



计算机应用研究  
Application Research of Computers  
ISSN 1001-3695, CN 51-1196/TP

## 《计算机应用研究》网络首发论文

题目: 基于自注意力机制的方面情感分类  
作者: 王拂林, 刘丹, 昌茜  
DOI: 10.19734/j.issn.1001-3695.2019.07.0259  
收稿日期: 2019-07-10  
网络首发日期: 2019-12-26  
引用格式: 王拂林, 刘丹, 昌茜. 基于自注意力机制的方面情感分类[J/OL]. 计算机应用研究. <https://doi.org/10.19734/j.issn.1001-3695.2019.07.0259>



**网络首发:** 在编辑部工作流程中, 稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定, 且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式(包括网络呈现版式)排版后的稿件, 可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定; 学术研究成果具有创新性、科学性和先进性, 符合编辑部对刊文的录用要求, 不存在学术不端行为及其他侵权行为; 稿件内容应基本符合国家有关书刊编辑、出版的技术标准, 正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性, 录用定稿一经发布, 不得修改论文题目、作者、机构名称和学术内容, 只可基于编辑规范进行少量文字的修改。

**出版确认:** 纸质期刊编辑部通过与《中国学术期刊(光盘版)》电子杂志社有限公司签约, 在《中国学术期刊(网络版)》出版传播平台上创办与纸质期刊内容一致的网络版, 以单篇或整期出版形式, 在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊(网络版)》是国家新闻出版广电总局批准的网络连续型出版物(ISSN 2096-4188, CN 11-6037/Z), 所以签约期刊的网络版上网络首发论文视为正式出版。

# 基于自注意力机制的方面情感分类

王拂林, 刘 丹, 昌 茜

(电子科技大学 电子科学技术研究院, 成都 611731)

**摘 要:** 基于方面的情感分类判断句子中给定实体或属性的情感极性, 针对使用全局注意力机制计算属性词和句子其他词的注意力分数时, 会导致模型关注到与属性词不相关的词, 并且对于长距离的依赖词、否定词关注不足, 不能检测到并列关系和短语的问题。提出了基于自注意力机制的语义加强模型(semantic reinforcement model based on self-attention mechanism, SRSAM), 模型首先使用双向长短时记忆神经网络模型(bidirectional long short-term memory, BiLSTM)获取文本编码, 其次用自注意力机制计算文本编码的多个语义编码, 最后将属性词和语义编码交互后判断属性词在句中的情感极性。使用 SemEval 2014 数据集实验, 由于模型能发现长距离依赖和否定词, 对并列关系和短语有一定检测效果, 相比基础模型在分类精度上有 0.6%~1.5% 的提升。

**关键词:** 方面词; 情感分类; 自注意力机制; 语义编码

**中图分类号:** TP391 doi: 10.19734/j.issn.1001-3695.2019.07.0259

## Aspect-level sentiment classification based on self-attention mechanism

Wang Fulin, Liu Dan, Chang Xi

(Research Institute of Electronic Science & Technology, University of Electronic Science & Technology of China, Chengdu Sichuan 611731, China)

**Abstract:** Aspect-level sentiment classification to determine the emotional polarity from sentence towards a specific aspect word. When using the global attention mechanism to calculate the attention scores of attribute words and other words of the sentence, the model will focus on words that are not related to the attribute words, and pay insufficient attention to long-distance dependent words and negative words, and cannot detect side-by-side relationships or phrases. A semantic enhancement model based on the self-attention mechanism is proposed. The model first uses the bidirectional long short-term memory model to obtain the text encoding, and then uses the self-attention mechanism to calculate the multiple semantic encodings of the text encoding. Finally, the attribute words and semantic coding are used to interact to determine the emotional polarity of the attribute words in the sentence. Using the SemEval 2014 dataset experiment, since the model can find long-distance dependence and negative words, it has a certain detection effect on the parallel relationship and the phrase, and the classification accuracy is 0.6%~1.5% higher than the basic model.

**Key words:** aspect word; sentiment classification; self-attention mechanism; semantic coding

## 0 引言

最近几年, 自然语言处理的一个分支: 情感分类(sentiment classification), 逐渐受到研究者的关注。

学术界中情感分析也称为观点挖掘(opinion mining)<sup>[1]</sup>, 其研究目标是从文本中分析展示人们对于实体及其属性所表达的观点、情感、评价、态度和情绪。

在大数据时代, 网络上有大量消费者对于电商、餐饮、旅游、影视等各行业各种评论, 从这些评论中不仅可以分析出消费者对于商品、服务的整体态度, 并且可以从更细粒度的角度分析出消费者对于商品或服务的某一个属性的满意度, 这样商家可以根据消费者的反馈知道自己产品或服务存在的不足和优势, 从而能够进行改进。另一方面消费者也可以根据这以往消费者的反馈找到自己需要的商品和服务。由于评论数据量大, 人工分析识别的效率低, 因此自动化的文本情感分析成为了必要手段, 广泛的应用到医疗、旅游、金融、舆情分析等领域<sup>[2]</sup>。

传统的情感分类根据情感词典、机器学习等方法判断句子或文档整体所表达的情感时积极、中立、还是消极<sup>[2, 3]</sup>, 由

于这种方式不能判断出一个句子中不同实体或实体属性的情感极性, 而产生了细粒度的基于方面的情感分类(aspect-level sentiment classification)<sup>[2]</sup>。句子实体或实体属性称为方面词, 基于方面的情感分类是在已知句子中方面词的情况下, 判断不同方面词的极性<sup>[5]</sup>, 例如对于句子 “The price is reasonable although the service is poor.”, 句子中的 price、service 为方面词, 对于方面词 price 态度是积极的, 对于方面词 service 的态度是消极的。

注意力机制(attention mechanism)模拟了人们关于一幅场景能够对不同的部分有不同的关注程度, 应用到了机器翻译、文本摘要、问答系统等领域<sup>[2, 5]</sup>, 也应用到了基于方面的情感分析任务<sup>[19-26]</sup>。判断句子中方面词的情感时, 对于句中不同词的关注度是不一样的, 例如对于句子中的标点符号应该减少关注度, 对于情感词应该加强关注。前人研究的模型通常计算其他词和方面词的相关度分数, 虽然模型能够将注意力集中在那些和方面词相关的词上<sup>[18, 21]</sup>, 但难以发现否定、并列等其他词之间隐含的对判断方面词情感极性有用的信息<sup>[17]</sup>, 另外使用 LSTM(long-short term memory, LSTM)虽然能够捕捉到词语之间关系特征, 但对于长距离依赖关系发现较

收稿日期: 2019-07-10; 修回日期: 2019-09-17

**作者简介:** 王拂林(1994-), 男, 四川人, 硕士研究生, 主要研究方向为大数据与云计算(phelanwang@foxmail.com); 刘丹(1969-), 男, 四川人, 副教授, 硕导, 博士, 主要研究方向为网络安全、大数据及云计算、分布式并行计算; 昌茜(1994-), 女, 四川人, 硕士研究生, 主要研究方向为大数据与网络安全。

弱<sup>[6, 8]</sup>。

Ashish Vaswani 等人于 2017 年提出基于多头注意力机制 (multi-head attention) 的翻译模型, 与基于 RNN(recurrent neural network)的端到端的翻译模型框架不同, 该模型在编码和解码过程中都完全使用多头自注意力机制, 相比 RNN 能够学习到长距离依赖, 模型在 WMT2014 数据集的英德和英法翻译任务上取得了先进结果<sup>[6, 14]</sup>。Tan 等人为了在语义角色标注任务上发现标签之间的依赖关系引入自注意力机制, 在 CoNLL-2005 和 CoNLL-2012 的 SRL 数据集上都取得了先进结果<sup>[13]</sup>。此外自注意力机制还应用到关系抽取、文本表示<sup>[7, 9-11]</sup>, 阅读理解等领域<sup>[12]</sup>, 对于词语间的长距离依赖, 词语间的相关关系发现有重要意义。

由上述原因, 提出了基于自注意力机制的方面情感分析模型, 根据 Lin<sup>[8]</sup>提出的自注意力语义编码结构, 对每个句子中的不同词赋予不同的权重, 能够表示该句子某个维度的语义信息, 例如对句中情感词分配较高的权重能够得到和句子情感相关的语义; 对句中标点符号、连词赋予较高的权重能够得到句子的分割点相关的语义; 对句子中的短语组成词赋予较高的权重, 能够得到句子的短语组成相关的语义。

方面词通常有固定的修饰词集合, 例如方面词 price 的情感修饰词通常为 low、high、reasonable 等, 而修饰方面词 food 通常用 fresh、good、tasty 等。如果在判断给定方面词的情感倾向时能够准确的定位到与其对应的修饰词集合, 则能够提高判断方面词情感极性的准确率。因此用自注意力机制获取情感词语义、句子分割点语义等多个维度的语义向量, 情感词语义向量使判断给定方面词的情感极性更准确, 句子分割点语义根据句子分割点让模型对于其他方面词的情感修饰词减少关注。由于在这些语义向量中存在对于分类无关的向量, 所以再将这些语义向量和方面词进行交互, 通过有监督训练使模型自适应的分配较高权重到对于情感分类有效的语义向量上, 以达到提高分类精度的效果。

模型首先用双向 LSTM(bi-directional RNN, Bi-RNN)获取每个词和临近词之间的依赖关系, 表示为双向 LSTM 的隐层向量。接着用自注意力机制计算句子的多个语义编码, 每个语义编码代表了句子某个维度的语义信息<sup>[6]</sup>。然后方面词向量和句子的语义编码向量进行多次迭代交互得到更抽象的编码信息, 使模型关注到和方面词情感倾向相关的语义编码, 最后用编码信息进行分类预测。同时为了融入更多的有用信息, 将方面词向量和词的情感向量拼接到句子的词嵌入上作为 LSTM 的输入<sup>[20, 30]</sup>, POSP-CNNAM<sup>[31]</sup>也表明在对模型输入融入更多的信息后能够进一步提高分类的准确率。

基于方面的情感分析找出给定方面词在句子中的情感极性。传统的 SVM(support vector machine)<sup>[3]</sup>统计所有句子的词信息, 利用词袋模型对方面词进行情感倾向进行判断, SVM 的模型效果对于特征的质量有很大关系, 并且特征分析耗时耗力。相比之下基于神经网络的模型能够自动发现重要特征, 特别是能够发现词语间的关系 LSTM 网络。使用 LSTM<sup>[15, 16]</sup>网络进行情感分析有一定的效果, 但对于基于方面的情感分析准确率较低<sup>[19]</sup>, TD-LSTM、TC-LSTM<sup>[18]</sup>从方面词处将句子分割成两段, 将其分别用 LSTM 编码得到的输出拼接后分类, 提高了准确率, 但模型忽略了方面词和其他词之间的关系。

随后的模型引入了注意力机制: ATAELSTM<sup>[19]</sup>将方面词嵌入连接到 LSTM 的输入词向量和输出隐层向量上, 方面词嵌入和输出隐层向量使用注意力机制得到语义编码, 使模型能够发现方面词和其他词的关系; ALAN<sup>[20]</sup>对方面词和文本分别编码, 并融入了情感词的情感极性; IAN<sup>[21]</sup>将方面词和文本词交互后将结果拼接分类。虽然上述模型使用注意力

机制改善了效果, 但方面词编码和文本编码只简单的交互了一次, 捕捉的方面词和其他词的关系不足。

动态记忆网络(dynamic memory networks, DMN)<sup>[22]</sup>首先用方面词向量和文本词向量做注意力机制, 同时方面词向量做一个线性映射, 然后两者的结果相加, 这个过程称之为一个 hop, DMN 进行多次 hop 得到高维的语义向量。FCMN<sup>[30]</sup>在 DMA 的向量中加入词性、情感词极性、位置信息, 另外 coattention-LSTM, coattention-MemNet<sup>[26]</sup>, 中都使用了注意力机制, 用方面词计算和句子中其他词的注意力得分, 发现和方面词相关的词, 但获取其他词之间的关系不足。

从直觉上来看, 离方面词较近的词对于判断方面词的极性的要比离得远的更高, PosATT-LSTM<sup>[28, 29]</sup>从绝对位置的角度加强了位置关系, 事实上并不是所有情感修饰词都靠近方面词, 绝对位置对于长距离依赖词的发现不足; LWOA-LSTM<sup>[31]</sup>针对绝对位置计算的不足改进位置计算为动态调整的方式; SHAN<sup>[32]</sup>从语义距离的角度来加强位置关系, 但评论中存在口语化、语法不规范的情况, 对于依存句法解析正确性会产生影响, 导致对结果产生影响。

## 1 模型

本文的 SRSAM 模型分成三个主要部分如图 1 所示, 第一个部分是输入向量通过双向 LSTM 得到隐层向量, 隐层向量中包含了临近词之间的依赖关系; 第二部分是隐层向量通过自注意力机制得到不同的语义编码向量, 每个语义编码向量代表了不同维度的语义信息; 第三部分是通过多层 hop, 获取方面词和不同语义编码之间的关系。

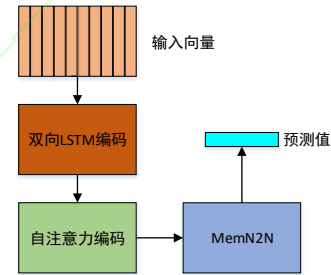


图 1 模型总体结构

Fig. 1 Overall structure of model

### 1.1 任务定义和概念

一个句子  $C$  包含  $n$  个词, 表示为

$$C = [w_1, w_2, \dots, w_n] \quad (1)$$

句子中的每个词表示为  $w_i$ , 句子中包含方面词表示为

$$T = [w_1, w_2, \dots, w_m], \quad (0 < m \leq n) \quad (2)$$

基于方面的情感分析任务是判断  $T$  在  $C$  中的情感极性。例如 “Great food but the service was dreadful!”, 方面词 food 的是积极的情感倾向, service 是消极的情感倾向。

首先需要通过词嵌入矩阵  $V_w \in R^{d \times |V|}$  <sup>[33]</sup> 将每个词  $w_i$  转换为词向量  $e_i$ , 其中  $d$  是词向量维度,  $|V|$  是词语总数,  $e_i \in R^d$  是  $V_w$  中对应于  $w_i$  的一列向量, 每个句子  $E_s$  表示为

$$E_s = [e_1, \dots, e_n] \quad (3)$$

其中  $E_s \in R^{d \times n}$ ,  $n$  表示句子长度, 方面词向量为

$$T_s = [e_{i_1}, e_{i_2}, \dots, e_{i_m}] \quad (4)$$

其中  $T_s \in R^{d \times m}$ ,  $m$  是句子中方面词的长度, 方面词可能有多个单词, 取所有词向量的平均值作为方面词嵌入, 计算公式如下

$$e_a = \frac{1}{m} \times \sum_{i=1}^m e_{i_i} \quad (5)$$

为了将更多有用信息添加到词嵌入中, 通过词典嵌入矩阵  $L_w \in R^{k \times |V|}$  给每个词  $w_i$  生成一个词典嵌入向量  $l_i$  <sup>[20]</sup>,  $k$  表示每



个词典嵌入向量的维度,  $l_i$  是词  $w_i$  在词典嵌入矩阵  $L_w$  中对应的向量。每个情感词典中对于词  $w_i$  有一个情感极性评分, 词典嵌入向量  $l_i$  由  $w_i$  在所有情感词典中的评分拼接而成, 对于情感词典中不包含的词评分设置为 0, 用到的情感词典如下:

- Bing Liu Opinion Lexicon.
- NRC Hashtag Sentiment Lexicon.
- Sentiment140 Lexicon.
- NRC Sentiment140 Lexicon.
- MaxDiff Twitter Sentiment Lexicon.
- National Research Council Canada (NRC) Hashtag Affirmative and Negated Context Sentiment Lexicon.
- Large-Scale Twitter-specific Sentiment Lexicon.

对于句子  $S$  的词典嵌入矩阵表示为

$$L_s = [l_1, \dots, l_i, \dots, l_n], L_s \in R^{k \times n} \quad (6)$$

将文本词嵌入向量、词典嵌入向量、方面词嵌入向量拼接作为输入:

$$S = [s_1, s_2, \dots, s_n], S \in R^{(2d+k) \times n} \quad (7)$$

每个拼接后每个词的表示为  $s_i = [e_i, e_a, l_i]$  维度为  $2d+k$ , 如图 2 所示。

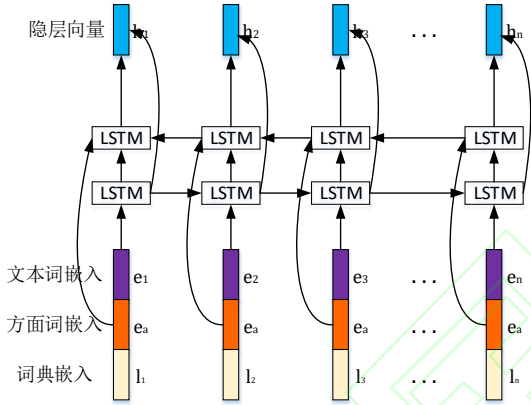


图 2 文本词嵌入向量、词典嵌入向量、方面词嵌入向量拼接作为输入通过双向 LSTM 得到隐层向量

Fig. 2 Text word embedding vector, dictionary embedding vector and aspect word embedding vector are used as input to obtain hidden layer vector through bidirectional LSTM.

## 1.2 双向 LSTM 编码

本文模型的第一部分先对输入进行双向 LSTM 编码, LSTM 能够克服 RNN 网络存在的梯度消失和梯度下降的缺陷, 每个 LSTM 单元有输入门  $i$ 、遗忘门  $f$ 、输出门  $o$  和记忆状态  $c$ , 隐层状态  $h_t$  的更新公式为

$$i_t = \sigma(W_i[h_{t-1}; s_t] + b_i) \quad (8)$$

$$f_t = \sigma(W_f[h_{t-1}; s_t] + b_f) \quad (9)$$

$$o_t = \sigma(W_o[h_{t-1}; s_t] + b_o) \quad (10)$$

$$g_t = \tanh(W_g[h_{t-1}; s_t] + b_g) \quad (11)$$

$$c_t = i_t \odot g_t + f_t \odot c_{t-1} \quad (12)$$

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

其中:  $\sigma$  是 sigmoid 函数,  $\odot$  是按位乘运算,  $s_t$  是输入向量,

$$W_i, W_f, W_o \in R^{\left(\frac{d}{2}\right) \times \left(\frac{d}{2} + 2d + k\right)}, b_i, b_f, b_o, b_g \in R^{\frac{d}{2}}.$$

前向 LSTM 能够获得处于  $t$  时刻的词与  $[1, \dots, t-1]$  时刻词的关系, 不能得到与  $[t+1, \dots, n]$  时刻词的关系, 因此本文模型使用双向 LSTM 将前向隐层单元

$$H_f = [\vec{h}_1, \vec{h}_2, \dots, \vec{h}_n], H_f \in R^{n \times \left(\frac{d}{2}\right)}$$

和后向隐层单元

$$H_b = [\overleftarrow{h}_1, \overleftarrow{h}_2, \dots, \overleftarrow{h}_n], H_b \in R^{n \times \left(\frac{d}{2}\right)}$$

拼接得到

$$H = [(\vec{h}_1, \overleftarrow{h}_1), (\vec{h}_2, \overleftarrow{h}_2), \dots, (\vec{h}_n, \overleftarrow{h}_n)] = [h_1, h_2, \dots, h_n] \quad (14)$$

其中  $H \in R^{n \times d}$ ,  $H$  表示上下文的语义关系, 过程如图 2 所示。

## 1.3 自注意力语义编码

第二部分使用自注意力机制对隐层向量进行进一步的编码, 为获取某一维度的语义信息, 对于每个隐层输出的权重计算过程如图 3 所示, 公式为

$$a = \text{softmax}(w_{s_2} \tanh(W_{s_1} H^T)) \quad (15)$$

其中  $a \in R^{1 \times n}$ ,  $H^T \in R^{d \times n}$  是隐层单元矩阵的转置,  $W_{s_1} \in R^{d_a \times d}$ ,  $w_{s_2} \in R^{1 \times d_a}$  为权重矩阵,  $d_a$  为一个超参数。softmax 函数让所有的权重之和为 1, 计算公式为

$$\text{softmax}(g_i) = \frac{e^{g_i}}{\sum_j e^{g_j}} \quad (16)$$

根据权重  $a$  得到的一个语义向量  $A_i$  计算为

$$A_i = a \times H \quad (17)$$

其中  $A_i \in R^{1 \times d}$ ,  $a$  表示了为了得到某一语义关系而赋予隐层单元的不同权重, 得到的语义向量通常会关注到句子中一些特别的成分, 比如短语、否定词、连词、情感词等。

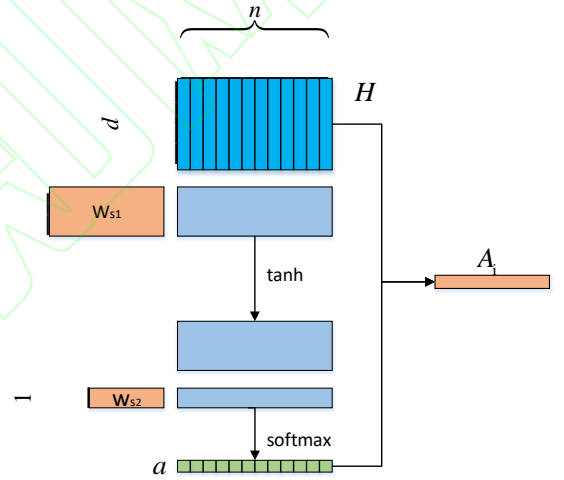


图 3 从隐层单元计算单个语义向量的过程

Fig. 3 The process of computing a single semantic vector from a hidden layer cell

获取句子的多个维度的语义信息, 对  $w_{s_2}$  进行拓展, 计算如下:

$$A = \text{softmax}(W_{s_2} \tanh(W_{s_1} H^T)) \quad (18)$$

其中  $A \in R^{r \times n}$ ,  $W_{s_2} \in R^{r \times d_a}$  为扩展的权重矩阵,  $r$  为超参数, 表示最终需要得到的上下文语义嵌入向量个数, 通过这种嵌入方式能够获取多方面的文本语义向量:

$$M = A \times H \quad (19)$$

其中  $M \in R^{r \times d}$ ,  $A \in R^{r \times n}$  表示  $r$  个维度的权重,  $M \in R^{r \times d}$  表示根据权重向量获取的  $r$  个语义编码向量。

## 1.4 端到端记忆网络

最后使用端到端的记忆网络 (end-to-end memory networks, MemN2N)<sup>[34]</sup>, 将 MemN2N 用于问答系统中类似, 基于方面的情感分析中将方面词作为 MemN2N 模型中的答案。本文模型将方面词编码和每个语义编码计算一个权重, 然后对语义编码加权求和得到一个输出, 同时方面词编码非线性映射得到另一个输出, 将两个输出相加得到交互编码, 这个过程称之为一个 hop。将方面词编码和文本编码进行多次 hop, 用最后的输出进行预测, 一个 hop 的过程如图 4(a) 所示, 两个 hop 的过程如图 4(b) 所示。

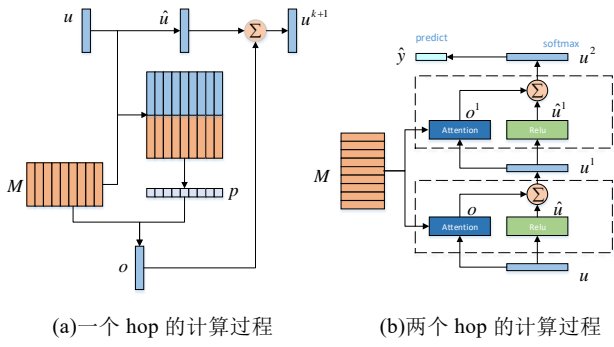


图 4 hop 的计算过程

Fig. 4 The calculation process of hop

每个 hop 的计算过程如图 4(a)所示, 计算公式如下:

$$g_i = \text{ReLU}(W_{att}[m_i; u]) + b_{att} \quad (20)$$

将方面词编码和包含上下文语义的编码  $m_i$  做一次交互获取对于  $m_i$  的权重,  $m_i$  是自注意力机制中得到的第  $i$  个语义编码,  $u$  的初值为方面词向量  $e_a$ , 参数  $W_{att} \in R^{n \times 2d}$ ,  $b_{att} \in R^{n \times 1}$ 。

$$p_i = \text{softmax}(g_i) \quad (21)$$

经过 softmax 使得权重之和能够为 1。

根据每个语义编码的权重, 计算权重和得到输出向量:

$$o = \sum_{i=1}^n p_i \times m_i \quad (22)$$

对方面词编码进行非线性映射:

$$\hat{u} = \text{ReLU}(W_{reu}u) + b_{reu} \quad (23)$$

其中  $W_{reu} \in R^{n \times d}$ ,  $b_{reu} \in R^{n \times 1}$  为参数, 最后将两方面的记忆信息融合:

$$u^{k+1} = \hat{u}^k + o^k \quad (24)$$

每个 hop 可类比当前的信息和以前的记忆进行一次交互获取的新的记忆, 经过多次 hop 之后用最后输出出来预测:

$$\hat{y} = \text{softmax}(W(\hat{u}^n + o^n)) \quad (25)$$

其中  $W \in R^{n \times d}$  为参数。

## 1.5 模型训练

模型使用有监督的训练方式, 损失计算为预测的结果和实际结果交叉熵, 同时为了防止过拟合使用 dropout 和 L2 正则化:

$$\text{loss} = \sum_i \sum_j y_i^j \log \hat{y}_i^j + \sum_{\theta \in \Phi} \lambda \theta^2 \quad (26)$$

$i$  是第句子的索引,  $j$  是第类别的索引,  $\lambda$  是 L2 正则项的系数,  $\Phi$  是所有的参数,  $y_i^j$  是第  $i$  个句子的实际类别,  $\hat{y}_i^j$  是对应的预测类别。

## 2 实验

### 2.1 数据集

实验使用 SemEval2014 Task4<sup>[5]</sup>的餐馆领域(restaurant)和电脑领域(laptop)的评论数据集。数据集中的方面词情感极性有积极、中立、消极、矛盾, 由于矛盾的评论数量较少, 将其排除, 餐馆领域训练集 3608 条评论、测试集 1120 条评论, 电脑领域训练集 2328 条评论、测试集 638 条评论, 统计结果如表 1 所示。

表 1 评论数据集统计

Tab. 1 Comment dataset statistics

	餐饮数据集		电脑数据集	
	train	test	train	test
positive	2164	728	994	341
neutral	637	196	464	169
negative	807	196	870	128
total	3608	1120	2328	638

### 2.2 实验环境及参数设置

实验环境为 64 位 Ubuntu 16.04 操作系统、Intel Xeon E312xx 6 核 CPU、8G 内存、模型搭建使用 tensorflow-1.13.1。

实验使用 Glove<sup>[33]</sup>的 300 维预训练词向量。隐层大小设置为词嵌入维度的一半, 超参数  $r$  设置为 30, 超参数  $da$  设置为 300, 权重用分布  $U(-\epsilon, \epsilon)$ ,  $\epsilon=0.01$  初始化, L2 正则项设置为 0.0001, hop 进行多次实验选择。使用 AdaGrad 优化算法, 小批量梯度下降算法, 每个批次 32 个样本。

### 2.3 实验算法流程

本文实验的算法具体执行流程如下所示。

a) 从数据集中分离评论语句、评论中的方面词、评论中方面词的极性, 对应于一个方面词生成一个数据样本为: (评论句、评论方面词、方面词在句子极性)。

b) 将数据集进行分词, 根据 Glove 预训练词向量将文本数据集转换为词向量矩阵, 类别标签转换为 one-hot 编码, 得到转换后的数据集。

c) 根据分词后的数据集和情感词典, 得到每个词的情感词向量和每条评论的情感矩阵, 将条评论的情感矩阵拼接到每条评论的词向量矩阵上作为输入数据集, 具体参照 1.1 节。

d) 根据第 1 节内容所述构建 SRSAM 模型, 并称模型为 M1, 同时构建不经过自注意力机制编码的模型 M2 来对比效果。

e) 设置 M1 和 M2 模型第三部分 MemN2N 中的 hop 为 1-9 进行对比实验, 对比不同 hop 对于实验结果的影响。

f) 分别使用电脑评论数据集(laptop)和餐饮评论数据集(restaurant)进行实验。

### 2.4 模型比较

本文 SRSAM 模型和以下 baseline 方法进行比较:

a) Majority: 根据训练集中数据简单的判断测试集中方面词的极性。

b) LSTM<sup>[19]</sup>: 使用标准的 LSTM 循环神经网络, 句子序列作为输入, 最后一层的隐层状态作为输入输出。

c) TD-LSTM<sup>[20]</sup>: 从方面词将句子分成两段, 分别使用前向 LSTM 和后向 LSTM 处理。

d) TC-LSTM<sup>[20]</sup>: TD-LSTM 的拓展, 将方面词向量连接到每个 LSTM 的输入向量上

e) ATAE-LSTM<sup>[18]</sup>: 模型应用方面词向量和词典向量, 基于 LSTM 用注意力机制计算输出。

f) MemNet<sup>[22]</sup>: 用深度记忆网络代替基于循环神经网络的模型, 让方面词和文本进行多次注意力交互。

g) IAN<sup>[21]</sup>: 在文本和方面词之间交互使用注意力机制, 生成方面词和文本的编码。

h) RAM<sup>[24]</sup>: 对通过双向 LSTM 的词嵌入编码应用 MemNet, 多次注意机制应用获取依赖关系。

表 2 中展示了在 laptop 评论数据集和 restaurant 评论数据集上的模型结果。最基本的 Majority 模型效果最差, LSTM 相比 Majority 模型在 restaurant 数据集上的效果有大幅度提升, TD-LSTM、TC-LSTM 将每个样本分成两段使用 LSTM, 能够发现一个方向上的依赖关系, 分类效果得到了提升。上述深度学习方法都没有使用注意力机制, 基于 LSTM 模型的 AE-LSTM、ATAE-LSTM、IAN 模型加入了注意力机制, 分类精度提高了 1%~2%。

相比表中的其他模型, 本文的 SRSAM 模型取得了最好结果, 特别是在 laptop 评论数据集上。本文的模型和 RAM 模型最相似, 但本文的模型中加入了自注意力机制, 将文本编码为语义向量使得模型能够捕获到更多的语义信息, 相比 RAM 在 laptop 数据集上的准确率提高了 0.6%, 在 restaurant 评论数据集上的准确率提高了 1.4%。

表 2 模型结果对比

Tab. 2 Model performances

methods	laptop	restaurant
Majority	0.650	0.535
LSTM	0.665	0.743
TD-LSTM	0.681	0.756
TC-LSTM	0.682	0.760
AE-LSTM	0.689	0.762
ATAE-LSTM	0.687	0.772
MemNet	0.703	0.781
IAN	0.721	0.786
RAM	0.744	0.802
SRSAM	0.750	0.816

## 2.5 自注意力编码有效性

为验证自注意力编码对于数据集分类精度提升上的有效性进行了引入注意力机制和未引入注意力机制的模型在数据集上的分类精度的对比。由图 1 中, 有自注意力编码过程的 SRSAM 称为 M1, 没有第二个自注意力机制编码过程的模型称为 M2, 设置第三个过程的 hop 从 1 到 9 进行了 M1 和 M2 在 laptop 和 restaurant 数据集上的分类精度对比, 结果数据如表 3 和图 5 所示。

表 3 M1 和 M2 在两个数据集上不同 hop 的分类精度对比

Tab. 3 Comparison of classification accuracy of M1 and M2 on two

data sets with different hop				
hops	M1_Laptops	M2_Laptops	M1_Restaurant	M2_Restaurant
1	0.7272	0.7304	0.8000	0.8000
2	0.7178	0.7304	0.7928	0.8008
3	0.7460	0.7413	0.8071	0.8071
4	0.7319	0.7335	0.8017	0.8053
5	<b>0.7501</b>	0.7391	0.8151	0.8053
6	0.7445	0.7351	0.8044	0.8026
7	0.7319	<b>0.7476</b>	0.8026	0.7991
8	0.7319	0.7272	<b>0.8167</b>	<b>0.8098</b>
9	0.7364	0.7342	0.8151	0.8026

从表 3 中可知在第 4 个 hop 之前 M2 的分类准确率和 M1 的分类准确率的波动较大, 不能明确知道哪个模型准确率比较高, 从第 4 个 hop 开始, M1 的准确率比 M2 的准确率在 restaurant 数据集上高出 0.35%-0.98%, 在 laptop 数据集上的准确率高出 0.22%-0.94%, 但是在 laptop 数据集上第 7 个 hop 的 M2 的准确率高于 M1。laptop 数据集在 M1 上的第 5 个 hop 有最高精度 0.7501%, 在 restaurant 数据集在 M1 的第 8 个 hop 有最高精度 0.8167%, 对比两个数据集和各个 hop, 添加了自注意力机制的 M2 的分类精度是高于 M1 的, 因此自注意力编码对于分类精度的提升是有效果的, M1 和 M2 在两个数据集上的分类精度对比图 5 所示。

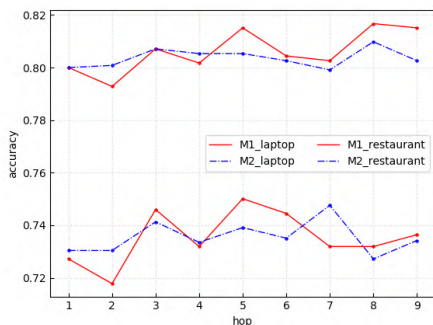


图 5 M1 和 M2 在两个数据集上不同 hop 的分类精度对比

Fig. 5 Comparison of classification accuracy of M1 and M2 on two data sets with different hop

## 2.6 模型可视化

使用自注意力机制对文本向量编码成语义向量的过程中, 对于不同的隐层向量有不同的权重, 从而得到不同的语义向量。可以将这些权重进行可视化来表明模型的有效性。

实验中取的  $r$  为 30, 将隐层向量编码为 30 个语义向量, 每个语义向量对于每个隐层向量都有权重, 取其中两个可视化结果如图 6 和 7 所示。图 6 中对于评论 “It’s fast, light, and simple to use.” 模型关注到了情感词 fast、light、simple, 并且对于连词 and 也有一定的关注, 同时忽略了其他的不相关词。图 7 中例句 “Our waiter was friendly and it is a shame that he didnt have a supportive staff to work with.” 中对于方面词 waiter 模型主要关注到情感词 friendly, 而对于方面词 staff 模型能够捕捉到长距离依赖的情感词 shame, 并且对于否定词 not 也有关注。

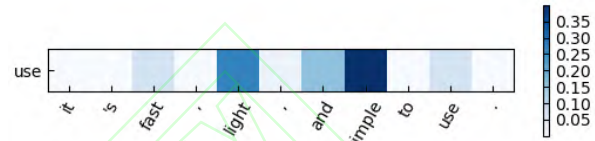


图 6 对评论 “It’s fast, light, and simple to use.” 的可视化结果

Fig. 6 Visualization result for the comment

“It’s fast, light, and simple to use.”

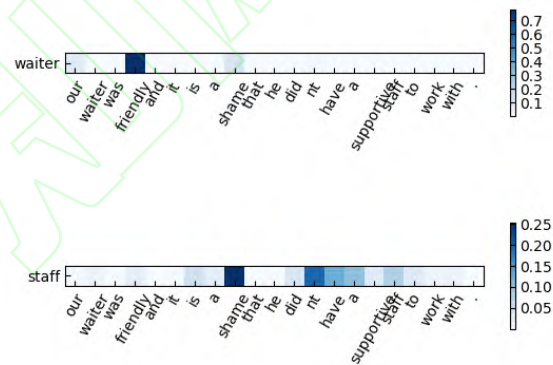


图 7 对例句 “Our waiter was friendly and it is a shame that he didnt have a supportive staff to work with.” 的可视化结果

Fig. 7 Visualization result for the example “Our waiter was friendly and it is a shame that he didnt have a supportive staff to work with.”

对于句子 “It’s fast, light, and simple to use.”, 权重如图 6 所示, 句子中的方面词为 use。可以看到模型对于方面词 use 有一定的关注, 而对于情感词 simple、light 有明确的关注, 甚至对较远的 fast 也有关注, 同时也忽略了无关的逗号, 捕捉到了词 and, 一定程度上说明模型对并列关系有关注。

对于句子 “Our waiter was friendly and it is a shame that he didnt have a supportive staff to work with.” 中的方面词为 waiter、staff, 对于方面词 waiter 的极性为积极的, 对于 staff 的极性为消极的。从图 7 的权重中能够看到对于方面词 waiter 关注到了情感词 friendly, 对于方面词 staff 关注到了长距离依赖的 shame 和否定词。

从上面的可视化例子中可以分析, 本文的模型能够明确的捕捉到和方面词相关的情感词而忽略不相关的标点符号和一些其他的词, 捕获到较长的依赖关系, 并且能够识别出否定关系, 因此自注意力机制的引入对于分类精度的提升是有效的。

## 3 结束语

本文引入自注意力机制对同一个文本进行不同的语义编码, 然后进行多层 hop 获取方面词和每个语义编码的关系, 最后对语义编码进行分类以确定方面词在句子中的极性。模



型用 SenEval 2014 Task4 数据集进行测试, 验证了模型能够捕捉到长距离依赖的情感词, 一定程度上识别否定、短语, 并和一些基础模型比较, 显示了自注意力机制在基于方面的情感分析分类精度提升有效性。

虽然模型能够捕捉语义信息, 但是对于隐含语义却不能捕捉到。模型应用情感词典融入了更多的情感信息, 另外可以进一步融入词性等额外信息来进一步提高模型的准确率, 此外对于模型中的自注意力编码语义向量的个数和 hop 的阶数是手动设置的, 不同的数据集可能会对对应于不同的值, 因此如何能够自动找出最优值可以进一步开展研究。

## 参考文献:

- [1] 江红. 情感分析研究综述 [J]. 智能计算机与应用, 2018, 8 (5): 103-105. (Jiang Hong. Research on sentiment analysis [J]. Intelligent Computer and Applications, 2018, 8 (5): 103-105.)
- [2] 唐晓波, 刘广超. 细粒度情感分析研究综述 [J]. 图书情报工作, 2017, 61 (05): 132-140. (Tang Xiaobo, Liu Guangchao. A review of fine-grained emotional analysis [J]. Library and Information Service, 2017, 61 (05): 132-140.)
- [3] 刘爽, 赵景秀, 杨红亚, 等. 文本情感分析综述 [J]. 软件导刊, 2018, 17 (6): 1-4, 21. (Liu Shuang, Zhao Jingxiu, Yang Hongya, et al. A survey of sentiment analysis [J]. Software Guide, 2018, 17 (6): 1-4, 21.)
- [4] Pontiki M, Galanis D, Pavlopoulos H, et al. SemEval-2014 Task 4: Aspect Based Sentiment Analysis [C]// Proc of the 8th International Workshop on Semantic Evaluation, SemEval@COLING. 2014: 27-35.
- [5] Luong T, Pham H, Manning C D. Effective Approaches to Attention-based Neural Machine Translation [C]// Proc of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP. 2015: 1412-1421.
- [6] Vaswani A, Shazeer N, Parmar N, et al. Attention is All you Need [C]// Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems. 2017: 6000-6010.
- [7] Paulus R, Xiong Caiming, Socher R. A Deep Reinforced Model for Abstractive Summarization [C]// the 6th International Conference on Learning Representations, ICLR. 2018.
- [8] Zhou Hanlin, Min Weifeng, Santos C N, et al. A Structured Self-Attentive Sentence Embedding [C]// the 5th International Conference on Learning Representations, ICLR. 2017.
- [9] Cheng Jianpeng, Dong Li, Lapata M. Long Short-Term Memory-Networks for Machine Reading [C]// Proc of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP. 2016: 551-561.
- [10] Shen Tao, Zhou Tianyi, Long Guodong, et al. DiSAN: Directional Self-Attention Network for RNN/CNN-Free Language Understanding [C]// Proc of the Thirty-Second {AAAI} Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th {AAAI} Symposium on Educational Advances in Artificial Intelligence (EAAI-18). New Orleans, 2018: 5446-5455.
- [11] Im J, Cho S. Distance-based Self-Attention Network for Natural Language Inference [J]. CoRR, 2017, abs/1712.02047.
- [12] Verga P, Strubell E, McCallum A. Simultaneously Self-Attending to All Mentions for Full-Abstract Biological Relation Extraction [C]// Proc of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT. 2018: 872-884.
- [13] Tan Zhixing, Wang Mingxuan, Xie Jun, et al. Deep Semantic Role Labeling With Self-Attention [C]// Proc of the Thirty-Second {AAAI} Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th {AAAI} Symposium on Educational Advances in Artificial Intelligence (EAAI-18). 2018: 4929-4936.
- [14] Shaw P, Uszkoreit J, Vaswani A. Self-Attention with Relative Position Representations [C]// Proc of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT. 2018: 464-468.
- [15] Palang H, Deng Li, Shen Yelong, et al. Deep sentence embedding using long short-term memory networks: analysis and application to information retrieval [J]. IEEE Transactions on Audio, Speech, and Language Processing, 2016, 24 (4): 694-707.
- [16] Wang Di, Nyberg E. A Long Short-Term Memory Model for Answer Sentence Selection in Question Answering [C]// Proc of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL. 2015: 707-712.
- [17] Poria S, Cambria E, Gelbukh A F, et al. Sentiment Data Flow Analysis by Means of Dynamic Linguistic Patterns [J]. IEEE Computational Intelligence Magazine, 2015, 10 (4): 26-36.
- [18] Tang Duyu, Qin Bing, Feng Xiaocheng, et al. Effective LSTMs for Target-Dependent Sentiment Classification [C]// COLING, the 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers. 2016: 3298-3307.
- [19] Wang Yequan, Huang Minlie, Zhu Xiaoyan, et al. Attention-based LSTM for Aspect-level Sentiment Classification [C]// Proc of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP. 2016: 606-615.
- [20] Thuy N T, Tojo S, Hanh T, et al. Effective Attention Networks for Aspect-level Sentiment Classification [C]// the 10th International Conference on Knowledge and Systems Engineering, KSE. 2018: 25-30.
- [21] Ma Dehong, Li Sujian, Zhang Xiaodong, et al. Interactive Attention Networks for Aspect-Level Sentiment Classification [C]// Proc of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI. 2017: 4068-4074.
- [22] Tang Duyu, Qin Bing, Liu Ting. Aspect Level Sentiment Classification with Deep Memory Network [C]// Proc of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP. 2016: 214-224.
- [23] Zhou Ke, Zeng Jiangfeng, Liu Yu, et al. Deep sentiment hashing for text retrieval in social CIoT [J]. Future Generation Computer Systems, 2018, 86: 362-371.
- [24] Chen P, Sun Z Q, Bing L D, et al. Recurrent Attention Network on Memory for Aspect Sentiment Analysis [C]// Proc of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP. 2017: 452-461.
- [25] Ma Yukun, Peng Haiyun, Cambria E. Targeted Aspect-Based Sentiment Analysis via Embedding Commonsense Knowledge into an Attentive LSTM [C]// Proc of the Thirty-Second {AAAI} Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th {AAAI} Symposium on Educational Advances in Artificial Intelligence (EAAI-18). 2018: 5876-5883.
- [26] Yang Chao, Zhang Hefeng, Jiang Bin, et al. Aspect-based sentiment analysis with alternating coattention networks [J]. Information Processing and Management, 2019, 56 (3): 463-478.
- [27] 武婷, 曹春萍. 融合位置权重的基于注意力交叉注意力的长短期记忆方面情感分析模型 [J/OL]. 计算机应用: 1-6 [2019-08-23]. <http://kns.cnki.net/kcms/detail/51.1307.TP.20190417.1750.002.html>. (Wu Ting, Cao Chunping. Aspect level sentiment classification model

- with location weight and long-short term memory based on attention-over-attention [J/OL]. *Journal of Computer Applications*: 1-6 [2019-08-23]. <http://kns.cnki.net/kcms/detail/51.1307.TP.20190417.1750.002.html>.)
- [28] Ma Xiao, Zeng Jiangfeng, Peng Limei, *et al.* Modeling multi-aspects within one opinionated sentence simultaneously for aspect-level sentiment analysis [J]. *Future Generation Computer Systems*, 2019, 93: 304-311.
- [29] Zeng Jiangfeng, Ma Xiao, Zhou Ke. Enhancing Attention-Based LSTM With Position Context for Aspect-Level Sentiment Classification [J]. *IEEE Access*, 2019, 7: 20462-20471.
- [30] Ma Ruixin, Wang Kai, Qiu Tie, *et al.* Feature-based Compositing Memory Networks for Aspect-based Sentiment Classification in Social Internet of Things [J]. *Future Generation Computer Systems*, 2019, 92: 879-888.
- [31] 王行甫, 王磊, 苗付友, 等. 结合词性、位置和单词情感的内存网络的方面情感分析 [J]. *小型微型计算机系统*, 2019, 40 (2): 383-389. (Wang Xingfu, Lei, Miao Fuyou, *etal.* Aspect level sentiment analysis with memory network with POS, position and polarity of word [J]. *Journal of Chinese Computer Systems*, 2019, 40 (2): 383-389.)
- [32] Wang Xinyi, Xu Guangluan, Zhang Jingyuan, *et al.* Syntax-Directed Hybrid Attention Network for Aspect-Level Sentiment Analysis [J]. *IEEE Access*, 2019, 7: 5014-5025.
- [33] Pennington J, Socher R, Manning C D, *et al.* Glove: Global Vectors for Word Representation [C]// *Proc of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*. 2014: 1532-1543.
- [34] Sukhbaatar S, Szlam A, Weston J, *et al.* End-to-end memory networks [J]. *neural information processing systems*, 2015, 2440-2448.