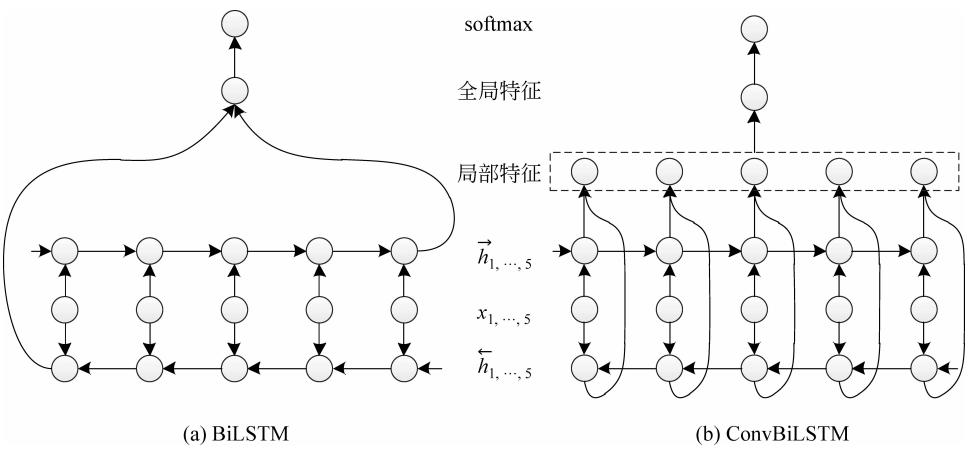

图 2　BiLSTM 和 ConvBiLSTM 模型结构图，卷积和 pooling 的方式构建了一条局部特征通往全局特征的路径

4　局部化双向 LSTM

尽管 LSTM 通过门的控制，相对于传统 RNN 能够捕获更长距离的依赖，但是信息衰减在扫描的过程中仍然不可避免地发生，这使得 LSTM 在捕获局部特征的时候不够高效。而在文本分类中，某些局部信息是很关键的。如果能在中间表达和最终表达之间直接建立一条直达的通道，那么当 LSTM 在扫描句子过程中遇到强的局部信息就可以直接进入最终的全局特征中，从而避免该特征在后续的扫描中受到干扰和衰减。为了达到这个效果，本文提出 MaxBiLSTM 和 ConvBiLSTM。这两个模型主要借鉴了 CNN 的优点，将 CNN 中的卷积和 pooling 的方式用在所有 BiLSTM 得到的中间表达上。这些中间表达比 LSTM 扫描完整个句子得到的表达具有更强的局部性。ConvBiLSTM 模型架构如图 2(b)所示。

4.1　MaxBiLSTM

MaxBiLSTM 直接对 BiLSTM 进行 max pooling 得到最终表达。max pooling 是一种在卷积网络中常用的 pooling 方式，可以有效地提取出具有局部不变性的特征。Max pooling 的作用机制正是适合提取强的局部特征，因为强特征在前向和后向传播(feedforward，backpropgation)过程中不会有任何的梯度衰减。具体方式为先将 $\overrightarrow{h_t}$ 和 $\overleftarrow{h_t}$ 进行拼接得到 h_t，然后对 $1 \cdots T$ 上所有的 h_t 进行 max pooling，

$$r_i = \max_{1,\cdots,T} h_{ti} \quad (10)$$

其中，r 为 max pooling 得到的句子表达。

4.2　ConvBiLSTM

MaxBiLSTM 的做法较为直接，但是存在的问题是，两个方向的 LSTM 得到的局部特征是独立做

pooling的，直到pooling之后这两个方向的特征才有交互(interaction)。因此，这种方式中两个方向的LSTM缺乏局部的交互，所以提取的局部特征仍然不够充分。

于是，在MaxBiLSTM的基础上，我们进一步提出ConvBiLSTM。ConvBiLSTM先采用卷积的方式在每个位置上将两个方向LSTM的局部表达合并为一个新的表达。由于每个位置上这两个表达分别捕获了该位置左侧的信息和右侧的信息，因此卷积之后的表达可以认为是以该位置为中心的局部化表达。然后，我们依然通过max pooling的方式得到最终的全局表达。

卷积的具体操作如下，在所有位置上$(1,\cdots,T)$对两个方向LSTM得到的表达做一个非线性变换，即

$$l_t = g(W_{ll}\overrightarrow{h}_t + W_{lr}\overleftarrow{h}_t + b_l) \tag{11}$$

其中，$W_{ll}\in \mathrm{R}^{|l_t|\cdot|\overrightarrow{h}_t|}$，$W_{lr}\in \mathrm{R}^{|l_t|\cdot|\overleftarrow{h}_t|}$，$b_l\in R^{|l_t|}$是卷积层的参数，$l_t$是在$t$位置上的局部表达。得到所有位置上的局部表达$(l_1,\cdots,l_T)$之后，对其进行max pooling，即

$$r_i = \max_{1\cdots T}(l_{ti}) \tag{12}$$

以上两种模型得到句子表达之后，我们使用softmax层来得到每个类别的概率，即

$$s = W_{\mathrm{sm}}r + b_{\mathrm{sm}} \tag{13}$$

$$p_i = \frac{e^{s_i}}{\sum_{j=0}^{c} e^{s_j}} \tag{14}$$

其中，$W_{\mathrm{sm}}\in \mathrm{R}^{|c|\times|r|}$和$b_{\mathrm{sm}}\in \mathbb{R}^{|c|}$为softmax层的参数，$|c|$为类别数，$p_i$为第$i$个类别的概率。

4.3 模型训练

我们使用最大熵损失函数，并在除单词表达之外的其他所有模型参数上面使用权重衰减(weight decay)来对参数进行正则化。损失函数为

$$\mathrm{Loss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{c} y_{ik}\log p_{ik} + \lambda \|\theta\|_2^2 \tag{15}$$

其中，N为训练样本数，c为数据集包含的类别数，λ为正则化系数，θ为所有参数。本文中我们使用时序后向传播(back propogation through time，BPTT)来对网络进行训练。这是训练RNN常用的训练方法。LSTM神经元的求导过程较为复杂，具体公式可以参见文献[6]。

5 实验

我们在两个公开的常用文本分类数据集上对比了现有的多个模型，包括前述的BiLSTM、MaxBiLSTM、ConvBiLSTM，以及相应的RNN版本(BiRNN、MaxBiRNN、ConvBiRNN)。为了更有效地进行模型之间的对比，我们在数据的预处理上及其他的实验设置上尽可能采用了和文献[2]类似的方式。

5.1 数据集

我们在电影评论(Movie Reviews，MR)①[15]和TREC②[16]两个文本分类数据集上进行了实验。MR是情感分类数据集，这个数据集包含了从电影评论网站上采集到的电影评论数据，这些评论的标签信息是根据和该评论同时发布的用户对电影的评分决定的。TREC数据集是一个问题分类数据集，主要是把一个疑问句根据问题的类型分成多个类别。由于MR数据集并没有切分好的训练集和测试集，因此这个数据集上我们采用了十折交叉验证的方式得到最终结果，具体切分方式和文献[2]一样。预处理之后的数据情况见表1。

表1 MR和TREC两个数据集的相关信息

数据集	类别数	文档数	词表大小	测试集
MR	2	10 662	18 765	交叉验证
TREC	6	5 952	9 592	500

5.2 优化方法与参数设置

RNN及LSTM的表达维度设置为50。通常如果维度设置过小学到的表达区分度不够，效果较差，维度设置太大则容易过拟合。我们分别对RNN和LSTM尝试了若干维度，发现50维之后效果就没有明显提升，因此全部都设置成50维。关于模型的训练方式，我们采用了文献[2]中使用的AdaDelta[17]方法。这种方法相对于传统的随机梯度下降方法具有自适应的学习速率，通常也能取得比传统SGD更好的效果。BatchSize及正则化因子是根据验证集上的表现人工调整的。由于深度学习在小数据集上非常容易过拟合，因此本文也采用了early stopping的方式，根据验证集上的效果决定训练终止的时刻。

使用在无监督数据上预训练的单词表达来对监

① http://www.cs.cornell.edu/people/pabo/movie-review-data/

② http://cogcomp.cs.illinois.edu/Data/QA/QC/

督模型进行初始化是一种常用的做法，尤其是在小数据集上通常能取到非常大的提升。本实验室中，我们使用 word2vec[18] 公开发布的 300 维度的单词向量①，如果实验数据集中的单词没有在 word2vec 的词表中出现，我们对其进行随机初始化。所有单词表达在模型训练过程中和模型其他参数一样同时被更新。

5.3　实验结果

本文主要对比了以下基准方法。

(1) NBSVM 和 MNB[19]：使用 uni-gram 和 bi-gram 作为特征的朴素贝叶斯 SVM 及多项式朴素贝叶斯模型。

(2) SVM_S[20]：使用了 uni-gram、bi-gram、tri-gram，以及 60 多个人工提取的特征训练的 SVM 模型。

(3) MV-RvNN[21]：基于语法树的 Matrix-Vector Recursive Neural Network 模型。

(4) CNN[2]：具有单个卷积和 pooling 层的卷积神经网络。

(5) DCNN[8]：用于文本表达学习的深层卷积神经网络。

表 2　各数据集上的准确率

数据集	MR/%	TREC/%
NBSVM	79.4	—
MNB	79.0	—
MV-RvNN	79.0	—
SVM_S	—	95.0
CNN	81.5	93.6
DCNN	—	93.0
BiRNN	79.9	91.8
BiLSTM	81.5	93.4
MaxBiRNN	80.4	93.0
ConvBiRNN	81.2	94.6
MaxBiLSTM	81.9	93.6
ConvBiLSTM	82.3	95.2

从结果中可以看出，深度学习模型的效果已经开始超越传统的基于特征的机器学习方法。局部化 LSTM 和 RNN 的效果一致性优于只使用最后一个表达的做法。其中，ConvBiLSTM 表现更为突出，明显超过了传统的 CNN、RNN 及 LSTM。ConvBiLSTM 还在 TREC 数据集上超过了基于复杂特征工程的方法，取得了新的最优结果。这也充分说明局部特征对于文本分类的重要性。相对于 BiRNN，BiLSTM 的结果有着大幅提升，这和目前其他论文中的实验结果一致，这也再次说明 LSTM 的神经元结构可以有效地克服 RNN 的缺点。相对于 CNN，BiLSTM 取得了相近的结果，但是却只用了更低的维度和更少的参数。相对于 MaxBiLSTM，ConvBiLSTM 进一步取得了效果提升，说明 ConvBiLSTM 中的卷积层能够有效地对两个方向的 LSTM 表达进行组合从而得到更好的局部化表达。

6　总结和未来工作

LSTM 目前已经成为自然语言处理领域的常用模型，并在诸如语言模型、机器翻译及文本分类等应用中取得很好的效果。在用于文本分类时，传统的 LSTM(包括双向 LSTM)通常对整个句子进行扫描，并把扫描结束时的表达作为整个句子的表达输入到分类器，这种做法并没有充分利用扫描过程得到的中间表达，因此并不能高效地提取一些对分类至关重要的局部信息。本文提出的局部化 LSTM 在双向 LSTM 的中间表达上进一步进行卷积和 pooling 操作，这种做法结合了 LSTM 和 CNN 的优点，可以有效克服上述问题。我们在两个数据集上的实验结果表明，这种做法的一致性取得了比传统做法更好的效果。作为下一步工作，我们准备把这种结构应用到更多的文本相关的应用中，并对其内部机制进行更深入的分析。

参考文献

[1] Richard Socher, Alex Perelygin, Jean Y Wu, et al. Recursive deep models for semantic compositionality over a sentiment Treebank[C]//Proceedings of EMNLP, 2013: 1631-1642.

[2] Yoon Kim. Convolutional neural networks for sentence classification[arXiv]. 2014. arXiv preprint arXiv: 1408.5882.

[3] Ilya Sutskever, Oriol Vinyals, Quoc Le. Sequence to sequence learning with neural networks[C]//Proceedings of NIPS, 2014:3104-3112.

[4] Sepp Hochreiter, Jürgen Schmidhuber. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.

① https://code.google.com/p/word2vec/

[5] Mike Schuster, Kuldip K Paliwal. Bidirectional recurrent neural networks[J]. IEEE Transactions on Signal Processing, 1997, 45(11): 2673-2681.

[6] Hamid Palangi, Li Deng, Yelong Shen, et al. Deep sentence embedding using the long short term memory network: analysis and application to information retrieval[arXiv]. 2015. arXiv preprint arXiv: 1502.06922.

[7] Jiwei Li, Dan Jurafsky, Eudard Hovy. When are tree structures necessary for deep learning of representations? [arXiv]. 2015. arXiv preprint arXiv: 1503.00185.

[8] Nal Kalchbrenner, Edward Grefenstette, Phil Blunsom. A convolutional neural network for modelling sentences[arXiv]. 2014. arXiv preprint arXiv: 1404.2188.

[9] Jiwei Li. Feature weight Tuning for recursive neural networks[arXiv]. 2014. arXiv preprint arXiv: 1412.3714.

[10] Martin Sundermeyer, Tamer Alkhouli, Joern Wuebker, et al. Translation modeling with bidirectional recurrent neural networks[C]//Proceedings of the Conference on Empirical Methods on Natural Language Processing, 2014:14-25.

[11] Yoshua Bengio, Patrice Simard, Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult[J]. IEEE Transactions on Neural Networks, 1994, 5(2): 157-166.

[12] Yelong Shen, Ruoming Jin. Learning personal social latent factor model for social recommendation[C]//Proceedings of KDD, 2012:1303-1311.

[13] Kai Sheng Tai, Richard Socher, Christopher D Manning. Improved semantic representations from tree-structured long short-term memory networks[arXiv]. 2015. arXiv preprint arXiv:1503.00075.

[14] Siwei Lai, Liheng Xu, Kang Liu, et al. Recurrent convolutional neural networks for text classification[C]//Proceedings of AAAI, 2015.

[15] Bo Pang, Lillian Lee. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales[C]//Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, 2005:115-124.

[16] Xin Li, Dan Roth. Learning question classifiers[C]//Proceedings of the 19th International Conference on Computational Linguistics, 2002:1-7.

[17] Matthew DZeiler. ADADELTA: An adaptive learning rate method[arXiv]. 2012. arXiv preprint arXiv: 1212.5701.

[18] Tomas Mikolov, Ilya Sutskever, Kai Chen, et al. Distributed representations of words and phrases and their compositionality[C]//Proceedings of NIPS, 2013:3111-3119.

[19] Sida Wang, Christopher D Christopher. Baselines and bigrams: Simple, good sentiment and topic classification[C]//Proceedings of Annual Meeting of the Association for Computational Linguistics, 2012:90-94.

[20] Joao Silva, Luísa Coheur, Ana Cristina Mendes, et al. From symbolic to sub-symbolic information in question classification[J]. Artificial Intelligence Review, 2011,35(2):137-154.

[21] Richard Socher, Brody Huval, Christopher D Manning, et al. Semantic compositionality through recursive matrix-vector spaces[C]//Proceedings of Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, 2012.

万圣贤(1989—),博士研究生,主要研究领域为深度学习与文本挖掘。
E-mail: wanshengxian@163.com

郭嘉丰(1980—),博士,副研究员,主要研究领域为信息检索与数据挖掘。
E-mail: guojiafeng@ict.ac.cn

兰艳艳(1982—),博士,副研究员,主要研究领域为统计机器学习、排序学习和信息检索。
E-mail: lanyanyan@ict.ac.cn