

# 基于极性转移和 LSTM 递归网络的情感分析\*

梁军, 柴玉梅, 原慧斌, 高明磊, 咎红英

郑州大学 信息工程学院, 河南 郑州 450001

**摘要:** 长短期记忆(long short term memory, LSTM)是一种有效的链式循环神经网络(recurrent neural network,  $R^2NN$ ), 被广泛用于语言模型、机器翻译、语音识别等领域。但由于该网络结构还是一种链式结构, 不能有效表征语言的结构层次信息, 本文将 LSTM 扩展到基于树结构的递归神经网络(recursive neural network, RNN)上, 用于捕获文本更深层次的语义语法信息, 并根据句子前后词语间的关联性引入情感极性转移模型。实验证明本文提出的模型优于 LSTM、递归神经网络等。

**关键词:** LSTM, 递归神经网络, 情感分析

**中图分类号:** TP391

**文献标识码:** A

## LSTM with Recursive Networks Based on Polarity Shifting for Sentiment Analysis

LIANG Jun, CHAI Yumei, Yuan Huibin, Gao Minglei, ZAN Hongying

School of Information Engineering, Zhengzhou University, Zhengzhou, Henan 450001, China

**Abstract:** The chain-structured long short-term memory (LSTM) has showed to be effective in a wide range of problems such as language model, machine translation and speech recognition. Because of the network is only a line chain, it cannot storage the structure of hierarchical information language. We extend it based tree structure to recursive neural network to capture more syntactic and semantic information, and sentiment polarity shifting model is introduced. Compared to LSTM, RNN etc, the proposed model achieves a state-of-the-art performance.

**Key words:** LSTM, Recursive neural network, Sentiment analysis

### 1 引言

近年来随着学者对深度学习的不断探索, 其在图像、语音识别领域都取得了重大进展。与此同时在 NLP 领域众多学者也在进行各种深入研究, 尤其在利用神经网络进行语言模型建立、文本特征学习和短文本情感分析方向都有突破。

在利用神经网络进行文本特征学习时需要首先要面对的问题就是对文本数据建模, 通常有三种模型: 词袋模型(bag-of-words model)、序列化模型(sequence model)和结构化模型(structure model)。这三种模型各有优劣, 对于词袋模型, 文本被看作是无序的词汇集合, 忽略语法甚至是单词的顺序, 该模型被广泛应用于传统贝叶斯模型、LSA、LDA 等文本分类模型中, 被证明是简单有效的, 但同时该模型的缺点也是十分明显的, 它忽略了文本本身的语法语义信息所以很难捕获到文本中深层的语义信息。而序列化模型则将文本看作是有序的词语序列, 这种模型考虑了文本的有序性以及词语间的关联性可以学习到一些词袋模型无法学习到的语义信息, 如 Bengio 等人提出的神经网络语言模型<sup>[1]</sup>就考虑了词语间的顺序性, 但为了降低模型的复杂度仅仅考虑窗口内的词语。Mikolov 等人<sup>[2]</sup>在 2010 年提出利用循环神经网络来构建语言模型, 这使得网络模型可以充分地利用上下文信息从而更好地进行语言建模。不过这些模型依然是将文本看作有序列的词语组合, 仅考虑了文本的有序性信息仍然忽略了文本在语义上的结构性。结构化模型则试图将文本看作有结构的词语组合, 使得在学习文本特征时能够充分保存文本的结构特征, 例如 Socher 等人<sup>[3]</sup>利用 Recursive

\*基金项目: 本文承国家自然科学基金项目(14BYY096)、国家自然科学基金项目(61402419, 61272221)、国家高技术研究发展 863 计划(2012AA011101)、计算语言学教育部重点实验室(北京大学)开放课题(201401)、国家重点基础研究发展计划 973 课题(2014CB340504)、河南省高等学校重点科研项目(15A520098)支持资助。

<sup>1</sup> 为了与递归神经网络(Recursive Neural Network)区别, 本文称之为  $R^2NN$ 。

Autoencoder 方法试图以一种树形结构来描述文本, 希望能够学习到文本的句法信息, 该方法在解决短文本情感分析的问题上取得了不错的效果。

在解决序列化问题时  $R^2NN$  被证明是一种有效的方法, 可以充分的利用上下文信息, 但该方法存在着梯度爆炸和消失的问题<sup>[4]</sup>, Hochreiter 和 Schmidhuber 等人<sup>[5]</sup>提出的 LSTM 型  $R^2NN$  有效的解决了这一问题。然而 LSTM- $R^2NN$  也是一种链式网络并不能利用文本的结构化信息, 因此本文将 LSTM 这一网络结构拓展到基于树形结构的 RNN 网络上构成 LSTM-RNN 模型, 并结合情感分析中的情感极性转移现象 (Polarity Shifting) 提出 PLSTM-RNN 模型从而更好地利用文本上下文信息、结构化信息和情感语义信息来解决文本情感分析这一问题。

本文具体章节安排如下: 第 2 节介绍了与本文相关的一些工作进展; 第 3 节介绍了融合 LSTM 和情感极性转移模型之后的网络模型; 第 4 节介绍将融合之后的模型应用于情感分析; 第 5 节为实验及分析; 最后对本文做出总结, 并指出后续的工作方向。

## 2 相关工作

### 2.1 递归神经网络

Socher 等人在 2011 年将递归神经网络成功应用在句法解析上<sup>[6]</sup>, 之后人们对它进行了许多研究。随后 Socher 在 RNN 的基础上添加一个感知层从而提高模型的表达能力<sup>[7]</sup>; Irsoy 和 Cardie<sup>[8]</sup>将 RNN 扩展为 3-layer RNN 从而成为一个深层网络结构。除了对模型的结构上进行优化, RNN 也被应用在 NLP 的其他领域, Li P 和 Liu Y<sup>[9]</sup>等人将 RNN 应用到机器翻译领域; Le 和 Zuidema<sup>[10]</sup>将其应用在依存句法分析上, 都取得了不错的效果。这些方法通过递归神经网络为语言建模, 可以有效的捕获到文本的语义信息, 从而提高了这些传统任务的性能。

### 2.2 情感分析

情感分析是一个新兴的研究课题, 具有很大的研究价值和应用价值<sup>[11]</sup>。鉴于此, 该研究课题受到国内外越来越多的研究机构的重视。Pang 和 Lee 等人在 2008 年就利用词袋模型对文本进行情感分析, 随后许多人尝试设计更好的工程特征或者使用基于句法结构的极性转移规则来提升情感分析的准确率。这些模型都是基于词袋模型, 无法获取到文本中的深层语义信息, 因此效果并不理想。2013 年 Socher 等人利用递归神经网络模型对影评数据进行情感分析取得了比传统方法更好地效果。梁军等人<sup>[12]</sup>通过向 Recursive Autoencoder 模型中融入情感极性转移模型在中文微博情感分析上取得不错效果。这两种模型都是树形结构网络, 但没有充分利用句子的历史信息。

## 3 基于极性转移和 LSTM 的递归网络模型

神经网络模型可以学习到不同长度、句法结构的短语的特征表示。这些特征表示可以作为分类器的特征输入为短语、句子进行分类。在利用神经网络模型进行句子特征表征学习时, 需要先将句子中的词语映射为  $d$  维的实数向量。这些词语对应的  $d$  维实数向量可以使用高斯分布  $U(-r, r)$  进行初始化, 也可以使用一些无监督方法进行预训练, 例如 google 推出的将词语转化为词向量的工具 word2vec, Socher 等人提出的 Recursive Autoencoder 方法。所有词语对应的词向量将会存储在一个词向量矩阵  $L \in \mathbb{R}^{d \times |V|}$  中, 其中  $|V|$  是词表的大小。词向量矩阵  $L$  同样会作为神经网络模型的参数被训练, 当模型训练完成后词向量矩阵  $L$  就会存储每个词对应的特征表示。在有了词语对应的词向量之后需要一定的方式来计算词语组合形成的词组、句子所对应的特征表示, 有了词组、句子对应的特征表示之后, 将这些特征作为 softmax 分类器的输入就可以得到它们的分类结果。

本节将介绍两个学习词语、句子特征表示的网络结构模型 RNN 和  $R^2NN$  模型, 并结合

两个模型的特点提出一个新的 LSTM-RNN 模型。该模型可以有效利用句子的语法结构信息和语义的历史信息进行词语、短语和句子的特征表征学习。通过该模型学习到的特征已经可以有效地被用来进行文本情感分析,但语言中会广泛出现情感极性转移的现象,本文进一步结合该现象对模型进行改进提出 PLSTM-RNN 模型。

### 3.1 RNN

递归神经网络最早由 Goller<sup>[14]</sup>提出,它根据句子语法的树形结构由底向上递归地合并两个相邻的节点,如图 1 所示。

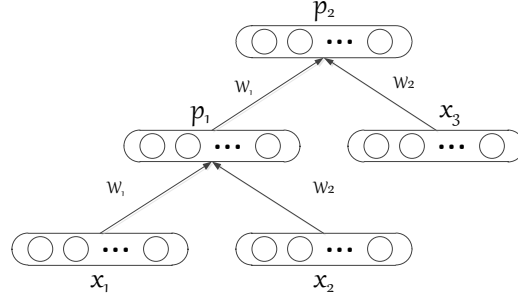


图 1 RNN 的网络结构

递归神经网络可以根据一个给定的有向无环图从下往上递归的应用相同的权重组合去构建网络结构。通过遍历该有向无环图拓扑结构中的节点,可以不断递归的由子节点的向量表示学习到父节点的向量表示。一般来说有向无环图的结构是没有限定的,但为了结构的统一性和简洁性通常将该有向无环图转换为二叉树的形式。

假定文本为  $x_{i-1}x_ix_{i+1}$ , 其句法解析树对应的二叉树结构为  $(p_{i+1}(p_i(x_{i-1}x_i)x_{i-1}))$ ,  $x_{i-1}, x_i, x_{i+1} \in \mathbb{R}^d$  表示词语,  $p_i \in \mathbb{R}^d$  表示短语  $x_{i-1}x_i$ ,  $p_{i+1} \in \mathbb{R}^d$  表示  $x_{i-1}x_ix_{i+1}$ ;  $W_1, W_2 \in \mathbb{R}^{d \times d}$  为系数矩阵, 则可以表述为:

$$p_i = g\left(W \begin{bmatrix} x_{i-1} \\ x_i \end{bmatrix} + b\right), p_{i+1} = g\left(W \begin{bmatrix} p_i \\ x_{i+1} \end{bmatrix} + b\right) \quad (1)$$

其中  $g$  为激活函数  $\tanh$ 。如图 1 所示递归神经网络通过不断合并两个短语的特征表示来学习更大跨度的短语的特征表示,在特征学习的过程中它根据语法结构关系先后合并句子中的词语,既保留了句子的词序特征又结合了句子的语法层次结构特征。

### 3.2 LSTM 型 R<sup>2</sup>NN

循环神经网络与递归神经网络的区别在于组成这些网络的神经元相互关联的结构不同:递归神经网络中节点的结合是按照有向无环图进行的,而循环神经网络中节点是按照出现的时序关系结合。事实上,可以把循环神经网络看作是一个具有特定结构的递归神经网络。如图 2 所示:

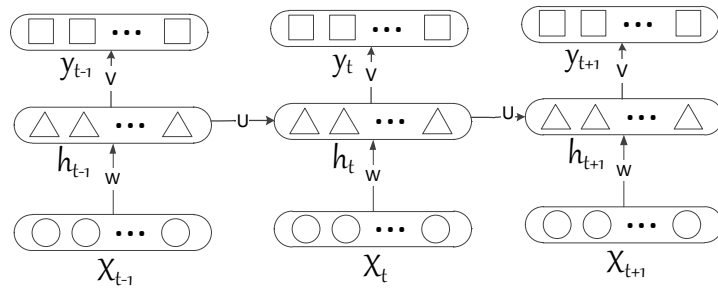


图 2 循环神经网络

在第  $t$  次网络传播时假定输入为  $x_t \in \mathbb{R}^V$ ,  $t-1$  次传播时隐含层为  $h_{t-1} \in \mathbb{R}^D$ , 则  $t$  次传播时隐含层计算方式如下:

$$h_t = \delta(W_x x_t + W_h h_{t-1})$$

其中  $W_x \in \mathbb{R}^{D \times V}$  是第  $t$  次输入的词向量  $x_t$  对应的系数矩阵,  $W_h \in \mathbb{R}^{D \times D}$  是  $t-1$  次传播输出  $h_{t-1}$  对应的系统矩阵,  $\delta$  是非线性激活函数 (sigmoid 函数或者 tanh 函数)。这样隐含层在每次计算时都会把上次输出作为本次输入从而达到保存历史信息的目的。但是在使用 BP、BPTT 算法优化网络时会遭遇梯度爆炸或者梯度衰减的问题<sup>[4]</sup>, 而 LSTM 使用一个记忆单元 (memory cell) 来替代  $R^2NN$  中的隐含层从而可以避免梯度在反向传播中遇到的爆炸和衰减问题。LSTM 的网络结构如图 3 所示:

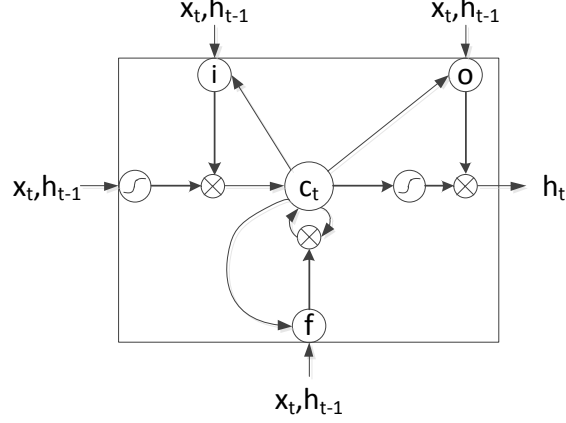


图 3 LSTM 网络结构

LSTM 型  $R^2NN$  由 input gate  $i$ 、output gate  $o$  和 forget gate  $f$  和 memory cell  $c$  组成, 其中 input gate、output gate 和 forget gate 是控制 memory cell 的读、写和丢失操作的控制器。假设 memory cell 的维度为  $d$ , 那么 LSTM 可以表述为:

$$\begin{aligned} i_t &= \delta(W_i x_t + U_i h_{t-1} + V_i c_{t-1} + b_i) \\ f_t &= \delta(W_f x_t + U_f h_{t-1} + V_f c_{t-1} + b_f) \\ c_t &= f_t \odot c_{t-1} + i_t \odot \tanh(W_c x_t + U_c h_{t-1} + b_c) \\ o_t &= \delta(W_o x_t + U_o h_{t-1} + V_o c_t + b_o) \\ h_t &= o_t \odot \tanh(c_t) \end{aligned} \quad (2)$$

其中  $\delta$  是激活函数 sigmoid;  $\odot$  是点乘运算;  $W_*, U_*, V_*, b_*$  分别表示系数矩阵和偏置向量,  $i_t, f_t, o_t$  分别表示  $t$  时刻 input gate、forget gate 和 output gate 的计算方法,  $c_t$  表示  $t$  时刻 memory cell 的计算方法,  $h_t$  为  $t$  时刻 LSTM 单元的输出。由图 2 可以看出 3 个控制门 Input gate, Output gate 和 Forget gate 的输出分别连接到 1 个乘法单元上, 从而分别控制网络的输入、输出以及 cell 单元的状态。

### 3.3 基于 LSTM 的 RNN

本小节介绍如何将 LSTM 扩展到 RNN 模型上, 使得既可以获得文本的机构信息又可以利用 LSTM 对历史信息的存储。这样在计算父节点  $p$  的向量表示时不仅仅可以用到两个子节点的结构信息还可以利用历史信息进行计算。

与 LSTM 单元类似, LSTM-RNN 的记忆单元也同样由 input gate, forget gate, output gate 和 memory cell 组成, 但由于 LSTM-RNN 是基于树结构, 所以在  $t$  时刻 LSTM-RNN 的输入不止一个 (对于二叉树来说, 每次输入有两个), 所以 LSTM-RNN 对于每个 LSTM 单元都有 2 个 input gate 和 forget gate, 如图 4 所示, 一个基于二叉树结构的 LSTM-RNN 单元是有 2 个 input gate, 1 个 output gate, 1 个 memory cell 和 2 个 forget cell 组成。

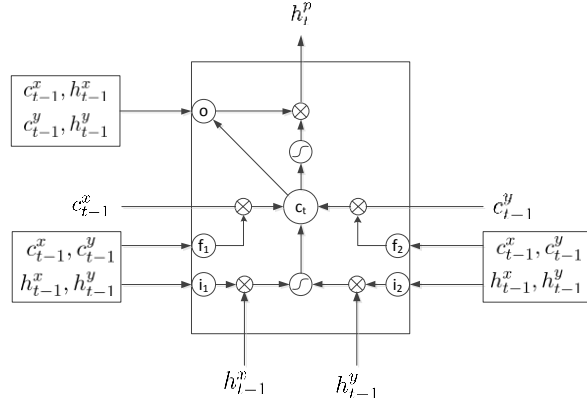


图 4 LSTM-RNN 网络结构

假定两个子节点分别为 $x, y$ ，父节点为 $p$ ， $h_{t-1}^x, h_{t-1}^y$ 分别为孩子节点 $x, y$ 在隐藏层的向量表示， $c_{t-1}^x, c_{t-1}^y$ 分别表示子节点 $x, y$ 的记忆向量（cell vector）， $W_*, U_*$ 为系数矩阵， $b_*$ 为偏置项，那么 LSTM-RNN 可以描述为：

$$\begin{aligned}
 i_1 &= \delta(W_{i1}^x h_{t-1}^x + W_{i1}^y h_{t-1}^y + U_{i1}^x c_{t-1}^x + U_{i1}^y c_{t-1}^y + b_{i1}) \\
 i_2 &= \delta(W_{i2}^x h_{t-1}^x + W_{i2}^y h_{t-1}^y + U_{i2}^x c_{t-1}^x + U_{i2}^y c_{t-1}^y + b_{i2}) \\
 f_1 &= \delta(W_{f1}^x h_{t-1}^x + W_{f1}^y h_{t-1}^y + U_{f1}^x c_{t-1}^x + U_{f1}^y c_{t-1}^y + b_{f1}) \\
 f_2 &= \delta(W_{f2}^x h_{t-1}^x + W_{f2}^y h_{t-1}^y + U_{f2}^x c_{t-1}^x + U_{f2}^y c_{t-1}^y + b_{f2}) \\
 c_t &= f_1 \odot c_{t-1}^x + f_2 \odot c_{t-1}^y + \tanh(W_c^x h_{t-1}^x \odot i_1 + W_c^y h_{t-1}^y \odot i_2 + b_c) \\
 o &= \delta(W_o^x h_{t-1}^x + W_o^y h_{t-1}^y + U_o^x c_{t-1}^x + U_o^y c_{t-1}^y + U_o^c c_t + b_o) \\
 h_t^p &= o \odot \tanh(c_t)
 \end{aligned} \tag{3}$$

其中  $i_1, i_2$  分别表示两个孩子节点对应的 input gate 的计算方法， $f_1, f_2$  分别表示两个孩子节点对应的 forget gate 的计算方法， $o$  表示父节点对应的 output gate 的计算方法， $c_t$  则表示父节点对应的 memory cell 的计算方法， $h_t^p$  则是父节点对应的输出。

由于每个孩子节点都对应一个 input gate，通过训练可以使得对父节点贡献大的子节点对应的 input gate 更接近 1，对父节点贡献小的子节点对应的 input gate 则更接近 0；同样的 forget gate 控制在计算父节点时利用哪个子节点的历史信息。

### 3.4 融合极性转移的 LSTM-RNN

在各个语言中，极性转移对于文本情感分析都是一个极大的挑战。句子中的一些否定词、程度副词等的使用都可能会使得句子的极性发生偏转。张小倩<sup>[13]</sup>等人对极性转移现象进行研究分析分为三类，本文根据其研究进行如下分类：

表 1 极性转移实例

类型	样例	线索	作用域
强调	他 <b>很喜欢</b> 这本书	显式	句内
否定	我 <b>不喜欢</b> 这个地方	显式	句内
转折	这个手机 <b>还可以</b> ， <b>但是</b> 我不喜欢	显式	句内
句间极性转折	这个手机 <b>款式不错</b> ，但是 <b>配置不行</b>	隐式	句间

由上表可以看出，情感极性转移现象大致可以分为三类：强调、否定和转折。但其实可以看出，这种情感极性转移现象是和前后词语本身的情感极性密切相关的。本文针对这种语言现象对 LSTM-RNN 模型进行进一步的优化，在计算父节点的向量表示时使其不仅与两个子节点的语义信息相关，同时也与两个子节点的情感信息关联，从而得到基于极性转移模型的 PLSTM-RNN。假定文本的情感标签集合为  $A=\{\text{Negative}, \text{Positive}\}$ ，其中有  $|A|$  个标签，假如当前节点的左子节点的情感标签为  $A_m$ ，其右子节点的情感标签为  $A_n$ ，则定义其极性转移向量为  $b_*^{mn}$ 。即当前节点的向量表示由其两个子节点的语义信息和情感信息共同决定。

$$\begin{aligned}
i_1 &= \delta(W_{i1}^x h_{t-1}^x + W_{i1}^y h_{t-1}^y + U_{i1}^x c_{t-1}^x + U_{i1}^y c_{t-1}^y + b_{i1}^{mn}) \\
i_2 &= \delta(W_{i2}^x h_{t-1}^x + W_{i2}^y h_{t-1}^y + U_{i2}^x c_{t-1}^x + U_{i2}^y c_{t-1}^y + b_{i2}^{mn}) \\
f_1 &= \delta(W_{f1}^x h_{t-1}^x + W_{f1}^y h_{t-1}^y + U_{f1}^x c_{t-1}^x + U_{f1}^y c_{t-1}^y + b_{f1}^{mn}) \\
f_2 &= \delta(W_{f2}^x h_{t-1}^x + W_{f2}^y h_{t-1}^y + U_{f2}^x c_{t-1}^x + U_{f2}^y c_{t-1}^y + b_{f2}^{mn}) \\
c_t &= f_1 \odot c_{t-1}^x + f_2 \odot c_{t-1}^y + \tanh(W_c^x h_{t-1}^x \odot i_1 + W_c^y h_{t-1}^y \odot i_2 + b_c^{mn}) \\
o &= \delta(W_o^x h_{t-1}^x + W_o^y h_{t-1}^y + U_o^x c_{t-1}^x + U_o^y c_{t-1}^y + U_o^c c_t + b_o^{mn}) \\
h_t^p &= o \odot \tanh(c_t)
\end{aligned} \tag{4}$$

这样针对子节点情感极性的不同组合就可以由其对应的 $b_*^{mn}$ 感知。通过反馈训练网络就可以使得 $b_*^{mn}$ 捕获到对应的情感极性组合的极性转移模型。

#### 4 PLSTM-RNN 应用于情感分析

本节介绍如何将扩展的 PLSTM-RNN 模型应用于情感分析，具体网络结构图如图 5 所示。

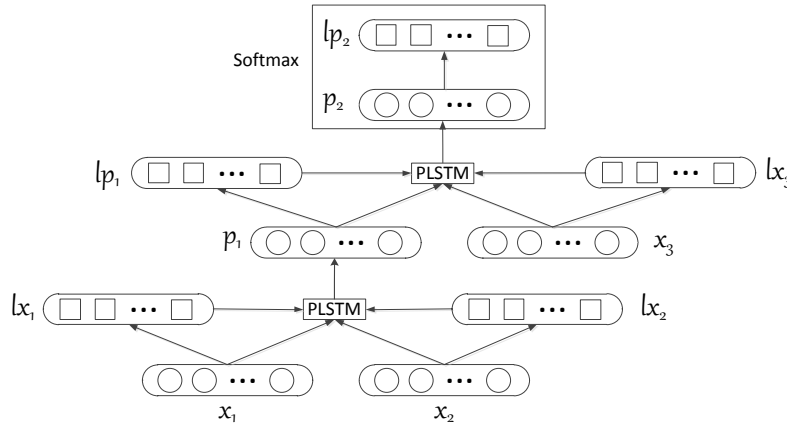


图 5 PLSTM-RNN

其中  $l_x$  是节点  $x$  通过 PLSTM-RNN 模型训练得到特征向量表示后作为 softmax 回归的输入得到的情感极性标签，同时  $l_x$  作为 PLSTM 的输入决定偏置向量的选取；假定  $y$  是节点  $x$  对应的真正的情感。那么节点  $x$  的情感极性为  $A_k$  的概率计算方法如下：

$$pr(A_k|x, \theta) = \frac{e^{W_k^l x + b_k^l}}{\sum_{i=1}^{|A|} e^{(W_i^l x + b_i^l)}} \tag{5}$$

假定训练样本为  $T$ ，那么在训练是定义损失函数为：

$$J(\theta) = -\frac{1}{|T|} \sum_{s \in T} \sum_{n \in s} 1\{A_i = j\} \log pr(A_i|n, \theta) + \frac{\lambda}{2} \|\theta\|^2 \tag{6}$$

其中  $s$  为样本集  $T$  中的句子， $n$  为每个句子中的节点， $1\{\cdot\}$  是示性函数，其取值规则为： $1\{\text{Ture}\} = 1$ 。在训练模型的时候可以使用批梯度下降<sup>[14]</sup>的方法来最小化损失函数 $J(\theta)$ ，当然也可以采用 LBFGS 或 AdaGrad<sup>[15][14]</sup>等方法来优化参数。模型的算法描述如下所示：

---

##### PLSTM-RNN

---

输入：训练语料  $T$  及其对应标签：1

输出： $\theta$

- 1) 初始化词向量及 $\theta$ ;
-

## 2) while 不收敛 do

$\nabla J = 0$

for all  $\langle s, t \rangle \in T$  do

while  $p$  isn't Root do

$p = \text{PLSTM\_RNN}(x_l, x_r)$

end while

计算  $\nabla J_i = \partial J(s, t) / \partial \theta$

更新  $\nabla J = \nabla J + \nabla J_i$

end for

更新  $\Theta = \frac{1}{N} \nabla J + \lambda \theta$

end while

## 5 实验

### 5.1 数据集

为了验证模型的有效性, 本文选用斯坦福大学在影评数据上开发的 Stanford Sentiment Treebank<sup>[7]</sup>, 该数据集的标签分为五类(very negative, negative, neutral, positive, very positive), 共有 11,855 个句子, 平均每个句子包含 19 个词语, 共有 215,154 个短语, 按照原有的数据集分割方案: 训练集 8544 句, 开发集 1101 句, 测试集 2210 句。另外, 将其中没有极性的句子去掉可以作为二分类的验证数据集, 没有极性的句子约占整个数据集的 20% 左右, 去掉之后的开发集 6920 句, 开发集 872 句, 测试集 1821 句。

### 5.2 向量维度的选择

在进行实验时, 词向量使用 Jeffrey Pennington et al<sup>[16]</sup>使用 Common Crawl 数据训练得到的 300 维向量<sup>2</sup>初始化, 中间节点则使用不同维度的向量进行表征。因为随着内部节点维度的增加整个模型的复杂度成倍正常, 因此需要为内部节点的向量表征选择一个合适的维度, 下图是开发集上调节内部节点不同的向量维度 PLSTM-RNN 模型的评估:

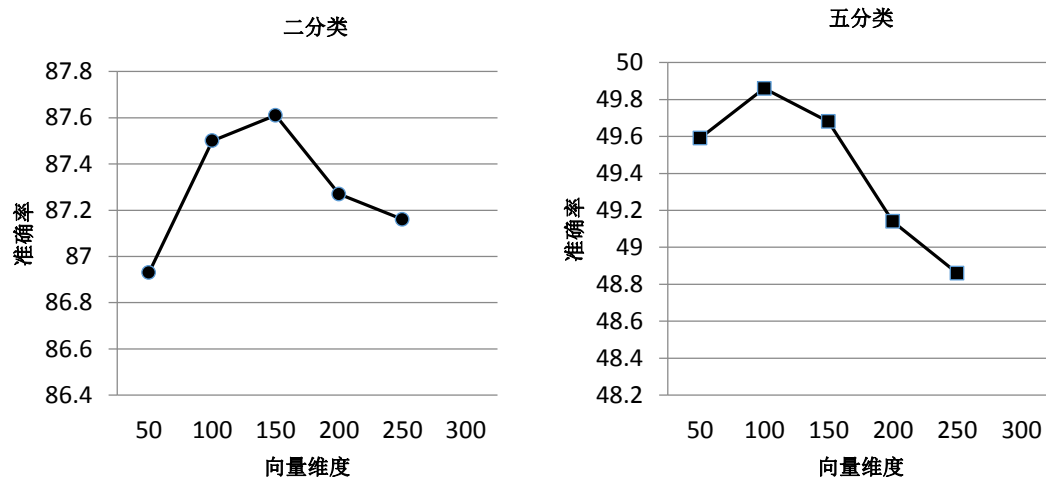


图6 内部节点向量维度的影响

由上图可以看出, 内部节点向量维度在 100 维时即可达到较理想的效果。

<sup>2</sup> <http://nlp.stanford.edu/projects/glove/>

### 5.3 模型对比实验

本节将该文提出的模型 PLSTM-RNN 和之前在该数据集上试验过的模型进行对比（使用准确率作为评估标准），主要有 Naive Bayes with bag of bigram features (BiNB)，Recursive neural tensor network (RNTN)<sup>[7]</sup>，Convolutional neural network (CNN)<sup>[17]</sup>，Dynamic convolutional neural network (DCNN)<sup>[18]</sup>和 Deep RNN(DRNN)<sup>[8]</sup>模型。

表 2 模型对比结果

模型	2-class 分类	5-class 分类
BiNB	83.1	41.9
RNTN	85.4	45.7
CNN	<b>88.1</b>	48.0
DCNN	86.8	48.5
DRNN	86.6	49.8
LSTM	86.7	45.8
LSTM-RNN	87.7	50.0
PLSTM-RNN	<b>88.1</b>	<b>50.3</b>

通过上表可以看出，在对文本进行二分类或者五分类时 PLSTM-RNN 模型都能够达到很好的效果，证明该模型是有效的。从直观上来看，利用 LSTM 可以有效保存文本的历史信息，这对文本进行情感分析是非常有帮助的；同时结合 RNN 网络的特点可以充分利用句子本身的结构信息，使得模型学习到的特征表示更完备；在此基础上进一步加入节点的联合情感信息，最终结合了文本的语义信息、结构信息和情感信息从而使模型可以达到最好的效果。

## 6 结语

本文将基于链式结构的 LSTM 型网络扩展到 RNN 网络，使得模型可以在更好地获取文本历史信息的同时，挖掘到文本的结构信息，丰富特征学习，同时针对情感极性转移这一问题提出情感极性转移模型，并将其融入到 LSTM-RNN 中，进一步优化了模型的表征能力。

基于文本结构的神经网络模型在文本处理上有较强的优势，但对于如何快速有效学习到文本的结构特征仍然是一个需要亟待解决的问题，下一步工作也将在这方面进行探索，从而希望更好地对文本进行结构化表征。

## 参考文献：

- [1] Bengio Y, Ducharme R, Vincent P, et al. A neural probabilistic language model[J]. The Journal of Machine Learning Research, 2003, 3: 1137-1155.
- [2] Mikolov T, Karafiát M, Burget L, et al. Recurrent neural network based language model[C]//INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010. 2010: 1045-1048.
- [3] Socher R, Pennington J, Huang E H, et al. Semi-supervised recursive autoencoders for predicting sentiment distributions[C]//Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011: 151-161.
- [4] Hochreiter S, Bengio Y, Frasconi P, et al. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies[J]. 2001.
- [5] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.
- [6] Socher R, Lin C C, Manning C, et al. Parsing natural scenes and natural language with recursive neural networks[C]//Proceedings of the 28th international conference on machine learning (ICML-11). 2011:



- [7] 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). 2013, 1631: 1642.
- [8] Irsoy O, Cardie C. Deep Recursive Neural Networks for Compositionality in Language[C]//Advances in Neural Information Processing Systems. 2014: 2096-2104.
- [9] Li P, Liu Y, Sun M. Recursive Autoencoders for ITG-Based Translation[C]//EMNLP. 2013: 567-577.
- [10] Le P, Zuidema W. Inside-Outside Semantics: A Framework for Neural Models of Semantic Composition[J].
- [11] 赵妍妍, 秦兵, 刘挺. 文本情感分析[J]. 软件学报, 2010, 21(8):1834-1848. DOI:10.3724/SP.J.1001.2010.03832.
- [12] 梁军, 柴玉梅, 原慧斌, et al. 基于深度学习的微博情感分析[J]. 中文信息学报, 2014, 28(5):155834-1
- [13] 张小倩. 情感极性转移现象研究及应用[D]. 苏州大学, 2012. DOI:10.7666/d.y2120830.
- [14] Goller C, Kuchler A. Learning task-dependent distributed representations by backpropagation through structure[C]//Neural Networks, 1996., IEEE International Conference on. IEEE, 1996, 1: 347-352.
- [15] Duchi J, Hazan E, Singer Y. Adaptive subgradient methods for online learning and stochastic optimization[J]. The Journal of Machine Learning Research, 2011, 12: 2121-2159.
- [16] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation[J]. Proceedings of the Empirical Methods in Natural Language Processing (EMNLP 2014), 2014, 12.
- [17] Kim Y. Convolutional neural networks for sentence classification[J]. arXiv preprint arXiv:1408.5882, 2014.
- [18] Blunsom P, Grefenstette E, Kalchbrenner N. A convolutional neural network for modelling sentences[C]//Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 2014.



梁军 (出生年 1990—), 通讯作者, 硕士研究生, 主要研究领域为机器学习、自然语言处理, Email: [zhengdaxg@163.com](mailto:zhengdaxg@163.com) ;



柴玉梅 (1964—), 教授, 主要研究领域为机器学习、数据挖掘和自然语言处理, Email: [ieymchai@zzu.edu.cn](mailto:ieymchai@zzu.edu.cn) ;



原慧斌 (1966—), 女, 学士, 研究员级高工, 主要研究领域为中文全文检索技术, 自然语言处理, Email: [1074821578@qq.com](mailto:1074821578@qq.com)。