基于递归神经网络的文本分类研究

黄 磊 杜昌顺
(北京交通大学 经济管理学院, 北京 100044)

摘 要: 使用基于长短项记忆(LSTM)和门阀递归单元(GRU)计算节点的双向递归神经网络提取文本特征,然后使用 softmax 对文本特征进行分类。这种基于深度学习的神经网络模型以词向量作为基本输入单元,充分考虑了单词的语义和语法信息,并且在神经网络的计算过程中严格遵守单词之间的顺序,保留原文本中语义组合的方式,可以克服传统文本分类方法的不足。使用本文所提方法在第三届自然语言处理和中文计算会议(NLPCC 2014)公布的新华社新闻分类语料和路透社 RCV1-v2 语料上进行实验,其分类 $F1$ 值分别达到了 88.3% 和 50.5%,相较于传统的基线模型有显著的提升。由于该方法不需要人工设计特征,因此具有很好的可移植性。

关键词: 文本分类; 深度学习; 长短项记忆(LSTM); 门阀递归单元(GRU); 双向递归神经网络; 词向量
中图分类号: TP391.1 **DOI**: 10.13543/j.bhxbzr.2017.01.017

引 言

在步入信息时代的今天,互联网以惊人的速度蓬勃发展,产生了海量的文本数据。如何对这些文本数据进行有效的文本分类,进而发现有价值的信息一直是人们研究的热点。

当前,针对文本分类方法已经出现了许多研究。姚全珠等[1]使用 latent dirichlet allocation (LDA)模型对文本进行自动分类,将文本表示为固定的概率分布,利用 Markov chain Monte Carlo (MCMC)中的 Gibss 抽样进行推理,以间接的方式计算模型的参数,从而获得文本在固定主题上的概率分布,概率大的对应为文本的类别。张爱丽等[2]使用支持向量机(SVM)算法进行多类别的文本分类,该方法主要使用向量空间模型,以此作为特征项,将文档构造成一个高维度、稀疏的向量作为文本的特征表示,然后输入到 SVM 分类器中。刘华[3]使用文本的关键短语进行分类,该方法认为反映文本类别信息的关键单词或者短语的作用更加重要,因此先用统计的方法抽取关键短语的向量特征,然后通过计算余弦相似度来判断类别。随着近年来深度学习方法的兴起,受限玻尔兹曼机也被广泛地应用到文本分类的方法中来。Hinton 等[4]利用深度玻尔兹曼机模拟文档,自动学习文档的分类特征,在英文文档的分类上取得了良好的效果。

尽管上述方法在一些实验中已经取得了一定的效果,但是存在两个主要的问题: 第一,它们通常都是将文本看作由许多单词构成的无机体,认为各个单词是相互独立的,并且忽略其顺序; 第二,这些方法仅仅是把单词看作一个符号,记录文本中有无出现该符号以及该符号对某主题(类别)的贡献率,而忽略单词本身所代表的语义。然而,文本中各个单词之间是相互联系的,共同出现才能构成文本所表达的完整语义,并且其顺序非常重要。其次,文本的语义是由单词的语义组合得到的,如果不能准确捕获单词的语义,那么也难以获取文本的准确语义。

针对这两个方面,本文设计了基于 long short-term memory (LSTM)[5]和 gated recurrent unit (GRU)[6]的递归神经网络[7]结构,以 LSTM 或 GRU 为计算单元的递归神经网络在处理长句子或者长文本时有独特的优势,它能够记住句子中远距离的依赖关系,使得网络能够保留文本的主要语义信息。为了验证基于 LSTM 和 GRU 的递归神经网络模型的正确性和有效性,收集了中文和英文的新闻分类的数据集[4,8]。运用递归神经网络抽取新闻的特征向量,最后将特征向量传递给 softmax 分类器,并对分类结果进行了比较分析。

1 基于 LSTM 和 GRU 节点的递归神经网络模型

模型主要包括两个部分: 第一部分是特征提

收稿日期: 2016-09-21
第一作者: 男,1965 年生,教授
E-mail: summer2015@bjtu.edu.cn

取操作,主要是使用递归神经网络逐步合成文本的向量特征,本文首先介绍传统的递归神经网络,然后详细描述 LSTM 和 GRU 计算节点的递归神经网络; 第二部分是分类器,主要由 dropout 层[9] 和 softmax 层构成。该模型的最大优点是只需要对文本进行简单的预处理,即可同时进行特征和分类的学习。

1.1　单词的表示

在文本中,将单词表示为分布式的词向量,目前已经有许多研究工作提出了在向量空间中学习单词的表示方法[10-14],本文选用文献[12]提出的语言模型学习单词的表示。首先在百度百科上收集无监督的文本语料以及纽约时报语料,预训练单词的词向量。文献[15-17]中指出,在大规模无监督的语料上学习得到的词向量可以改善模型的效果,为模型提供一个较好的初始值。在本文中,使用 E 表示词向量的矩阵,其每一列代表一个单词的向量,列向量的维度为 d。将第 k 个单词表示为二元向量 v_k (第 k 个位置为 1,其余位置为 0),则第 k 个单词的向量可表示为 Ev_k。

1.2　递归神经网络(RNN)

RNN 模型已经在很多自然语言处理任务中表现出强大的学习能力,它的特点是可以对序列数据进行很好的建模,能够充分地利用序列的信息。由于 RNN 是依次对文本中的每个单词进行语义合成,因此它可以适应变长的句子,即不要求文本长度的统一,对长文本和短文本皆可以学习。图 1 给出了一个传统的递归神经网络结构。

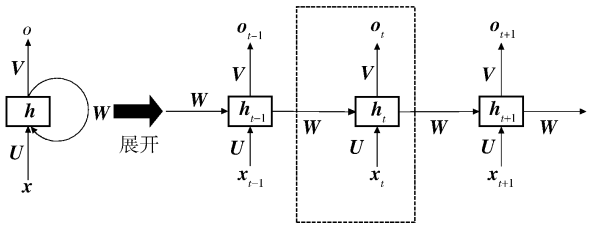

图 1　传统的递归神经网络结构

Fig. 1　The traditional recurrent neural network structure

图 1 中,x_t 是第 t 步的输入单元,在文本中,它代表第 t 个单词的词向量; h_t 是第 t 步的隐藏状态; o_t 表示第 t 步的输出,通常这一步输出是一个 softmax 分类器,该输出是否选用可根据模型的需要确定; U,V 和 W 是网络的权重参数,需要在模型中学习得到。如图 1 所示,虚线方框内是第 t 个单元的计算过程,具体如下

$$\begin{cases} \boldsymbol{h}_t = f(\boldsymbol{W}\boldsymbol{h}_{t-1} + \boldsymbol{U}\boldsymbol{x}_t + \boldsymbol{b}_h) \\ \boldsymbol{o}_t = \mathrm{softmax}(\boldsymbol{V}\boldsymbol{h}_t + \boldsymbol{b}_o) \end{cases} \tag{1}$$

这里变量 b_h 和 b_o 表示偏置项。由公式(1)可知,递归神经网络的每个隐藏状态由当前的输入词和前一步的隐藏状态决定。如果在特定的任务中,不需要对每个合成步骤都附加分类器,则 o_t 可不输出。传统递归神经网络的缺点是随着文本长度的增加,网络的层数逐渐加深,网络在信息合成的过程中损失比较大,往往偏重于记忆最后阶段内容的学习,因此对长文本学习效果欠佳。

1.3　LSTM 和 GRU 计算节点

由于传统的 RNN 在处理长文本时存在缺陷,因此本文采用 LSTM 和 GRU 节点作为 RNN 的计算节点。LSTM 和 GRU 节点的优势是可以在合成的过程中设置一些门来控制当前合成步骤中应当对前面信息接收多少,遗忘多少,并且向后面传递多少信息。通过这些门域的控制,RNN 对长文本具有很好的学习能力。LSTM 与 GRU 不同之处在于 LSTM 具有更多的参数,GRU 参数较少,因而具有更快的计算速度。因此,对于大数据学习而言,LSTM 节点具有更强的学习能力,在后续的实验部分本文将验证这一点。

LSTM 和 GRU 是 RNN 的两种计算节点,在计算隐藏状态的方法上较传统方法不同,但在主体结构上与 RNN 一致。LSTM 和 GRU 两种节点的计算方法如图 2 所示。

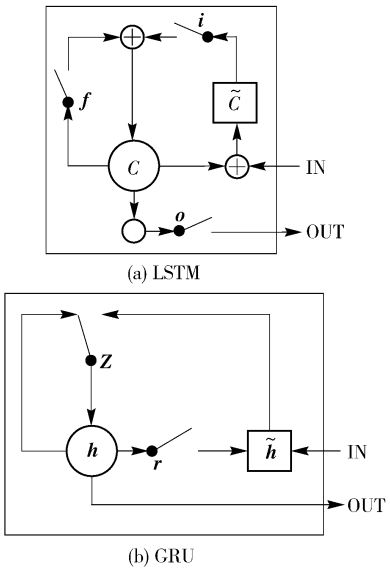

图 2　LSTM 和 GRU 单元结构图

Fig. 2　Structures of LSTM and GRU units

LSTM 节点计算方式

$$\begin{cases} \boldsymbol{i} = \sigma(\boldsymbol{U}^i \boldsymbol{x}_t + \boldsymbol{W}^i \boldsymbol{h}_{t-1}) \\ \boldsymbol{f} = \sigma(\boldsymbol{U}^f \boldsymbol{x}_t + \boldsymbol{W}^f \boldsymbol{h}_{t-1}) \\ \boldsymbol{o} = \sigma(\boldsymbol{U}^o \boldsymbol{x}_t + \boldsymbol{W}^o \boldsymbol{h}_{t-1}) \\ \boldsymbol{g} = \tanh(\boldsymbol{U}^g \boldsymbol{x}_t + \boldsymbol{W}^g \boldsymbol{h}_{t-1}) \\ \boldsymbol{c}_t = \boldsymbol{c}_{t-1} \otimes \boldsymbol{f} + \boldsymbol{g} \otimes \boldsymbol{i} \\ \boldsymbol{h}_t = \tanh \boldsymbol{c}_t \otimes \boldsymbol{o} \end{cases} \tag{2}$$

GRU 节点计算方式

$$\begin{cases} \boldsymbol{z} = \sigma(\boldsymbol{U}^z \boldsymbol{x}_t + \boldsymbol{W}^z \boldsymbol{h}_{t-1}) \\ \boldsymbol{r} = \sigma(\boldsymbol{U}^r \boldsymbol{x}_t + \boldsymbol{W}^r \boldsymbol{h}_{t-1}) \\ \boldsymbol{s} = \tanh(\boldsymbol{U}^s \boldsymbol{x}_t + \boldsymbol{W}^h(\boldsymbol{h}_{t-1} \otimes \boldsymbol{r})) \\ \boldsymbol{h}_t = (1 - \boldsymbol{z}) \otimes \boldsymbol{s} + \boldsymbol{z} \otimes \boldsymbol{h}_{t-1} \end{cases} \tag{3}$$

如图 2 所示，在 LSTM 节点中，i，f，o 分别是输入门、记忆门和输出门，这些门控制着信息通过的比例。对于 GRU 节点，图 2 中 z 是更新门，r 是重置门，它们分别控制着信息通过的比例。在公式(2) 中，g 是候选的隐藏状态，与传统 RNN 中计算隐藏状态的方式类似; c_t 是内部记忆，由 $t-1$ 步的记忆 c_{t-1} 和 g 通过记忆门和输入门加权构成; h_t 为真实的输出状态，是内部记忆 c_t 在输出门输出的信息量。式(3) 中 σ 代表 sigmoid 函数; 符号⊗表示向量对应元素相乘的运算; s 是当前候选的隐藏状态，通过 s 的计算方式可以看出，重置门控制前一个节点信息 h_{t-1} 被保存的量，最后输出状态 h_t 由当前候选的隐藏状态 s 和前一个节点输出状态 h_{t-1} 通过更新门 z 进行加权得到。

1.4 双向 RNN 模型结构

本文采用双向 RNN 学习文本的特征，因为一个单词的含义不仅与它前面的文本内容有关，而且与它后面的文本内容也相关。本文使用双向的 RNN 进行文本的表示学习，然后将两个方向学习到的特征向量拼接在一起，作为文本的向量表示，这样相对于单向的 RNN，其特征向量所表示的语义更加全面和丰富。图 3 展示了模型的整体架构。

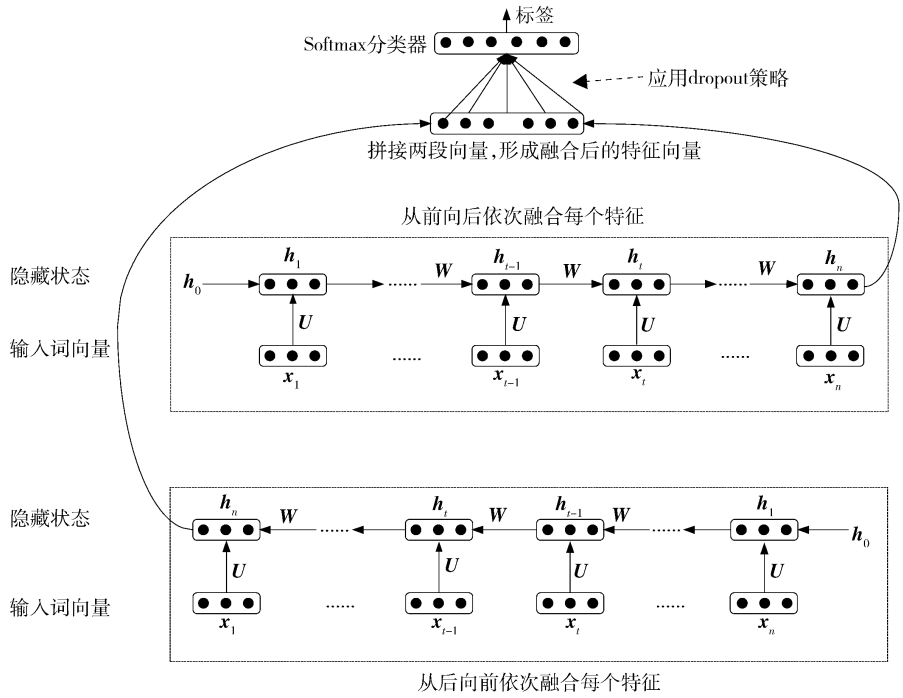

图 3 双向递归神经网络结构

Fig. 3 The bidirectional recurrent neural network structure

1.5 softmax 分类器

如图 3 所示，在双向递归神经网络提取文本的特征之后，将这些特征向量输入 softmax 分类器进行分类。本文使用 dropout 方式连接特征向量和 softmax 分类器。传统的神经网络的连接方式是全连接方式，dropout 算法的连接方式是随机地将输入数据(在本文中是拼接后的特征向量) 按照一定比例 ρ 置 0，只有其他没有置 0 的元素参与运算和连接。为了方便说明，假设一个学习样本更新一次参数，具体过程如下: 首先对输入的向量按照比例 ρ 置 0 其

中的部分元素,没有置 0 的元素参与分类器的运算和优化; 然后接受第二个样本的输入向量,此时同样按照随机置 0 的方式选择参与训练的元素,直到所有的样本都学习过一次。由于每次输入一个样本,置 0 的方式都是随机的,因此每次更新的网络权重参数都不一样。在最终预测的过程中,将整个网络的参数乘以 $1-\rho$,就得到了最终的分类器网络参数。因为每次更新的参数都不相同,因此 dropout 算法可以看作是把神经网络变成了多个模型的组合,可以有效地防止过拟合和提升模型的预测确率[9]。根据文献[9]的观点,dropout 算法类似于进化论,后代的基因是由父母两方各一半的基因组合而成,这种组合有产生更加优秀基因的倾向,与此类似,dropout 算法的最终网络参数是由多个模型的参数组合而成,是一个取精去糟的过程,因而具有更好的泛化能力。

假设双向递归神经网络得到的向量为 $\boldsymbol{c}$,dropout 算法将其元素置 0 的方式可以用伯努利分布 B 表示。先使用伯努利分布产生与 $\boldsymbol{c}$ 等维度的二元向量(元素只有 0 或者 1) $\boldsymbol{r}$

$$\boldsymbol{r} \sim B(\rho)$$

输入到 softmax 分类器的向量记为

$$\boldsymbol{c}_{\mathrm{d}} = \boldsymbol{c}\boldsymbol{r}$$

记 softmax 分类器的网络参数为 $\boldsymbol{W}_{\mathrm{c}}$ 和偏置项为 $\boldsymbol{b}_{\mathrm{c}}$,则网络的输出为

$$\boldsymbol{o} = f(\boldsymbol{W}_{\mathrm{c}}\boldsymbol{c}_{\mathrm{d}} + \boldsymbol{b}_{\mathrm{c}})$$

其中 f 为 sigmoid 函数或者 tanh 函数。则当前文本属于第 i 个类别的概率为

$$p(i \mid S) = \frac{e^{o_i}}{\sum_{j=1}^{N} e^{o_j}}$$

其中 o_i 表示向量 $\boldsymbol{o}$ 的第 i 个元素,N 表示类别数。

1.6　目标函数

本文的主要目的是研究文本分类问题,需要优化的参数包括两个部分: 词向量和双向递归神经网络的参数。词向量用 $\boldsymbol{E}$,双向 RNN 的参数用 $\hat{\boldsymbol{W}}$ 和 $\hat{\boldsymbol{U}}$ 表示,分类器的参数用 $\boldsymbol{W}_{\mathrm{c}}$ 表示,记 $\theta = \{\boldsymbol{E}, \hat{\boldsymbol{W}}, \hat{\boldsymbol{U}}, \boldsymbol{W}_{\mathrm{c}}\}$,训练集的样本集合为 $\Omega = \{(T_1, y_1), (T_2, y_2), \cdots\}$,其中 T_i 表示第 i 个文本,y_i 表示它的类别标签,$|\Omega|$ 表示训练集样本的个数。$p(y_i \mid T_i, \theta)$ 表示在已知参数 θ 时将文本 T_i 的类别分为 y_i 的概率,则优化的目标函数为

$$L = \sum_{i=1}^{|\Omega|} \lg p(y_i \mid T_i, \theta) + \lambda \| \theta \|_2^2 \tag{4}$$

其中 λ 为正则项的参数。在实验中,采用随机梯度下降法优化该目标函数,则参数 θ 的更新方式为

$$\theta = \theta - \alpha \frac{\partial L}{\partial \theta}, \alpha \text{ 为学习率。} \tag{5}$$

2　实验及讨论

2.1　实验数据

实验中主要使用两个数据集。第一个是中文数据集,为 2014 年中文计算机学会举办的自然语言处理会议所公布的新闻分类评测数据集,由新华社负责整理和标注,是较大规模的新闻分类语料,可以从第三届自然语言处理和中文计算会议(NLPCC 2014)的官网上直接下载。语料训练集规模为 30000 篇新闻,测试集包含 11577 篇新闻,测试集和训练集在各个类别的分布上具有很好的一致性。该数据集共有两个类别层次,第一层类别有 24 类,第二层类别有 367 类。在本文中,文本的类别统一看作单层次类别,对于树状的多类别层次,则将最终的小类别看作单独一个类别,并且均报告这个最终类别的分类结果。第二个数据集是英文数据集,即路透社公布的 RCV1 - v2 数据集,该数据集包含 804414 篇新闻,共有 103 个主题(103 个类别,此处也是层次分类,处理方式同前一个数据集),仿照文献[4],本文把数据集随机地分成训练集和测试集,其中训练集包含 794414 篇新闻,测试集包含 1000 篇新闻。本文中所有的实验都在这两个数据集上进行。

2.2　数据前处理

对于中文数据集,首先使用中科院计算所开发的中文分词软件包 NLPIR 进行中文分词。NLPIR 的功能包括中文分词、词性标注、命名实体识别、用户词典,支持多种中文编码格式,并且具有新词发现和关键词提取等功能。由于本文的实验中有中文数据集,因此需要调用该软件包分词。英文数据本身就是独立的单词,因此不需要分词的操作。分词操作完成后,对这两个数据集的词频进行统计,删除低频率的词和停止词,因为这些词对主题的判断没有帮助,在分类时使用它们可能会带来额外的噪声,不利于分类模型的预测。

由于在训练的过程中使用 minibatch 训练模型(一次学习多个样本,多个样本的文本长度可能不相同),因此需要对文本的长度进行固定长度的操作。由于自然语言文本的长度不一致,首先计算出

最长的句子长度 l_{max},对于句子长度小于 l_{max}的句子,统一使文本用 < \s > 符号补齐到长度 l_{max}(< \s > 的向量始终设置为 0),这样就统一了文本长度。此做法的目的是为了提高计算效率,当数据的长度统一时,可以使用矩阵计算,比使用循环计算节省时间。

2.3 词向量的预训练

在进行模型训练之前,需要在无监督的大规模语料上预训练词向量。词向量是单词的一种分布式表示,这种分布式表示适合神经网络的输入。当前的许多研究都显示了在大语料上无监督学习的词向量更有利于神经网络模型收敛到一个好的局部最优解。本文使用 Skip-gram 模型预训练词向量,这个模型学习的词向量在许多自然语言处理任务中都有很强的表现。Skip-gram 算法已经集成在 word2vec 软件包中,可直接使用该软件包训练中文和英文单词的词向量。中文词向量的预训练使用百度百科上读取的文本内容,英文词向量的预训练在 New York Times 语料上进行。

2.4 实验参数的设置

在本文中,模型主要有以下参数:词向量的维度 d,递归神经网络中隐藏状态的维度 n,dropout 算法的比率 ρ,SGD 优化算法的学习率 α。本文采用网格搜索的方法确定这些参数。词向量的维度 d 在 {50,100,200,300} 中取值;隐藏状态的维度 n 在 {500,1000,2000} 中取值;dropout 算法的比率 ρ 根据经验取值为 0.5;SGD 算法的学习率在{1,0.1,0.01,0.001}中取值。这些参数的取值范围是根据经验而定的,一般在此范围内取值可以取得比较好的实验结果。根据文献[4],采用类似的 5 倍交叉验证的方法选取网络参数,对于新华社新闻数据集,最佳的参数值为:$d=100$,$n=1000$,$\alpha=0.1$。对于路透社 RCV1-v2 数据集,最佳的参数值为:$d=300$,$n=1000$,$\alpha=0.01$。在本文实验中,使用这些参数进行多次实验,然后得到结果的平均值。对于 LSTM 和 GRU 计算节点,使用同样的参数可以增加可比性。

2.5 数据实验及对比分析

本文设计了基于 LSTM 和 GRU 的双向递归神经网络来处理文本分类的问题。以 LSTM 或 GRU 为计算单元的递归神经网络在处理长句子或者长文本时有独特的优势,它能够记住文本中远距离的依赖关系,使得网络经过多层的合成计算之后仍可保留文本的主要语义信息;并且在神经网络模型的架构中,还可以结合词向量学习文本的语义,因为词向量能够很好的捕捉到词的语义和语法信息,可以为模型带来更好的语义特征。在实验中,对中文选择高频的 20000 个单词进行向量化,对英文数据集选择高频的 100000 个单词进行向量化。表 1 列出了一些基线模型和本文提出模型的结果。

表 1 中英文数据集的文本分类测试结果

Table 1 Test results of document classification of Chinese and English data sets

模型	精度/%		召回率/%		F1 值/%	
	新华社(367 类)	路透社 RCV1-v2(103 类)	新华社(367 类)	路透社 RCV1-v2(103 类)	新华社(367 类)	路透社 RCV1-v2(103 类)
TF-IDF + SVM	72.1	31.8	88.7	45.8	79.5	31.8
AveVec + SVM	70.8	29.3	92.3	33.2	80.1	29.3
TRNN	74.4	40.5	90.7	51.7	81.7	40.5
LDA	77.3	35.1	93.3	44.3	84.5	35.1
DocNADE	76.5	41.7	84.5	42.5	80.3	41.7
Replicated Softmax	82.3	42.1	94.0	47.2	87.8	42.1
Over-Rep. Softmax	82.7	45.3	89.3	51.4	85.9	45.3
Bi-TRNN	82.4	44.8	91.1	52.5	86.5	44.8
LSTM Bi-RNN	81.9	46.2	92.8	55.8	87.0	46.2
GRU Bi-RNN	83.3	45.8	93.9	51.9	88.3	45.8

为了验证本文模型的有效性和正确性,将它与一些基线系统的方法进行了比较。对于 RNN 的基线方法,使用 2.4 节的实验参数;对于非 RNN 并且需要设置超参数的方法,使用相关软件包的默认参数设置。第一种比较的方法是统计文本的词频-逆文档频率(TF-IDF)特征,构成一列向量,然后使用 SVM 对特征向量分类;第二种方法是使用文本词向量的平均值(文本经过预处理后,计算所有单词向量的均值);第三种是传统递归神经网络(TRNN)方法,该方法是仅由一个从左至右方向的递归合成过

程构成的; 第四种方法是调用 matlab 的 LDA 算法包对文本进行分类; 第五种方法是神经自回归密度估计方法; 第六和第七种方法是使用深度玻尔兹曼构建的 restricted Boltzmann machine(RMB) 模型[4]，对 softmax 分类器进行相应的变换; Bi-TRNN 是在传统的 RNN 的基础上，使用了双向网络的结果; LSTM Bi-RNN 和 GRU Bi-RNN 是本文中提出的方法，其中 LSTM 和 GRU 分别是计算节点，Bi-RNN 表示双向递归神经网络。

从表 1 中的结果可以看出，Bi-TRNN 具有比 TRNN 更强的分类结果，从而说明双向 RNN 提升了模型对文本的学习能力。Bi-RNN 模型无论是使用 LSTM 节点还是 GRU 节点，其分类效果都好于大多数的基线模型。在新华社新闻数据集上，GRU Bi-RNN 获得了 88.3% 的 $F1$ 值; 在路透社的新闻数据集上，LSTM Bi-RNN 模型达到了 50.5% 的 $F1$ 值。可见，在文本学习的任务上，考虑单词的语义和词语之间的信息构成，对文本分类任务具有积极的意义。因此，模型性能的提升得益于两点: 双向模型的使用和 LSTM(或 GRU) 计算节点的引入。

同时也可以看到，在 RCV1-v2 数据集上，分类的效果仍然非常低，难以达到实际应用的水平。根据以往的研究结果，文本分类存在着很严重的长尾效应问题[4]。在一段文本中，能够反映文本核心主题词的往往是一些关键词，而大多数的词都是非关键词，在合成文本语义的过程中，这些非关键词也被学习到向量表示当中，对分类造成了一定的干扰，这是存在较大分类错误的主要原因，也是下一步工作的研究重点。

在 LSTM 和 GRU 的对比上，由于 LSTM 拥有比 GRU 更多的拟合参数，从而更加适合大数据的学习和预测，因此在路透社新闻语料上，LSTM 的预测效果明显好于 GRU，而在新华社新闻数据集上，GRU 的结果则好于 LSTM 节点。由此可以得出结论，对于训练样本较大的数据，更加适合采用 LSTM 节点。并且由于当前的矩阵计算方式，尤其在 GPU 环境下，LSTM 和 GRU 所需的计算时间差别不大。

3　结束语

本文基于文本分类的任务，提出了 LSTM 和 GRU 计算节点的双向递归神经网络算法，在提取文本的向量特征之后，通过 dropout 方式将特征输入到 softmax 分类器中。在 NLPCC 2014 评测数据集和 RCV1-v2 数据集上进行的实验结果证明双向递归神经网络取得了超过所有基线模型的效果，并且得出了 LSTM 节点更加适合大样本训练的结论。同时，为了分别验证 GRU、LSTM 和双向网络结构的学习能力，单独设计了 TRNN、Bi-TRNN、LSTM Bi-RNN 和 GRU Bi-RNN 的结构，它们在文本分类上的效果对比表明双向网络结构和 LSTM、GRU 对提高递归神经网络的学习能力均有帮助。

由于文本分类问题有很严重的长尾效应，在合成文本语义的过程中，这些长尾效应也被学习到向量表示当中，形成了噪声信息。本文下一步的工作就是如何从文本中选取能够代表文本类别的单词进行语义合成，而对其他干扰主题的单词进行过滤。未来的工作将集中研究关注机制在文本分类中的作用，以及减弱长尾效应对文本分类结果的影响。

参考文献:

[1] 姚全珠，宋志理，彭程. 基于 LDA 模型的文本分类研究[J]. 计算机工程与应用，2011，47(13): 150-153.
Yao Q Z，Song Z L，Peng C. Research on text categorization based on LDA[J]. Computer Engineering and Applications，2011，47(13): 150-153. (in Chinese)

[2] 张爱丽，刘广利，刘长宇. 基于 SVM 的多类文本分类研究[J]. 情报杂志，2004(9): 6-10.
Zhang A L，Liu G L，Liu C Y. Research on multiple classes text categorization based on SVM[J]. Journal of Information，2004(9): 6-10. (in Chinese)

[3] 刘华. 基于关键短语的文本分类研究[J]. 中文信息学报，2007，21(4): 34-41.
Liu H. Text categorization based on key phrases[J]. Journal of Chinese Infromation Processing，2007，21(4): 34-41. (in Chinese)

[4] Srivastava N，Salakhutdinov R，Hinton G E，et al. Modeling documents with a deep Boltzmann machine[C]//Uncertainty in Artificial Intelligence. Washington D. C.，2013.

[5] Graves A，Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures[J]. Neural Networks，2005，18(5): 602-610.

[6] Chung J，Gulcehre C，Cho K，et al. Gatedfeedback recurrent neural networks[C]//International Conference on Machine Learning. New York，USA，2015.

[7] Socher R，Huval B，Bhat B，et al. Convolutional-recursive deep learning for 3D object classification[C]//Advances in Neural Information Processing Systems. Carson City，USA，2012: 665-673.

[8] 蒋卓人，陈燕，高良才，等. 一种结合有监督学习的动态主题模型[J]. 北京大学学报: 自然科学版，2015，51(2): 367-376.
Jiang Z R，Chen Y，Gao L C，et al. A supervised dynamic topic model [J]. Acta Scientiarum Naturalium Universitatis Pekinensis，2015，51(2): 367-376. (in Chinese)

[9] Srivastava N，Hinton G，Krizhevsky A，et al. Dropout: a simple way to prevent neural networks from overfitting [J]. Journal of Machine Learning Research，2014，15(1): 1929-1958.

[10] Pennington J，Socher R，Manning C D. GloVe: global vectors for word representation [C] // Conference on Empirical Methods in Natural Language Processing. Doha，Qatar，2014: 1532-1543.

[11] Huang E H，Socher R，Manning C D，et al. Improving word representations via global context and multiple word prototypes [C] // Association for Computational Linguistics. Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. Jeju Island，South Korea，2012: 873-882.

[12] Mikolov T，Sutskever I，Chen K，et al. Distributed representations of words and phrases and their compositionality [C] // Advances in Neural Information Processing Systems. South Lake Tahoe，USA，2013: 3111-3119.

[13] Mikolov T，Yih W T，Zweig G. Linguistic regularities in continuous space word representations [C] // Proceedings of NAACL-HLT. Atlanta，USA，2013: 746-751.

[14] Luong M T，Socher R，Manning C D. Better word representations with recursive neural networks for morphology [C] // Proceedings of the Seventeenth Conferences on Computational Natural Language Learning. Sofia，Bulgaria，2013: 104-113.

[15] Socher R，Perelygin A，Wu J Y，et al. Recursive deep models for semantic compositionality over a sentiment treebank [C] // Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). Seattle，USA，2013: 1631-1642.

[16] Socher R，Huval B，Manning C D，et al. Semantic compositionality through recursive matrix-vector spaces [C] // Association for Computational Linguistics. Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Jeju Island，South Korea，2012: 1201-1211.

[17] Zeng D J，Liu K，Lai S W，et al. Relation classification via convolutional deep neural network [C] // Proceedings of COLING 2014，the 25th International Conference on Computational Linguistics: Technical Papers. Dublin，Ireland，2014: 2335-2344.

Application of recurrent neural networks in text classification

HUANG Lei DU ChangShun
(School of Economics and Management，Beijing Jiaotong University，Beijing 100044，China)

Abstract: Text classification is one of the important tasks in machine learning. It requires that a computer can classify texts automatically given a classification model. This task can help human to manage text and mine useful information. The growth of text data on the internet both requires the design of proper algorithms to extract key features and classify the texts，and that the algorithms can be used on a computer. The traditional methods regard words as symbols and do not consider their combinations. This article use LSTM and GRU bidirectional recurrent neural Networks to extract text features，and uses softmax to classify them. The model considers the meaning of words and grammatical structure，and it preserves the combination semantics among the words in a text. Therefore，the new proposed method can overcome the shortcomings of traditional models. We conducted experiments on two news classification datasets published by NLPCC2014 and Reuters. The proposed model achieves *F*-value of 88.3% and 50.5%，respestively，with the two datasets. The experimental results show that our method outperforms all the traditional baseline systems. In addition，our model does not need any human input and can be used with a wide range of texts.

Key words: text classification; deep learning; long short-term memory; gated recurrent unit; bidirectional recurrent neural network; word vector

(责任编辑: 吴万玲)