

一种针对成分树的混合神经网络模型

霍欢^{1,2}, 薛瑶环¹, 黄君扬¹, 金轩城¹, 邹依婷¹

(1. 上海理工大学 光电信息与计算机工程学院, 上海 200093;

2. 复旦大学 上海市数据科学重点实验室, 上海 201203)

摘要: 为了提高自然语言处理的准确度, 很多工作将句法成分树与 LSTM 相结合, 提出了各种针对成分树的 LSTM 模型 (本文用 C-TreeLSTM 统称这类模型)。考虑到 C-TreeLSTM 模型在计算内部节点隐藏状态的过程中由于一个重要信息来源 (即单词) 的缺失导致文本建模的准确度不高, 为了改进这一缺点, 本文提出一种针对成分树的混合神经网络模型, 通过在 C-TreeLSTM 模型的节点编码过程中注入各节点所覆盖的短语语义向量来增强节点对文本语义的记忆, 故将此模型命名为 SC-TreeLSTM。实验结果表明, 在情感分类和机器阅读理解两类任务上 SC-TreeLSTM 的表现优异。

关键词: 成分树; C-TreeLSTM; 短语语义向量; 混合模型

中图分类号: TP391 文献标识码: A

A Hybrid Neural Network Model on Constituent Tree Structure

HUO Huan^{1,2}, XUE Yaohuan¹, HUANG Junyang¹, JIN Xuancheng¹, ZOU Yiting¹

(1.School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and technology, Shanghai 200093, P.R. China;

2.Shanghai Key Laboratory of Data Science, Fudan University, Shanghai 201203, P.R. China)

Abstract: In order to improve the accuracy of natural language processing, a lot of work combined constituent trees with LSTM and proposed various LSTM models for constituent trees (this paper uses C-TreeLSTM to collectively refer to this type of model). Since C-TreeLSTM has a low accuracy for text modeling due to the lack of the important information Sources (ie, words) in computing hidden state of internal nodes, this paper proposes a hybrid neural network model SC-TreeLSTM, which is based on the constituent tree structure. The model enhances node's memory of text semantics by injecting phrase semantic vectors which is covered by corresponding node during coding section in C-TreeLSTM. The experimental results show that the performance of SC-TreeLSTM is excellent in both sentiment classification and machine reading comprehension tasks.

Key words: Constituent tree; C-TreeLSTM; phrase semantic vector; hybrid model

1 引言

现有的文本处理模型大致可以分为三类: BOW (Bag-of-Words) 模型、基于序列结构的神经网络模型和基于树结构的神经网络模型。与 BOW 模型^[1]中词与词间的独立性假设相比, 基于序列结构的神经网络模型^[2,3]将词序信息纳入文本的表示学习, 并因其突出性能被广泛使用。但是这两种模型均未考虑文本自身存在的句法结构, 而句法结构对获取文本的语义特征非常重要。因此, 学者们将序列结构扩展为树结构, 将句法结

构特征纳入文本的表示学习中, 在各类自然语言处理任务上取得了很好的效果^[4-6]。现有的针对成分树 (Constituent Tree) 的 LSTM^[7]模型有三种: Zhu 等人提出的 S-LSTM^[8], Tai 等人提出的 Constituency Tree-LSTMs^[9]和 Le 等人提出的 LSTM-RNN^[10], 这类模型对序列 LSTM 的扩展方式基本相同, 都是将顺序处理的 LSTM 单元按成分树递归排布, 将句法结构纳入文本的表示学习中, 大大改善了文本表达的质量。本文将这类模型统称为 C-TreeLSTM。

但 C-TreeLSTM 模型的缺点也显而易见: 模型将文本句子解析成成分树, 其中叶子节点对应

收稿日期: 2018-06-10

定稿日期: 2018-07-25

基金项目: 国家自然科学基金 (61003031), 上海重点科技攻关项目 (14511107902), 上海市工程中心建设项目 (GCZX14014), 上海市一流学科建设项目 (XTKX2012), 上海市数据科学重点实验室开放课题资助课 (201609060003), 沪江基金研究基地专项 (C14001)。

作者简介: 霍欢 (出生年 1979), 女, 博士, 副教授, CCF 会员, 主要研究方向: 云计算、数据挖掘及不确定数据流技术、大数据、自然语言处理; 薛瑶环 (出生年 1993), 女, 硕士, 主要研究方向: 自然语言处理; 黄君扬 (出生年 1996), 男, 学士, 主要研究方向: 大数据; 金轩城 (出生年 1998), 男, 学士, 主要研究方向: 大数据; 邹依婷 (出生年 1994), 女, 硕士, 主要研究方向: 自然语言处理。

于输入单词, 内部节点表示句法成分。由于模型只在叶子节点直接使用输入单词的语义信息, 内部节点的隐藏状态仅由左右子节点的隐藏状态和记忆单元决定, 与输入词之间没有直接关联^[11]。这种计算方式在文本的长距离编码过程中对单词语义信息的记忆越来越少, 影响了文本建模的准确度, 尤其是对于序列较长的复杂任务, 文本建模的准确度不高。

为了解决这个问题, 本文提出一种针对成分树的混合神经网络模型。该模型借助 BiLSTM^[12]的编码方法计算文本成分树各内部节点所覆盖短语的语义向量, 然后将短语向量注入相应内部节点进行节点的编码, 增强节点对文本语义信息的记忆, 故将此模型命名为 Sementic C-TreeLSTM (SC-TreeLSTM)。模型包含短语向量计算模块和节点编码模块两个子模块。前者完成内部节点所覆盖的短语语义向量的计算, 后者进行节点成分的编码。本文将在情感分类和机器阅读理解两种自然语言处理任务上对模型进行测试, 实验结果表明 SC-TreeLSTM 的表现普遍优于 C-TreeLSTM。

本文的结构安排如下: 第 2 节对 C-TreeLSTM 进行概述, 第 3 节阐述了 SC-TreeLSTM 模型的详细设计, 第 4 节介绍实验设置并分析实验结果, 第 5 节介绍相关工作, 第 6 节总结全文。

2 背景知识

2.1 BiLSTM

BiLSTM 网络以 LSTM 网络为基础, 对每个训练序列按照向前和向后分别训练正向和反向两个 LSTM 网络, 则序列中的每个元素为正向 LSTM 的隐藏状态和反向 LSTM 的隐藏状态的拼接。LSTM 网络包括以下几个部分: 输入序列、输入门 i_j 、输出门 o_j 和遗忘门 f_j , 记忆单元 c_j 控制并调整长期的记忆单元记忆和遗忘哪些信息, 给定输入序列 $X: x_1, x_2, \dots, x_l$, 当前时刻的记忆单元 c_j 计算为:

$$c_j = f_j \otimes c_{j-1} + i_j \otimes \mathcal{C}_j^{\%} \quad (1)$$

其中,

$$\mathcal{C}_j^{\%} = \tanh(W_{x\mathcal{C}}x_j + W_{h\mathcal{C}}h_{j-1} + b_{\mathcal{C}}) \quad (2)$$

$$i_j = \sigma(W_{xi}x_j + W_{hi}h_{j-1} + W_{ci}c_{j-1} + b_i) \quad (3)$$

$$f_j = \sigma(W_{xf}x_j + W_{hf}h_{j-1} + W_{cf}c_{j-1} + b_f) \quad (4)$$

其中, j 为当前输入的索引, $1 \leq j \leq l$; \otimes 为 Hadamard 乘积操作; 输入门 i_j 控制新内容的输入, 遗忘门 f_j 控制旧内容的遗忘; σ 为对应元素相乘的函数, 如双曲正切函数或逻辑 S 形函数。最后, 根据当前时刻的记忆单元 c_j 和相应输出门 o_j 计算当前输入的隐藏状态 h_j :

$$o_j = \sigma(W_{xo}x_j + W_{ho}h_{j-1} + W_{co}c_j + b_o) \quad (5)$$

$$h_j = o_j \otimes \tanh(c_j) \quad (6)$$

其中, $W_{x\mathcal{C}}, W_{h\mathcal{C}}, b_{\mathcal{C}}, W_{xi}, W_{hi}, W_{ci}, b_i, W_{xf}, W_{hf}, W_{cf}, b_f, W_{xo}, W_{ho}, W_{co}$ 和 b_o 均为模型的参数。

对于输入序列 $X: x_1, x_2, \dots, x_l$, 其正向序列为 $\overset{\text{L}}{X}: x_1, x_2, \dots, x_l$, 反向序列为 $\overset{\text{R}}{X}: x_l, \dots, x_2, x_1$ 。单向 LSTM 如果仅接受正向序列 $\overset{\text{L}}{X}$, 则网络中的每个单元只包含该单元之前的文本信息, 同样, 如果仅接受反向序列, 则网络中的每个单元只包含该单元之后的文本信息。为了充分利用上下文信息, 构建 BiLSTM 网络, 序列 X 中的各元素表示为: $X = \left[\overset{\text{L}}{X}; \overset{\text{R}}{X} \right]$ 。

2.2 针对成分树建模的 C-TreeLSTM

本小节将对 C-TreeLSTM 模型进行概述。图 1 展示了将 John visited Mary this afternoon 进行解析后的成分树。

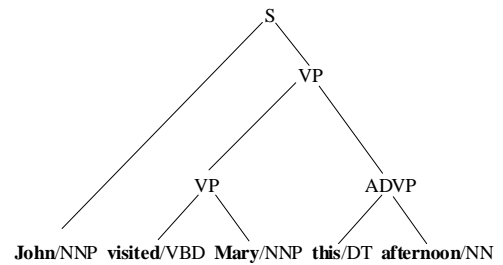


图 1 成分树示例

模型在句子的成分树上构建 LSTM 网络, 每个树节点代表 LSTM 的一个单元块。对内部节点来说, 模型将 LSTM 中前一时刻的隐藏状态 h_{t-1} 分解为左子节点的隐藏状态 h_{t-1}^L 和右子节点的隐藏状态 h_{t-1}^R , 同时将前一时刻的记忆单元 c_{t-1} 分解为左子节点的记忆单元 c_{t-1}^L 和右子节点的记忆单元 c_{t-1}^R , 如图 2 (a) 所示。当前节点的记忆单元 c_t 计算为:

$$c_t = f_t^L \otimes c_{t-1}^L + f_t^R \otimes c_{t-1}^R + i_t \otimes \mathcal{C}_t^{\%} \quad (7)$$

其中,

$$\mathcal{C}_t^{\%} = \tanh\left(\sum_{D \in \{L, R\}} W_{h\mathcal{C}}^D h_{t-1}^D + b_{\mathcal{C}}\right) \quad (8)$$

$$i_t = \sigma \left(\sum_{D \in \{L,R\}} (W_{hi}^D h_{t-1}^D + W_{ci}^D c_{t-1}^D) + b_i \right) \quad (9)$$

$$f_t^L = \sigma \left(\sum_{D \in \{L,R\}} (W_{hf_L}^D h_{t-1}^D + W_{cf_L}^D c_{t-1}^D) + b_{f_L} \right) \quad (10)$$

$$f_t^R = \sigma \left(\sum_{D \in \{L,R\}} (W_{hf_R}^D h_{t-1}^D + W_{cf_R}^D c_{t-1}^D) + b_{f_R} \right) \quad (11)$$

其中, t 为节点的索引, i_t 为输入门, 遗忘门 f_t 被分解为 f_t^L 和 f_t^R , 分别控制记忆单元 c_{t-1}^L 和 c_{t-1}^R 。

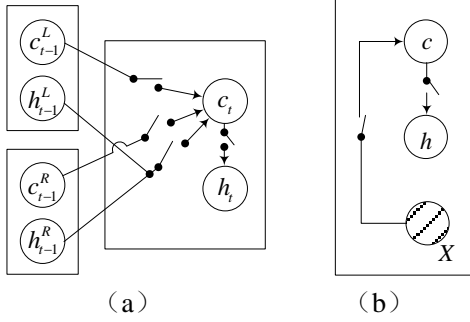


图2 节点隐藏状态计算

最后, 根据当前节点的记忆单元 c_t 和相应输出门 o_t 计算每个节点的隐藏状态 h_t :

$$o_t = \sigma \left(\sum_{D \in \{L,R\}} (W_{ho}^D h_{t-1}^D + W_{co}^D c_t) + b_o \right) \quad (12)$$

$$h_t = o_t \otimes \tanh(c_t) \quad (13)$$

其中, $W_{ci}^L, W_{ci}^R, W_{hf_L}^L, W_{hf_L}^R, W_{hf_R}^L, W_{hf_R}^R, b_{f_L}, b_{f_R}, W_{ho}^L, W_{ho}^R, W_{co}$ 和 b_o 均为模型的参数。

对于代表单个单词的叶子节点, 它们没有遗忘门, 输入是该单词的词嵌入向量, 如图 2 (b) 所示。根据公式 (7) - (13), 内部节点的隐藏状态 h_t 仅由左右子节点的隐藏状态 h_{t-1}^L, h_{t-1}^R 和记忆单元 c_{t-1}^L, c_{t-1}^R 决定, 与输入单词之间没有直接关联。这种计算方式在文本的长距离编码过程中对单词语义信息的记忆越来越少, 是导致文本建模准确度不高的根本原因。

3 混合神经网络模型 SC-TreeLSTM

本文提出一种针对成分树的 SC-TreeLSTM 模型, 该模型借助 BiLSTM 网络的单词编码方法计算各内部节点所覆盖短语的语义向量, 然后将短语向量注入相应内部节点进行树节点的编码, 增强节点对文本语义信息的记忆, 提高文本建模的准确度。

3.1 短语向量计算模块

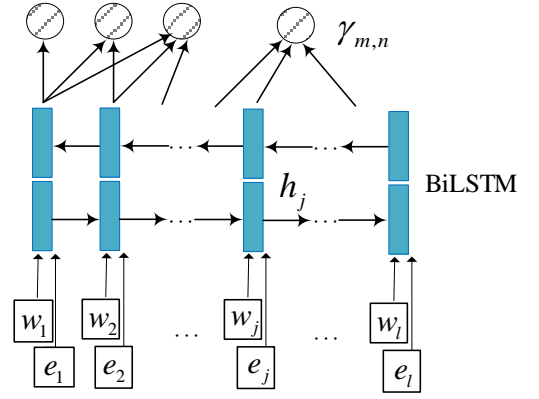


图3 短语语义向量计算过程

图 3 展示了短语语义向量的计算过程。对于文本句子 $X : x_1, x_2, \dots, x_l$, 首先, 将一系列单词输入 BiLSTM 网络进行编码, 获得包含单词本身词义和上下文信息的语义向量:

$$h_j^r = LSTM^r(h_{j-1}, [w_j, e_j]) \quad j=1, \dots, l \quad (14)$$

$$h_j^s = LSTM^s(h_{j+1}, [w_j, e_j]) \quad j=l, \dots, 1 \quad (15)$$

$$h_j = [h_j^r; h_j^s] \quad (16)$$

其中, w_j 是由 Glove^[13]或 Word2vec^[14]预先训练好的词嵌入向量, e_j 是由卷积神经网络 CNN^[15,16]获得的字符嵌入向量, 通过将单词中的每个字符输入到卷积神经网络来计算, 词嵌入和字符嵌入的连接即为文本的单词表示; $[\cdot; \cdot]$ 表示两个向量的连接 (concatenation) 操作, h_j 为 BiLSTM 网络的隐藏状态, 表示文本中第 j 个单词的语义向量, 由前向 LSTM 的隐藏状态 h_j^r 和后向 LSTM 的隐藏状态 h_j^s 连接得到。

然后, 计算文本的短语语义向量。对于句子中的任意短语 $V_{m,n}$, 其短语语义向量由以下公式计算:

$$\gamma_{m,n} = g(h_m, \dots, h_n) \quad (17)$$

其中, m, n 为输入序列的单词索引, $1 \leq m \leq n \leq l$, $g(g)$ 表示函数, $\gamma_{m,n}$ 为短语的语义向量。实验发现, 与 max, average, difference^[17] 等方法相比, 最佳函数是连接短语内前向 LSTM 第一个单词的隐藏状态 h_m^r 和反向 LSTM 最后一个单词的隐藏状态 h_n^s , 因此文本短语 $V_{m,n}$ 的语义向量计算为:

$$\gamma_{m,n} = g(h_m, \dots, h_n) = [h_m^r; h_n^s] \quad (18)$$

3.2 节点编码模块

图 4 展示了文本节点的编码过程。首先, 本

文将序列解析成成分树；然后，提取各内部节点所覆盖的短语，将上节计算出的短语语义向量 $\gamma_{m,n}$ 注入相应的内部节点。以 John visited Mary this afternoon 的成分树为例（见图 1），第 2 层的内部节点 ADVP（副词短语）覆盖的短语是 this afternoon，因此在计算此节点记忆单元的时候注入短语 this afternoon 的语义向量 $\gamma_{4,5}$ ，第 3 层的节点 VP（动词短语）为第二层节点 ADVP 与 VP 的父节点，覆盖短语 visited Mary this afternoon，因此在计算此节点记忆单元的时候注入短语 visited Mary this afternoon 的语义向量 $\gamma_{2,5}$ 。

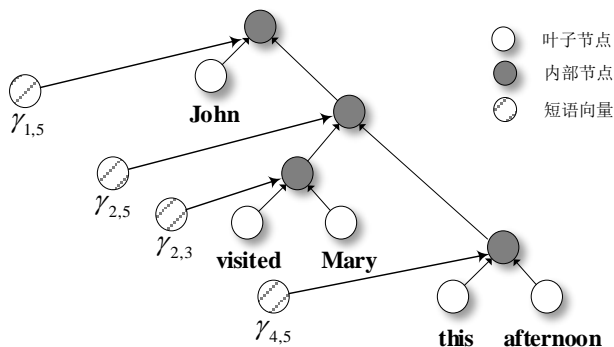


图 4 节点编码模块

最后，将公式 (7) - (13) 在成分树的结构上根据短语语义向量 $\gamma_{m,n}$ 的输入进行改进，则内部节点的记忆单元 c_t 为计算为：

$$c_t = f_t^L \otimes c_{t-1}^L + f_t^R \otimes c_{t-1}^R + i_t \otimes c_t^0 \quad (19)$$

其中：

$$c_t^0 = \tanh \left(W_{\gamma^0} \gamma_{m,n} + \sum_{D=\{L,R\}} W_{h^D} h_{t-1}^D + b_{c^0} \right) \quad (20)$$

$$i_t = \sigma \left(W_{\gamma i} \gamma_{m,n} + \sum_{D=\{L,R\}} (W_{h_i^D} h_{t-1}^D + W_{c_i^D} c_{t-1}^D) + b_i \right) \quad (21)$$

$$f_t^L = \sigma \left(W_{\gamma f^L} \gamma_{m,n} + \sum_{D=\{L,R\}} (W_{h_{f^L}^D} h_{t-1}^D + W_{c_{f^L}^D} c_{t-1}^D) + b_{f^L} \right) \quad (22)$$

$$f_t^R = \sigma \left(W_{\gamma f^R} \gamma_{m,n} + \sum_{D=\{L,R\}} (W_{h_{f^R}^D} h_{t-1}^D + W_{c_{f^R}^D} c_{t-1}^D) + b_{f^R} \right) \quad (23)$$

此处的 t 表示文本句子在成分树上的节点索引。

最后，根据当前节点的记忆单元 c_t 和相应的输出 o_t 计算每个内部节点的隐藏状态 h_t ：

$$o_t = \sigma \left(W_{\gamma o} \gamma_{m,n} + \sum_{D=\{L,R\}} (W_{h_o^D} h_{t-1}^D + W_{c_o^D} c_{t-1}^D) + b_o \right) \quad (24)$$

$$h_t = o_t \otimes \tanh(c_t) \quad (25)$$

其中， $w_{\gamma i}$ ， $w_{\gamma f}$ ， $w_{\gamma c^0}$ ， $w_{\gamma o}$ 为新引入的模型参数。

3.3 训练目标

本文将 SC-TreeLSTM 模型应用于情感分类

和机器阅读理解两种任务。对于情感分类任务，模型的输入是被解析为成分树的句子，句子的每个节点 t 都带有一个真实标签 y_t ，输出是离散的情感标签。由于 softmax 分类器需要在每个节点上完成一次分类计算并获得预测标签 \hat{y}_t ，本任务对训练参数 θ 有如下目标函数：

$$L(\theta) = -\frac{1}{S} \sum_{s=1}^S y_t \log \hat{y}_t + \frac{\lambda}{2} \|\theta\|^2 \quad (26)$$

其中 S 是训练样本中的节点数目， λ 是 L2 正则化超参数。

机器阅读理解任务是通过阅读和理解一段文本，在文本中选择一个连续的片段来回答与文本相关的问题，任务涉及到文本和问题两个输入序列，需要构建两棵树并分别用一个 SC-TreeLSTM 进行编码，输出是文本的两个单词索引。本任务的目标函数如下：

$$L(\theta) = -\frac{1}{K} \sum_{k=1}^K \log(p_{y_k^1}) + \log(p_{y_k^2}) \quad (27)$$

其中， θ 是模型中所有可训练权重的集合， K 为训练样本的数目， y_k^1 和 y_k^2 为第 k 个训练样本真实起止点的索引， p 表示预测结果为起止索引的概率值。

4 实验分析

实验选择情感分类和机器阅读理解两个自然语言处理任务来测试本文提出的 SC-TreeLSTM 模型的性能：（1）情感分类，分析一万多条带标签的电影评论的情感倾向；（2）机器阅读理解，根据文本与问题的句法及语义关系，从文本中抽取问题答案。

4.1 情感分类

数据集介绍 对于情感分类任务，本实验使用与 Zhu 等人相同的数据集 Stanford Sentiment Treebank (SST) [18]，该数据集包含从电影评论中收集的一万多条句子，每个句子都被解析成成分树结构，句子中的每个内部节点都对应一个短语并带有一个人工标记的情感标签，分别对应五种情绪类别：非常消极，消极，中立，积极和非常积极，根节点的标签表示整个句子的情感标签。本文在五分类（细粒度分类）和二分类两种设置上进行实验：对五分类任务，按 8544/1101/2210 的比例将数据集划分为训练集/验证集/测试集；对二分类任务，实验中不考虑标记为中立的样本，将非常消极和消极归为一类，积极和非常积极归为另外一类，且训练集/验证集/测试集按

6920/872/1821 的比例划分。

实验设置 机器硬件配置为 Intel/Xeon E5-2683V3 14核28线程, NVIDIA GTX1080显卡, 32GB 内存, 模型均使用 Tensorflow 深度学习框架实现, 使用 Stanford PCFG Parser^[19]解析输入的序列。

本文在验证集上对模型进行超参数调优, 得到如下训练参数: 模型选择预先训练好的 GloVe 方法^[14]来初始化词向量, 词向量的维度设置为 300, 若遇到 Unknown words 则赋予 300 维正态分布的随机向量。模型的隐藏层维度设置为 150, Embedding 层和 output 层使用的 dropout (keep probability) 分别设置为 0.6 和 0.8。优化器为 Adam^[20], 初始学习率为 0.01, batch size 为 25, L2 正则化系数为 0.0001。

实验结果 表 1 展示了 C-TreeLSTM 模型在五分类和二分类两个情感分类任务上分别训练 10 个 epochs 后的结果。模型对句子根节点 (Root) 标签和短语 (Phrase) 标签的准确度都进行了评估。其中, BiLSTM 是基于顺序结构的神经网络模型, S-LSTM、LSTM-RNN 和 Constituency Tree-LSTMs 均为针对成分树的 LSTM。经过细致的调优过程, 本文对以上模型的实现在准确度上与各自在原始文献中给出的结果相似。因此, 参照本文对各模型的实验结果来评估模型的性能, 将各模型与经过改进的 SC-TreeLSTM 模型进行比较。

表 1 C-TreeLSTM 在 SST 上的测试结果

模型	五分类 (%)		二分类 (%)	
	Root	Phrase	Root	Phrase
BiLSTM ^[21]	49.8	83.3	86.7	-
-Our Implementation	49.6	83.3	86.8	90.2
LSTM-RNN ^[10]	49.9	-	88.0	-
-Our Implementation	49.8	83.5	87.9	90.6
Constituency Tree-LSTMs ^[9]	49.7	-	87.5	-
-Our Implementation	49.7	83.6	87.4	90.8
S-LSTM ^[8]	50.1	-	-	-
-Our Implementation	50.0	83.6	88.2	91.0
SC-TreeLSTM	50.5	84.4	89.2	91.8

整体而言, 相比于 BiLSTM, S-LSTM、LSTM-RNN 和 Constituency Tree-LSTMs 的准确度均有所提升, 说明树形结构在情感分类任务上起到了一定的作用, 且在 S-LSTM 模型下的准确度最高。将短语语义向量加入 S-LSTM 后, 五分类和二分类测试集的准确度得到了普遍提高。对于五分类任务, 根节点标签的准确度提高了 0.5%,

短语标签的准确度提高了 0.7%, 这表明在注入节点所覆盖的短语语义向量后, 短语标签预测的准确度得到了更为明显的提升。对于二分类任务, 根节点标签的准确度提高了 1.0%, 短语标签的准确度提高了 0.8%, 说明短语语义信息对提高二分类任务的准确度作用更大, 这是由于二分类任务的短语级别的标注不够丰富, 只有消极和积极两种, 导致 S-LSTM, LSTM-RNN 和 Constituency Tree-LSTMs 模型在长距离的学习中丢失了学习到的情感信息, 本文的 SC-TreeLSTM 增强了节点对文本语义的记忆, 提高了标签预测的准确度。

表 2 其他模型在 SST 上的测试结果

模型	五分类		二分类	
	准确度 (%)	平均耗时 (s)	准确度 (%)	平均耗时 (s)
LCT-LSTM ^[22]	50.2	23.7	88.5	19.5
-Our Implementation	51.5	24.3	89.1	20.8
Lexicalized TreeLSTM ^[11]	50.5	20.4	89.2	17.7
-Our Implementation	50.5	20.4	89.2	17.7

另外, 本文还进一步对比了情感分类任务在其他模型上的实验结果, 记录了各模型对根节点标签准确度的评估及在两个训练任务上每个 epoch 的平均消耗时间, 如表 2 所示。与 LCT-LSTM 模型相比, 本文的 SC-TreeLSTM 模型在五分类任务上的准确度提高了 0.3%, 在二分类任务上的准确度提高了 0.7%; 与 Lexicalized TreeLSTM 模型相比, 本文的 SC-TreeLSTM 模型在五分类任务上的准确度并未提升, 但训练耗时减少了 3.9s, 在二分类任务上的准确度有所提升, 且训练耗时减少了 3.1s, 这是由于 Lexicalized TreeLSTM 模型需要处理两个方向的节点编码过程且无法并行化, 导致训练时间较长, 效率不高, 而本文的 SC-TreeLSTM 能够更加快速地训练迭代并获得较好的结果。

4.2 机器阅读理解

BiDAF 本实验在开源问答系统 BiDAF 模型^[23]的框架下测试本文提出的 SC-TreeLSTM 模型。BiDAF 模型引入了双向注意流机制来进行文本和问题间的信息交互。该框架由六个部分组成: 1) Character Embedding Layer, 2) Word Embedding Layer, 3) Contextual Layer, 4) Attention Flow Layer, 5) Modeling Layer, 和 6) Output Layer。其中, 三个部分包含 BiLSTM 网络: Contextual Layer 使用具有相同权重的两个 BiLSTM 对文本和问题进行

编码; Modeling Layer 进一步采用双层 BiLSTM 来提取文本中单词的高阶特征; Output Layer 使用另外一个 BiLSTM 来生成用于预测答案片段结束的特征。实验用 S-LSTM、LSTM-RNN、Constituency Tree-LSTMs、LCT-LSTM、Lexicalized TreeLSTM 和 SC-TreeLSTM 分别替换这三个部分的 BiLSTM 进行 BiDAF 模型的构建,其余部分的操作保持不变,将改进后的 BiDAF 模型分别称为 S-BiDAF、LR-BiDAF、CT-BiDAF、LCT-BiDAF、Lex-BiDAF 和 SC-BiDAF。

数据集介绍 本实验使用斯坦福大学开发的问答数据集 SQuAD^[24]。SQuAD 数据集包含 10 万个问答对,被划分为 10570 个验证样本,87599 个训练样本和大量隐藏的测试样本。问答对的答案均为相应文本段落的一部分,从单个单词到多单词文本片段,是可变长度的短语或从句。数据集同时提供几个人工答案,只要模型输出的答案与人工答案之一相符合,模型就被给予一个置信分。实验用两个指标来评估模型:精确匹配(Exact Match, EM)和模糊匹配(F1 score)^[24],分别用于表示精确率及精确率和召回率的加权平均值。

实验设置 机器配置与情感分类实验相同。实验使用斯坦福大学的 CoreNLP 工具^[25]对文本和问题单词进行预处理。对于词向量的构建,选择预先训练好的 GloVe 方法^[13]和卷积神经网络 CNN^[15]来映射,词向量的维度设置为 300,且在训练期间是固定不变的,Unknown words 同样赋予 300 维正态分布的随机向量。本实验的网络参数如下:模型的隐藏层维度设置为 150,dropout 比例为 0.2。在训练过程中,实验设置 batch size 为 32,并使用 Adam 优化器来更新参数;同时,初始学习率设置为 0.2。

表 3 SQuAD 验证集上的实验结果

模型	EM (%)	F1 (%)
BiDAF (Our Implementation)	67.6	77.3
LR-BiDAF	67.9	77.8
CT-BiDAF	68.0	78.2
S-BiDAF	68.2	78.4
LCT-BiDAF	68.8	79.5
Lex-BiDAF	69.0	79.5
SC-BiDAF	69.4	80.0

实验结果 表 3 展示了所有对比模型在 SQuAD 验证集上训练 12 个 epochs 后的实验结果。与 BiDAF 相比,LR-BiDAF 模型的 EM 值提升了 0.3%,F1 值提升了 0.5%;CT-BiDAF 模型的 EM

值提升了 0.4%,F1 值提升了 0.9%;S-BiDAF 模型的 EM 值提升了 0.6%,F1 值提升了 1.1%;LCT-BiDAF 模型的 EM 值提升了 1.2%,F1 值提升了 2.2%;Lex-BiDAF 模型的 EM 值提升了 1.4%,F1 值提升了 2.2%,这是由于 S-BiDAF、LR-BiDAF、CT-BiDAF、LCT-BiDAF 和 Lex-BiDAF 将句法结构特征纳入了文本与问题的表示学习中,改善了文本表达的质量。在内部节点中注入短语语义向量后,本文的 SC-TreeLSTM 模型使得 BiDAF 的准确度得到了更大的提高,相比于 LR-BiDAF,SC-BiDAF 模型的 EM 值提升了 1.5%,F1 值提升了 2.2%;相比于 CT-BiDAF,SC-BiDAF 模型的 EM 值提升了 1.4%,F1 值提升了 1.8%;相比于 S-BiDAF,SC-BiDAF 模型的 EM 值提升了 1.2%,F1 值提升了 1.6%;相比于 LCT-BiDAF 和 Lex-BiDAF,SC-BiDAF 模型的 EM 值分别提升了 0.6%和 0.4%,F1 值均提升了 0.5%。

表 4 模型准确度随句子长度的变化

句子长度 (个)	S-BiDAF		SC-BiDAF	
	EM (%)	F1 (%)	EM (%)	F1 (%)
5	74.4	80.5	75.3	81.7
10	71.7	76.4	72.8	77.9
15	67.5	70.6	69.0	72.5
20	53.8	65.1	55.6	67.3
25	46.6	55.4	48.8	57.9

为了更好地理解模型的性能,实验还在验证集上分析了文本的句子长度对 EM 值和 F1 值的影响,如表 4 所示。可以看出,当句子长度为 5 时,短语信息的注入使得 S-BiDAF 的 EM 提升了 0.9%,F1 值提升了 1.2%,随着句子长度的增加,EM 和 F1 值的提升幅度增大,当句子长度为 25 时,SC-BiDAF 的 EM 值提升了 2.2%,F1 值提升了 2.5%,这说明对于拥有大量长序列的更为复杂的自然语言处理任务,短语语义信息的输入作用更大,对文本建模的辅助更为有效。

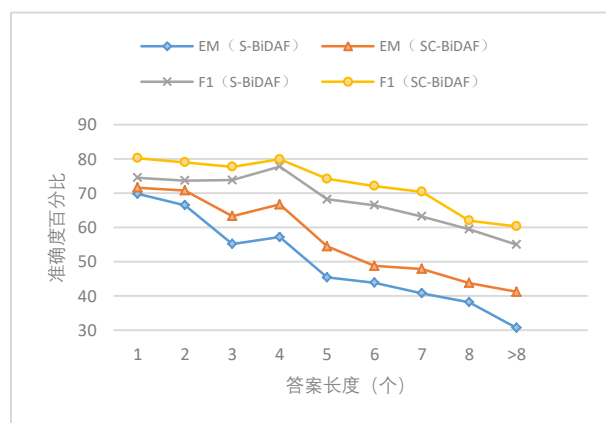


图 5 (a) 模型准确度随答案长度的变化

图 5 (a) 展示了 S-BiDAF 和 SC-BiDAF 模型的准确度随答案长度的变化。随着答案长度的增加, EM 和 F1 值均有所下降, 但速度不同。当答案为多个单词的时候, 本模型的准确度仍然较高。相比于 S-BiDAF 模型, 加入短语语义向量的 SC-BiDAF 模型的 EM 和 F1 值得到普遍提升, 且 F1 值的提升幅度更大。

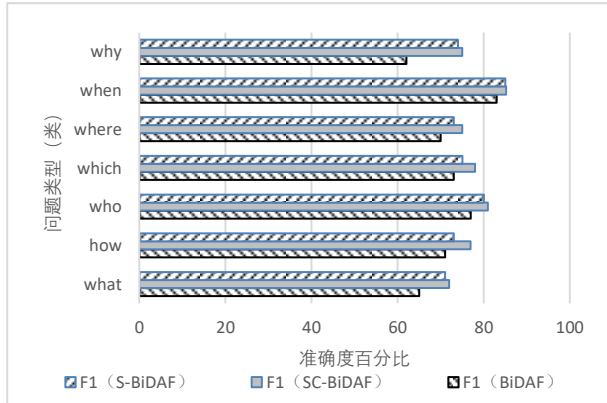


图 5 (b) 模型准确度随问题类型的变化

图 5 (b) 对比了 BiDAF, S-BiDAF 和 SC-BiDAF 模型在七类问题上的准确度。与 BiDAF 及 S-BiDAF 模型相比, 本文的 SC-BiDAF 模型在每种问题上的准确度都有所提高。与 BiDAF 模型相比, S-BiDAF 和 SC-BiDAF 模型在“*How*”, “*What*”, “*Why*”三种类型上的改进尤为显著, 这是由于与其他四种答案大多为实体的类型相比, 这三类问题的答案通常较长, 句法结构更加复杂, 因此 S-BiDAF 和 SC-BiDAF 模型的句法感知特性使其性能优于 BiDAF 模型。与 S-BiDAF 模型相比, SC-BiDAF 模型在“*How*”和“*Which*”两种问题上的准确度有较大的提高, 这是由于 SC-TreeLSTM 在节点编码时加入了短语的语义信息, 降低了答案检索的模糊性和不确定性。

5 相关工作

目前, 针对树结构建模的模型大致可以分为两类: 基于循环神经网络 (RNN) 在树结构上建模^[8,26]和基于卷积神经网络 (CNN) 在树结构上建模^[27]。

基于循环神经网络在树结构上构建的模型, 除了 C-TreeLSTM 模型外, Chen 等人提出了 TreeGRU 模型^[28], 该模型同样在句子的成分树上建模, 具有控制单元内部信息流的门控机制。TreeGRU 模型用 GRU 单元代替 LSTM 单元, 简化了 TreeLSTM 的网络结构。此外, 模型将树结构自底向上节点的隐藏状态作为成分树的信息交

给根节点, 并沿着成分树自顶向下传播到子节点, 形成双向 TreeGRU 网络, 使得每个节点的编码向量包含其上下文节点的句法信息, 但由于该模型无法并行化地处理两个方向的节点编码过程, 导致训练时间较长, 效率不高。

TBCNN^[29,30]是基于卷积神经网络在树结构上构建的典型模型, 该模型的树卷积方法能够在句法树上并行化地提取特征。与基于 RNN 在树结构上构建的模型相比, 该模型的训练效率有很大的提升, 但由于池化操作的空间不变性假设, TBCNN 模型无法在节点间对特征进行组合, 因此模型无法充分利用输入序列的结构信息。

目前存在很多将 CNN 与 RNN 相结合的混合神经网络模型。Wang 等人提出了 regional CNN-LSTM^[31]模型, 该模型将 CNN 和 LSTM 相结合, 利用 CNN 来提取各子句有价值的局部信息并进行加权, 同时利用 LSTM 来捕捉句子之间的长距离依赖关系。Xiao 等人提出的混合模型^[32]将卷积层与 BiLSTM 结合, 使用卷积层来处理输入文本, 并通过池化函数减小序列的长度, 然后将生成的特征提供给 BiLSTM 网络用于后续处理。但上述两种模型都是基于顺序结构来处理输入序列的, 因此计算得到的文本表示只包含了序列的顺序信息。

6 结论

本文提出了一种针对成分树的混合神经网络模型 SC-TreeLSTM, 该模型将 BiLSTM 和 C-TreeLSTM 结合, 在文本成分树的内部节点中注入由 BiLSTM 计算出的相应短语语义向量, 解决了 C-TreeLSTM 模型内部节点信息来源缺失的问题, 增强了节点在长距离编码过程中对文本语义的记忆。本文最后在情感分类和机器阅读理解两种自然语言处理任务上对 SC-TreeLSTM 模型进行了测试, 通过实验结果可以看出, 本文提出的 SC-TreeLSTM 的表现普遍优于 C-TreeLSTM 模型。今后的工作将研究 SC-TreeLSTM 模型在更复杂自然语言处理任务上的适用性。

参考文献

- [1] Peter W. Foltz, Walter Kintsch, Thomas K Landauer. The measurement of textual coherence with latent semantic analysis[J]. Discourse Processes, 1998, 25(2-3):285-307.
- [2] Elman J L. Finding structure in time[J]. Cognitive Science, 1990, 14(2):179-211.

- [3] Mikolov T A. Statistical Language Models Based on Neural Networks[J]. Presentation at Google, Mountain View, 2nd April, 2012.
- [4] 霍欢,张薇,刘亮,等.一种针对句法树的混合神经网络模型[J].中文信息学报, 2017, 31(06):58-66.
- [5] Miwa M, Bansal M. End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures[C]// Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, 2016:1105-1116.
- [6] Chen H, Huang S, Chiang D, et al. Improved Neural Machine Translation with a Syntax-Aware Encoder and Decoder[C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017:1936-1945.
- [7] Gers F A, Schmidhuber J. Recurrent Nets that Time and Count[C]// Ieee-Inns-Enns International Joint Conference on Neural Networks. IEEE Computer Society, 2000:189-194.
- [8] Zhu X, Sobhani P, Guo H. Long Short-Term Memory Over Tree Structures[J]. arXiv preprint arXiv: 1503.04881, 2015.
- [9] Tai K S, Socher R, Manning C D. Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks[J]. Computer Science, 2015, 5(1): 36.
- [10] Le P, Zuidema W. Compositional Distributional Semantics with Long Short Term Memory[J]. Computer Science, 2015,10-19.
- [11] Teng Z, Zhang Y. Bidirectional Tree-Structured LSTM with Head Lexicalization[J]. arXiv preprint arXiv: 1611.06788, 2016.
- [12] Graves A, Jaitly N, Mohamed A R. Hybrid speech recognition with Deep Bidirectional LSTM[C]// Automatic Speech Recognition and Understanding. IEEE, 2014:273-278.
- [13] Pennington J, Socher R, Manning C. Glove: Global Vectors for Word Representation[C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, 2014(14):1532-1543.
- [14] 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.
- [15] Yoon Kim. Convolutional Neural Networks for Sentence Classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, 2014(14):1746-1751.
- [16] Huo Huan, Zhang Wei, Liu Liang, et al. Collaborative Filtering Recommendation Model based on Convolutional Denoising Auto Encoder[C]// Proceedings of the 12th Chinese Conference on Computer Supported Cooperative Work and Social Computing (ChineseCSCW '17). ACM, New York, NY, USA, 2017,64-71.
- [17] Cross J, Huang L. Span-Based Constituency Parsing with a Structure-Label System and Provably Optimal Dynamic Oracles[C]// Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016:1-11.
- [18] 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, 2013:1631-1642.
- [19] Klein D, Manning C D. Accurate unlexicalized parsing[C]// Proceedings of Meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2003:423-430.
- [20] Kingma D P, Ba J. Adam: A Method for Stochastic Optimization[J]. Computer Science, 2014, 1412.6980:1-15.
- [21] Jiwei Li, Thang Luong, Dan Jurafsky, et al. When Are Tree Structures Necessary for Deep Learning of Representations? [C]// Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015: 2304-2314.
- [22] Ling Gan, Houyu Gong. Text Sentiment Analysis based on Fusion of Structural Information and Serialization Information[C]// Proceedings of the Eighth International Joint Conference on Natural Language Processing, 2017: 336-341.
- [23] Seo M, Kembhavi A, Farhadi A, et al. Bidirectional Attention Flow for Machine Comprehension[J]. arXiv preprint arXiv: 1611.01603, 2016.
- [24] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev. SQuAD: 100, 000+ Questions for Machine Comprehension of Text[C]// Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016:2383-2392.
- [25] Christopher D. Manning, Mihai Surdeanu, John Bauer. The Stanford CoreNLP Natural Language Processing Toolkit[C]// Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 2014:55-60.
- [26] Alexandros Potamianos, Filippos Kokkinos. Structural Attention Neural Networks for improved sentiment analysis[C]// Proceedings of the 15th Conference of the European Chapter of the Association for Computational

- Linguistics, 2017:586-591
- [27] Ma M, Huang L, Xiang B, et al. Dependency-based Convolutional Neural Networks for Sentence Embedding[C]// Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, 2015:174-179.
- [28] Huadong Chen, Shujian Huang, David Chiang, et al. Improved Neural Machine Translation with a Syntax-Aware Encoder and Decoder[C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017:1936-1945.
- [29] Lili Mou, Hao Peng, Ge Li, et al. Discriminative Neural Sentence Modeling by Tree-Based Convolution[C]// Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015:2315-2325
- [30] Mou L, Yan R, Li G, et al. Backward and Forward Language Modeling for Constrained Sentence Generation[J]. Computer Science, 2016, 4(6):473-482.
- [31] Jin Wang, Liang-Chih Yu, K. Robert Lai, et al. Dimensional Sentiment Analysis Using a Regional CNN-LSTM Model[C]// Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, 2016:225-230.
- [32] Xiao Y, Cho K. Efficient Character-level Document Classification by Combining Convolution and Recurrent Layers[J]. arXiv preprint arXiv: 1602.00367, 2016.

作者联系方式: 霍欢 上海市杨浦区军工路 516 号 200093 15601787212 huo_huan@yahoo.com

作者联系方式: 薛瑶环 上海市杨浦区军工路 516 号 200093 13262637687 1029621106@qq.com