

基于局部上下文关键词的方面级情感分析^①

曾碧卿, 黄梓芃

(华南师范大学 软件学院, 佛山 528225)

通信作者: 曾碧卿, E-mail: zengbiqing@scnu.edu.cn



摘要: 针对目前大多数方面级情感分析方法存在着没有重点关注局部上下文中关键词特征的问题, 本文提出了一种基于局部上下文关键词特征提取及增强的方面级情感分析模型 LCPM (local context pos mask). 首先提出了局部上下文词性掩码机制, 提取方面词周围重要词的特征, 减少噪声词的干扰. 其次对损失函数进行修改, 让模型重点关注与方面词有关的局部上下文关键词特征, 提升模型情感分类的表现. 最后设计了一种门控机制, 模型可以动态学习权重系数, 给局部上下文关键词特征和全局上下文特征分配不同的权重系数. 在 4 个公开数据集上的实验结果表明, 与现有的方面级情感分析模型相比, 准确率和 *MF1* 值都有提高, 验证了局部上下文关键词提取及增强的有效性, 在方面级情感分析任务上有较大的应用价值.

关键词: 方面级情感分析; 关键词特征; 词性掩码; 损失函数; 门控机制

引用格式: 曾碧卿, 黄梓芃. 基于局部上下文关键词的方面级情感分析. 计算机系统应用, 2023, 32(6): 1-11. <http://www.c-s-a.org.cn/1003-3254/9116.html>

Aspect Level Sentiment Analysis Based on Local Context Keyword

ZENG Bi-Qing, HUANG Zi-Peng

(School of Software, South China Normal University, Foshan 528225, China)

Abstract: Most aspect level sentiment analysis methods do not focus on keyword features in the local context. Therefore, this study proposes an aspect level sentiment analysis model LCPM (local context pos mask) based on local context keyword feature extraction and enhancement. First, a local context part of the speech mask mechanism is proposed to extract the important words features around aspect words and reduce the interference of noise words. Second, the loss function is modified, so that the model focuses on the local context keyword features related to aspect words and improves the performance of the model's sentimental classification. Finally, a gating mechanism is designed. The model can dynamically learn the weight coefficients and assign different weight coefficients to local context keyword features and global context features. The experiments on four open datasets show that, compared with existing aspect level sentiment analysis models, the proposed model has higher accuracy and *MF1* value, which verifies the effectiveness of local context keyword extraction and enhancement and is of application significance in aspect level sentiment analysis tasks.

Key words: aspect level sentiment analysis; keyword features; part of speech mask; loss function; gating mechanism

1 引言

随着网络发展和社交媒体软件的普及, 产生大量

的网络评论数据, 对这些评论数据进行情感分析有助于理解用户的需求. 目前情感分析在计算机领域有着

^① 基金项目: 国家自然科学基金 (62076103); 广东省普通高校人工智能重点领域专项 (2019KZDZX1033); 广东省信息物理融合系统重点实验室 (2020B1212060069); 广东省基础与应用基础研究基金 (2021A1515011171); 广州市基础研究计划基础与应用基础研究项目 (202102080282)

收稿时间: 2022-11-22; 修改时间: 2022-12-23; 采用时间: 2023-01-06; csa 在线出版时间: 2023-04-14

CNKI 网络首发时间: 2023-04-18

广泛的应用,方面级情感分析是自然语言处理中一种细粒度的任务,又称细粒度情感分析,主要是确定句子中不同方面词对应的情感极性。情感极性一般分为积极、中立、消极3类。例如,“The screen resolution attracts me but its battery is miserable.”这句话从“screen resolution”和“battery”两个方面进行评价。其中,“screen resolution”的情感极性是积极的,“battery”的情感极性是消极的。显然,粗粒度的情感分析无法充分分析句子的情感极性。近年来,方面级情感分析在自然语言处理领域中受到广泛的关注。先前大多数研究主要是使用循环神经网络(recurrent neural network, RNN)、长短时记忆网络(long short-term memory, LSTM)、门控循环单元(gated recurrent unit, GRU)等序列模型进行情感分析。随着注意力机制的兴起,上述模型与注意力机制结合在情感分析领域上取得了不错的效果。Zhang等^[1]把多层注意力模型与记忆网络相结合,增强模型对复杂情况的表达能力。Ma等^[2]提出一种基于长短期记忆网络和注意力机制结合的网络(interactive attention networks, IAN)。IAN分别对方面词和上下文进行建模,并加入注意力机制,提取方面词在句子中的情感信息。随后出现了图神经网络(graph neural network, GNN),有研究尝试提取句法依存特征,输入到图神经网络中,提取深层次的情感特征。GNN是一类基于深度学习的处理图域信息的方法,图神经网络包括图注意力网络(graph attention network, GAT),图卷积网络(graph convolutional network, GCN)等。Sun等^[3]利用双向长短时记忆网络提取句子特征,并在句子的依存树上增加图卷积网络(GCN)来增强词嵌入,获取情感特征。Wang等^[4]重新构造和修剪依赖解析树,设计一种面向方面词的句法依赖树结构,并使用关系图注意力网络(relational graph attention, RGAT)来编码句法依赖特征。Bai等^[5]认为句法依赖关系可以作为一种有用的外部知识,根据源句子生成句法解析树,使用图注意力网络(GAT)提取句法依存特征,并设计一种新的注意力函数,使用句法依存标签丰富单词表示,然后将Bi-LSTM提取的上下文特征与RGAT提取的上下文特征融合,形成最终的特征。Zhou等^[6]为解决句法解析树提取的方面词特征不全的问题,提出基于图注意力网络和BERT的动态多通道融合模型。使用两个动态融合模块自适应地调整语义特征,并为了捕获准确的句法特征,提出了注意力集成模块,整合GAT提取的多层特征,充分考虑句法信息。

上述方法在方面级情感分类任务上已经取得了较好的效果,但仍存在着部分缺陷及不足之处。

(1) 根据全局上下文提取句法特征信息,没有关注方面词与局部上下文之间的关系,没有重点考虑局部关键词特征。因为全局上下文中存在一些无用的情感词,会干扰最终的情感极性判断。相比于全局特征,局部上下文关键词蕴含更丰富的情感信息。

(2) 没有利用损失函数进一步加强关键词对情感极性的影响。因为在局部上下文中,不同词性的词对方面词影响完全不同。

(3) 没有对局部上下文关键词特征和全局上下文特征动态分配不同的权重。适当融合全局上下文特征,可防止忽视一些重要的全局信息。

为了解决这些问题,本文提出了局部上下文词性掩码网络,并对损失函数进行修改,保证模型训练的同时兼顾提取局部上下文关键词特征的能力,让模型重点关注关键词的特征。然后设计了一种门控机制,让局部上下文关键词特征与全局上下文特征进行加权融合,增强模型总体的信息容纳能力和语义表达能力。本文提出的LCPM模型可以更好关注与方面词相关的关键词特征,准确分析方面词对应的情感极性。

本文的主要贡献如下。

(1) 提出LCPM网络模型,模型重点关注局部上下文关键词特征,提取与方面词相关的信息,避免全局上下文冗余信息的干扰。

(2) 对损失函数进行修改,让模型注重与方面词有关的局部上下文关键词特征,捕获与方面词相关的上下文信息,提升模型情感分类的表现。

(3) 设计一种门控机制,模型可以动态学习权重系数,给局部上下文关键词特征和全局上下文特征分配不同的权重,提取丰富的情感信息。

(4) 在4个公开数据集的实验结果综合证明所提出的LCPM模型在方面级情感分类任务上均优于先进的基线方法。

2 相关工作

情感分析早期使用的是传统的机器学习算法,如朴素贝叶斯和支持向量机^[7]等,支持向量机与其他传统的机器学习算法相比,在情感分析任务上性能较优。但是传统机器学习算法分类效果很大程度上取决于所选取特征的质量,并且要设计大量的特征才能提高模

型分类效果, 耗费大量资源, 并且这类方法需要人工构建情感词典^[8]. 基于情感字典进行情感分析很大程度上取决于情感字典的质量, 且语义表达具有多样性 (如反问句和倒装句等), 该方法难以确定方面词对应的情感极性, 现在已经比较少用了.

近年来, 随着深度学习的发展, 出现了大量的神经网络, 卷积神经网络^[9]、循环神经网络^[10]、长短期记忆网络^[11]、记忆网络^[12]、图卷积网络^[13]等广泛应用在自然语言处理任务上. Tang 等^[11]使用两个 LSTM 建模方面词与上下文的关系, 充分提取方面词对应的情感极性. 在方面级情感分析任务中, 神经网络和注意力机制相结合的做法取得了不错的情感分类效果. Huang 等^[14]基于长短期记忆网络, 对情绪分类问题进行了研究, 提出了注意力交互模块 (attention-over-attention, AOA), 提取深层次的情感特征. Fan 等^[15]考虑了方面词和上下文的注意力, 提出了一种 loss 计算, 建模方面词和上下文的关系. Song 等^[16]使用注意力机制描绘方面词和上下文的语义信息, 并在损失函数中添加标签平滑正则化项, 促使模型学习中性标签. 然而, 该模型没有考虑句法信息和单词间的依赖关系, 并存在着把句法不相关的上下文单词作为判断方面词情感极性线索的问题. 随后, 有研究表明利用语法依赖树信息构建邻接矩阵, 再使用如图注意力网络和图卷积网络等结

合邻接矩阵, 建模方面词与上下文的关系, 能充分利用句法信息, 让模型重点关注与方面词有关的上下文信息. Bai 等^[5]把句法解析树作为邻接矩阵, 输入到图注意力网络, 并与 Bi-LSTM 提取的特征进行融合. Zhou 等^[6]充分利用 GAT 和 BERT 获取语义特征, 并利用多头注意力机制连接两个通道信息. 为了捕获更准确的句法信息, 使用注意力集成模块整合 GAT 提取的多层特征, 最后利用动态融合模块自适应地调整句法信息, 使模型具备提取丰富的语法知识的能力. 然而该方法没有考虑局部上下文关键词特征, 没有加强文本中重要词汇对情感分析的影响, 其任务性能有待进一步提升.

基于以上研究, 本文提出 LCPM 网络模型, 并修改了损失函数, 让模型重点关注局部上下文关键词特征. 最后让模型动态学习权重系数, 给局部上下文关键词特征和全局上下文特征分配不同的权重, 提取最终的特征, 使局部上下文关键词特征和全局上下文特征受益于彼此, 提升情感分类的性能.

3 LCPM 模型

本文提出的 LCPM 模型主要由词嵌入层、特征提取层、特征融合层和输出层所组成, 以及由局部上下文关键词特征提取模块、门控融合模块、损失函数辅助模块这 3 大模块组成, 模型框架如图 1 所示.

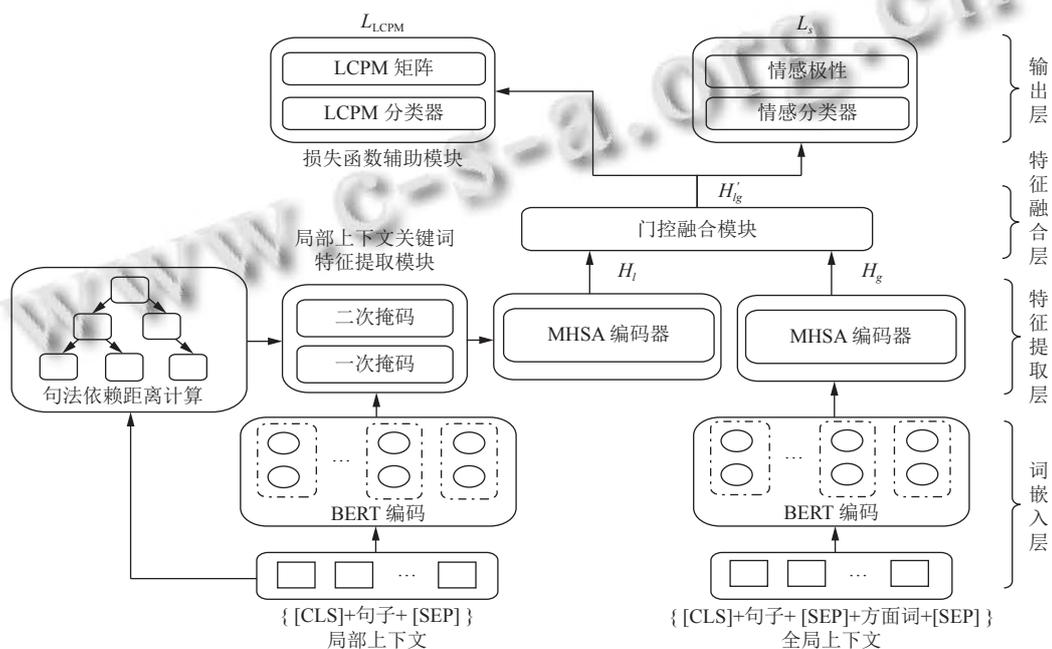


图 1 LCPM 模型图

3.1 任务描述

方面级情感分类任务可以视为是一个三分类任务. 给定一个含 N 个词的文本序列 $W^c = \{W_1^c, W_2^c, \dots, W_N^c\}$ 和 K 个方面词 $W^a = \{W^{a_1}, W^{a_2}, \dots, W^{a_k}\}$, 每个方面词 $W^{a_i} = \{W_1^{a_i}, W_2^{a_i}, \dots, W_{M_i}^{a_i}\}$ 均出现在给定的序列中, 且 $1 \leq M_i \leq N$. 方面词可以由一个或多个词组成. 本文任务是预测给定文本序列中每个方面词对应的情感极性 $y_i \in \{+1, 0, -1\}$. 其中, +1 表示积极情感, 0 表示中性情感, -1 表示消极情感.

3.2 词嵌入层

网络评论文本通常篇幅较长, 此时根据上下文动态计算 token 是相当重要的. 自然语言处理任务先前常用 Word2Vec 作为词嵌入, Word2Vec 对于同个 token 只有一种表示, 无法解决一词多义的问题, 并且缺乏对位置的建模, 考虑到 DMF-GAT-BERT 模型^[6]也是采用 BERT 作为词嵌入, 为了公平对比, 本文使用 BERT 预训练模型作为词嵌入, 构建依赖于特定方面词的文本表示. BERT 是基于 Transformer 的多层双向编码器结构, 可以同时使用前后两个方向的信息, 对输入文本数据进行整体感知, 面对不同文本中相同单词时, 可生成不同的词向量. BERT 的输入向量, 是由词向量 (token embeddings)、段向量 (segment embeddings)、位置向量 (position embeddings) 所组成. BERT 模型利用

大规模无标注语料进行训练, 能够挖掘文本中丰富的语义信息. 为了便于模型训练, 将输入句子和方面词分别转换为 “[CLS]+句子+[SEP]” 和 “[CLS]+句子+[SEP]+方面词+[SEP]” 的形式, 作为 BERT 模型的输入, 生成局部上下文向量 $W_{BERT}^l = \{W_1^l, W_2^l, \dots, W_{N_l}^l\}$ 和全局上下文向量 $W_{BERT}^g = \{W_1^g, W_2^g, \dots, W_{N_g}^g\}$, $W_{BERT}^l \in R^{d_h \times N_l}$, $W_{BERT}^g \in R^{d_h \times N_g}$, N_l 和 N_g 是局部上下文和全局上下文的长度, d_h 是词嵌入的维度.

3.3 特征提取层

3.3.1 句法依赖距离计算

本文通过 Spacy 工具包生成输入文本的句法依赖解析树. 以 “but the staff was so horrible to us.” 为例, 其句法依赖解析树如图 2 所示, 其中 “staff” 表示方面词, 箭头表示两个词存在句法依赖关系, 弧线中间的标签表示句法依赖关系的类型. 本文使用句法依赖解析树距离确定局部上下文区域, 重点关注名词、形容词、动词, 利用句法依赖关系以及词性充分提取与方面词有关的特征. 句法依赖解析树中每个词到方面词的路径称为句法依赖路径, 每个词到方面词可能存在多条路径, 把最短路径上边的个数定义为距离. 如 “to” 到 “staff” 路径为 “to” → “horrible” → “was” → “staff”, 该路径上总共有 3 条边, 所以 “to” 到 “staff” 的距离为 3. 如果方面词由多个词组成, 则距离定义为每个词到方面词子词距离的平均值.

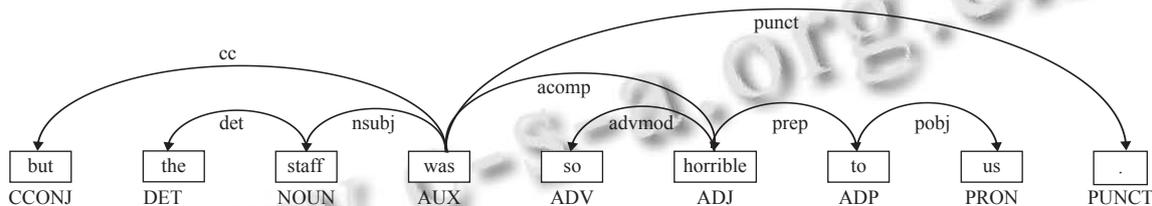


图 2 句法依赖解析树

3.3.2 局部上下文关键词特征提取模块

在语言学中, 词性是指以词的特点划分词类, 以语法特征为主要依据, 兼顾单词的含义, 对词进行划分的结果. 本文使用 NLTK 工具包对输入文本中的每个词进行词性标注. 本文提出的局部上下文关键词掩码, 主要计算每个词相对方面词在句法依赖解析树上的距离, 当距离小于等于给定阈值时认为是局部上下文, 对局部上下文中的词进行保留, 即一次掩码. 然后聚焦局部上下文内的形容词、名词、动词, 即二次掩码, 防止噪

声词影响情感极性判断. 这是因为在局部上下文中, 由于不同词性的词对方面词的情感极性影响不同, 如果保留所有词性的词, 会引入较多的噪音信息, 影响模型训练. 本文提出的局部上下文关键词掩码会掩盖输入序列中与方面词不相关的上下文词, 抛弃语义较少的词的特征, 重点关注局部上下文关键词信息, 减少冗余信息的影响.

掩码过程如图 3 所示, 式 (1)–式 (3) 是生成掩码矩阵, 式 (4) 是 W_{BERT}^l 经过掩码后的特征.

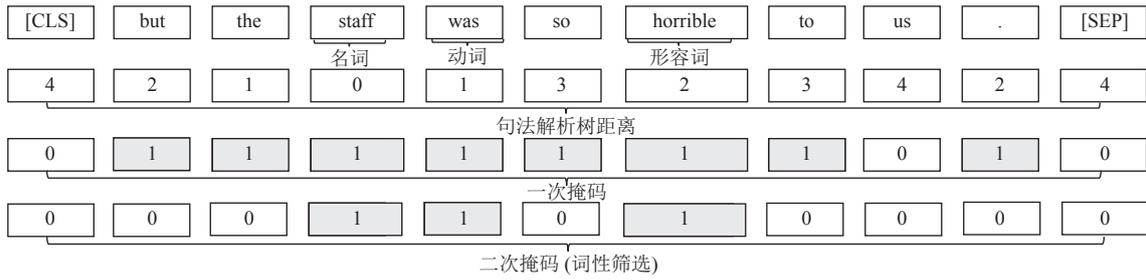


图3 局部上下文关键词掩码

$$P = \{\text{ADJ, NOUN, VERB}\} \quad (1)$$

$$V_i = \begin{cases} E, d_i \leq \partial \cap T_i \in P \\ O, d_i > \partial \cup \{d_i \leq \partial \cap T_i \notin P\} \end{cases} \quad (2)$$

$$M = [V_1, V_2, \dots, V_{N_i}] \quad (3)$$

$$W_{\text{LCPM}} = W_{\text{BERT}}^l \odot M \quad (4)$$

其中, d_i 表示每个词相对方面词在句法依赖解析树上的距离, $V_i \in \mathbb{R}^{d_h \times 1}$, $M \in \mathbb{R}^{d_h \times N_i}$, $O \in \mathbb{R}^{d_h \times 1}$, $E \in \mathbb{R}^{d_h \times 1}$ 代表全 0 和全 1 向量, $W_{\text{LCPM}} \in \mathbb{R}^{d_h \times N_i}$, d_h 是隐藏层的维度。

如图 3, 如果 ∂ 设置为 3, 当 $d_i \leq 3$ 时, 认为是局部上下文区域, 在局部上下文区域中重点关注名词、形容词、动词, 注重局部上下文关键词特征。

3.3.3 MHSA 编码器

提取输入向量的语义特征通过注意力机制实现的, 注意力机制是从大量信息中有筛选出重要信息, 并聚焦到这些信息上, 忽略不重要的信息。多头注意力机制是集成多个注意力函数, 用于计算每个上下单词的注意力分数, 提取语义特征。自注意力函数使用的是带缩放的点积注意力 (scaled dot-product attention, SDPA), 因为计算效率和安全性较高。为了缓解上下文中距离较远的词的负面影响, 主要是根据 Query (Q) 与 Key (K) 之间的匹配度来对 Value (V) 进行加权, 且 Query、Key、Value 都来自于输入, 如式 (5) 所示:

$$\text{SDPA} = \text{Softmax}\left(\frac{Q \cdot K^T}{\sqrt{d_k}}\right) \cdot V \quad (5)$$

本模块使用的是多头自注意力机制, 即把每个注意力头学习到的注意力表示拼接起来, 再乘以权重矩阵 W_h 。多头自注意力机制主要是由多个 SDPA 所组成, 可并行化计算多次的注意力, 同时关注多个子空间上的特征, 式 (6)–式 (10) 是多头自注意力的计算流程。

$$Q = X \cdot W_q \quad (6)$$

$$K = X \cdot W_k \quad (7)$$

$$V = X \cdot W_v \quad (8)$$

$$H_i = \text{Attention}(Q, K, V) \quad (9)$$

$$\text{MHSA}(X) = \tanh(\text{Concat}(H_1, H_2, \dots, H_h) \cdot W_h) \quad (10)$$

其中, X 是输入的 embedding, $W_q \in \mathbb{R}^{d_h \times d_q}$, $W_k \in \mathbb{R}^{d_h \times d_k}$, $W_v \in \mathbb{R}^{d_h \times d_v}$, $d_q = d_k = d_v = d_h/h$, d_h 是隐藏层的维度, h 是注意力头的个数, W_q, W_k, W_v 是权重矩阵, 这些权重矩阵是可训练的。 H_i 表示每个注意力头学习到的注意力表示。

本文将 \tanh 激活函数应用于 MHSA 编码器中, 可以更充分提取局部上下文关键词特征 H_i 。获取 H_i 如式 (11) 所示:

$$H_i = \text{MHSA}(W_{\text{LCPM}}) \quad (11)$$

经过词性筛选和 MHSA 编码提取的局部上下文关键词特征, 可聚焦局部范围内的形容词、名词、动词。然而, 有时会忽视一些重要的全局上下文特征。因此, 本文使用 BERT 模型提取全局特征, 再使用 MHSA 进行特征增强, 得到全局上下文特征 H_g , 如式 (12) 所示。 H_g 对局部上下文关键词特征进行补充, 增强特征质量, 提升方面级情感分析的性能。

$$H_g = \text{MHSA}(W_{\text{BERT}}^g) \quad (12)$$

3.4 特征融合层

LCPM 模型的特征融合层主要是基于门控特征融合机制实现的, 如图 4 所示。 Gate1 主要是拼接局部上下文关键词特征 H_i 及全局上下文特征 H_g , 输入到 Sigmoid 函数中得到权重系数, 权重系数值在 0, 1 之间, 如式 (13) 和式 (14) 所示。池化函数提取最终的局部上下文关键词池化特征 H_i^{pool} 和全局上下文池化特征 H_g^{pool} , 如式 (15) 和式 (16) 所示。 Gate2 是给 H_i^{pool} 分配 $1 - \phi$ 权重系数, 得到特征 H'_i , 如式 (17) 所示。 Gate3 是给 H_g^{pool} 分配 ϕ 权重系数, 得到特征 H'_g , 如式 (18) 所示。 Gate4 是融合 H'_i 和 H'_g 的特征, 得到完整的情感特征 H'_{ig} , 如式 (19)

所示.

$$H_{lg} = \text{Concat}(H_l, H_g) \quad (13)$$

$$\varphi = \text{Sigmoid}(W_{lg}H_{lg} + b_{lg}) \quad (14)$$

$$H_l^{\text{pool}} = \text{pool}(H_l) \quad (15)$$

$$H_g^{\text{pool}} = \text{pool}(H_g) \quad (16)$$

$$H'_l = (1 - \varphi) \odot H_l^{\text{pool}} \quad (17)$$

$$H'_g = \varphi \odot H_g^{\text{pool}} \quad (18)$$

$$H'_{lg} = H'_l + H'_g \quad (19)$$

其中, *Concat* 是特征拼接函数, H_{lg} 是拼接后的特征, W_{lg} 是权重矩阵, φ 是权重系数, H_l^{pool} 是局部上下文关键词池化特征, H_g^{pool} 是全局上下文池化特征, H'_l 和 H'_g 是分配权重系数后的特征, H'_{lg} 是完整的融合特征.

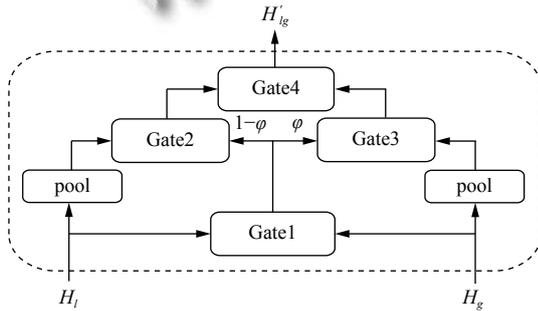


图4 门控融合模块

3.5 输出层

3.5.1 LCPM 的损失函数

本文把预测局部上下文关键词掩码矩阵作为辅助信息,使模型充分利用局部上下文关键词特征,增强特征提取能力.预测 LCPM 矩阵值可以视作为分类任务. LCPM 损失函数定义如式 (20) 所示.

$$L_{\text{LCPM}} = - \sum_1^{c'} \sum_1^l \hat{f} \log_2 p \quad (20)$$

其中, $c' = 2$, c' 是 LCPM 标签类型的数量, $\hat{f} \in \{0, 1\}$, 正类为 1, 负类为 0, p 是预测为正类的概率, l 是输入文本的长度.

3.5.2 分类器

本文使用全连接网络作为分类器,将融合特征 H'_{lg} 作为输入,对融合特征进行线性变化,得到 H_{lg}^Y 和

H_{lg}^T , 再使用 *Softmax* 函数计算出每个情感类别的概率,预测情感值如式 (23) 所示,预测 LCPM 标签值如式 (24) 所示.

$$H_{lg}^Y = W_{lg}^Y H'_{lg} + b_{lg}^Y \quad (21)$$

$$H_{lg}^T = W_{lg}^T H'_{lg} + b_{lg}^T \quad (22)$$

$$Y = \text{Softmax}(H_{lg}^Y) = \frac{\exp(H_{lg}^Y)}{\sum_{k=1}^C \exp(H_{lg}^Y)} \quad (23)$$

$$T = \text{Softmax}(H_{lg}^T) = \frac{\exp(H_{lg}^T)}{\sum_{k=1}^{C'} \exp(H_{lg}^T)} \quad (24)$$

其中, C 是情感类别的数量, C' 是 LCPM 标签类别的数量, Y 是预测的情感值, T 是预测的 LCPM 标签值, W_{lg}^Y 和 W_{lg}^T 是权重矩阵, b_{lg}^Y 和 b_{lg}^T 是偏置矩阵.

模型采用交叉熵作为损失函数,并引入 L2 正则化防止模型过拟合,如式 (26) 所示.

$$L_s = - \sum_1^C \hat{y} \log_2 p' \quad (25)$$

$$L = (1 - \epsilon)L_s + \epsilon L_{\text{LCPM}} + \lambda \sum_{\theta \in \Theta} \theta^2 \quad (26)$$

其中, L_s 是预测情感极性的损失函数值, C 是情感类别的数量, p' 是属于某个情感类别的概率,属于某个类别时 $\hat{y} = 1$, 否则 $\hat{y} = 0$, ϵ 是损失函数权重系数, $\lambda > 0$, Θ 是参数集合.

4 实验

4.1 数据集与评价指标

为了验证 LCPM 模型的有效性,从 SemEval 2014 中选取 Restaurant、Laptop 数据集,以及选取 Twitter、MAMS 数据集进行实验.这些数据集集中的每个样本都包含一个或多个方面词,每个方面词都标注有对应的情感极性,情感极性分为积极、消极、中性 3 类.根据实验结果, LCPM 模型在这些数据集上性能有显著提升,准确率和 MF1 值提升明显.这表明 LCPM 模型不只适用于特定的数据集,而且在大多数数据集上都表现良好.这是由语言特性所决定的,人们在社交软件上评论时,展示方面词对应情感极性的词通常都是在方面词附近的,并且这些词大多是名词、形容词和动词.大多数模型都选取以上数据集进行实验,这些数据集

具备一定的权威性. 数据集具体统计信息如表 1 所示, Train 表示训练集, Test 表示测试集.

表 1 实验数据统计

数据集	积极		消极		中立	
	Train	Test	Train	Test	Train	Test
Restaurant	2164	728	807	196	637	196
Laptop	994	341	870	128	464	169
Twitter	1561	173	1560	173	3127	346
MAMS	3380	400	2764	329	5042	607

本实验采用准确率 (Acc) 和宏平均 $F1$ 值 ($MF1$) 作为评估指标, Acc 表示正确分类的正样本数和负样本数之和占总样本数的比例, 比例越高代表模型分类的性能越好. $MF1$ 值是各个类别 $F1$ 值的平均值, 能体现模型在不平衡数据集上的平均性能, 式 (27) 是计算准确率, 式 (28)–式 (31) 是 $MF1$ 值的计算流程.

$$Acc = \frac{TP+TN}{TP+FP+FN+TN} = \frac{TP+TN}{N} \quad (27)$$

$$P = \frac{TP}{TP+FP} \quad (28)$$

$$R = \frac{TP}{TP+FN} \quad (29)$$

$$F1 = \frac{2 \times P \times R}{P+R} \quad (30)$$

$$MF1 = \frac{1}{C} \sum_{i=1}^C F1_i \quad (31)$$

其中, TP 是将正类预测为正类的样本数, FN 是将正类预测为负类的样本数, FP 是将负类预测为正类的样本数, TN 是将负类预测为负类的样本数. P 和 R 分别是精准率和召回率, C 为情感类别数.

4.2 参数设置与实验环境

LCPM 模型的超参数设置如表 2 所示, 本文选用 BERT 预训练的词向量作为词嵌入, 词嵌入维度为 768. 训练过程使用 Adam 进行参数优化. 为防止过拟合, 设置 dropout 为 0.5, L2 正则化值为 5×10^{-4} . 实验环境如表 3 所示.

表 2 LCPM 模型超参数设置

参数	值	参数	值
词嵌入维度	768	L2正则化值	5×10^{-4}
hidden size	768	学习率	2×10^{-5}
batch size	16	损失函数权重系数	0.3
dropout	0.5	局部上下文阈值	3
epoch num	10	最大句子长度	100

表 3 实验环境

实验环境	环境配置
操作系统	Ubuntu Server 18.04.5 LTS
显卡	NVIDIA Tesla V100
显存	32.0 GB
开发工具	PyCharm
CUDA版本	11.2
编程语言	Python 3.7.13
深度学习框架	PyTorch 1.10.1

4.3 对比实验

4.3.1 对比模型

为了验证 LCPM 模型的性能, 本文将该模型和一系列最先进的模型进行对比.

PhraseRNN^[17]: 通过添加两个短语组成函数, 并使用依赖树和成分树作为输入, 提取深层情感特征.

RAM^[18]: 在 LSTM 网络基础上使用多层注意力来捕获距离较远的情绪特征, 增强模型的表达能力.

SynAttn^[19]: 根据单词间的依存关系, 构建图神经网络, 再与 LSTM 网络结合, 捕获方面词相关的信息.

MGAN^[15]: 考虑方面词和上下文的注意力, 并提出了一种 loss 计算, 将粗粒度特征与细粒度特征结合, 充分考虑方面词和上下文的关系.

CDT^[3]: 提出一种依存树与卷积结合的模型, 利用双向长短期记忆网络来提取句子特征, 并利用依存树和图卷积网络结合来增强词嵌入.

ASGCN^[1]: 提出了一种情感分类框架, 在依存树基础上建立图卷积网络, 充分利用句法信息和单词依存关系, 并额外使用注意力机制获取最终的特征表示.

BERT-PT^[20]: 提出了后训练方法, 提高 BERT 模型微调的效果, 并应用到方面级情感分析任务上.

TD-GAT-BERT^[21]: 根据上下文单词间的依存关系, 建立图注意力网络, 再把图注意力网络和 LSTM 网络相结合, 捕获与方面词有关的特征.

AEN-BERT^[16]: 使用注意力机制建立上下文和方面词之间的关系, 提出了标签不可靠性问题且引入了标签平滑正则化, 并将 BERT 模型应用到方面级情感分析任务上.

FDDN-BERT^[22]: 将提取情感特征细化为初步蒸馏和精细蒸馏两大过程, 过滤和方面词不相关的冗余特征.

RGAT-BERT^[5]: 把句法解析树作为邻接矩阵, 输入到图注意力网络中, 提取句法依存特征, 并把该特征与双向长短期记忆网络提取的特征进行融合, 使模型

具备提取丰富的语法知识的能力。

DMF-GAT-BERT^[6]: 设计双层动态融合模块融合3个通道的语义特征, 并利用注意力集成模块整合GAT提取的多层特征, 提高情感极性识别准确率。

4.3.2 实验结果

表4为LCPM模型在常见4个数据集上的对比实验结果, 本文提出的模型准确率*Acc*和*MF1*值均优于基线模型, LCPM模型在Restaurant、Laptop、Twitter、MAMS数据集上准确率分别提升了1.22%、1.28%、0.54%、0.3%, *MF1*值分别提升了1.9%、1.19%、0.54%、0.33%。这是因为模型充分考虑了局部上下文

关键词的特征, 并且本文设计的损失函数辅助模块有助于关键词特征提取, 充分考虑了对情感极性判断贡献较大的关键词的特征。其中, LCPM模型在Restaurant数据集上提升了1.22%的*Acc*和1.9%的*MF1*值, 而在Twitter和MAMS数据集上提升不大, 这是因为这两个数据集文本的语法和结构相对混乱, 导致LCPM模型无法准确提取局部上下文关键词的特征, 可看出关键词特征的重要性。此外可以看出, 局部上下文关键词特征与全局上下文特征进行加权融合, 能显著增强模型总体的信息容纳能力和语义表达能力, 准确分析方面词对应的情感极性。

表4 对比实验结果图(%)

模型	Restaurant		Laptop		Twitter		MAMS	
	<i>Acc</i>	<i>MF1</i>	<i>Acc</i>	<i>MF1</i>	<i>Acc</i>	<i>MF1</i>	<i>Acc</i>	<i>MF1</i>
PhraseRNN	66.20	59.32	—	—	—	—	—	—
RAM	80.23	70.80	74.49	71.35	69.36	67.30	—	—
SynAttn	80.45	71.26	72.57	69.13	—	—	—	—
MGAN	81.25	71.94	75.39	72.47	72.54	70.81	—	—
CDT	82.30	74.02	77.19	72.99	74.66	73.66	80.70	79.79
ASGCN	80.77	72.02	75.55	71.05	72.15	70.40	—	—
BERT-PT	84.95	76.96	78.07	75.08	—	—	—	—
TD-GAT-BERT	83.00	—	80.10	—	—	—	—	—
AEN-BERT	83.12	73.76	79.93	76.31	74.71	73.13	—	—
FDDN-BERT	84.53	75.63	79.26	75.77	75.30	73.21	—	—
RGAT-BERT*	85.88	79.81	79.21	75.39	76.13	75.38	83.41	82.77
DMF-GAT-BERT	86.10	80.17	80.38	77.20	76.22	75.10	83.86	83.19
LCPM-BERT	87.32	82.07	81.66	78.39	76.76	75.64	84.16	83.52

注: “*”表示相同情况下按照原论文方法复现的模型, 其他结果是从已发表论文中得到的, 最佳效果用粗体标出。

表5 消融实验结果图(%)

Ablations	Restaurant		Laptop	
	<i>Acc</i>	<i>MF1</i>	<i>Acc</i>	<i>MF1</i>
-局部上下文关键词特征提取模块	85.62	79.26	79.15	76.23
-损失函数辅助模块	86.43	80.57	79.62	76.58
-门控融合模块	86.16	80.26	79.31	76.17
LCPM-BERT	87.32	82.07	81.66	78.39

注: “-”表示将该模块从模型中去除, 其他部分和LCPM模型保持一致, 最佳效果用粗体标出。

4.4 消融实验

为了验证本文设计的3大模块的重要性, 本文设计了一系列消融实验, 在Restaurant和Laptop数据集上观察实验结果。单独去掉局部上下文关键词特征提取模块、损失函数辅助模块、门控融合模块, 做消融实验。从表5的实验结果表明, 本文设计的3大模块对模型性能的提升均有影响, 其中局部上下文关键词特

征提取模块对模型性能的影响最大, 这充分说明了学习局部上下文关键词特征是十分重要的, 因为局部上下文关键词蕴含着丰富的情感信息, 可以增强特征的语义表达能力。此外, 本文设计的损失函数辅助模块对模型性能也有一定的影响, 因为该模块可以有效增强模型提取关键词特征的能力。最后移除了门控融合模块, 模型的性能会降低, 这充分说明了局部上下文关键词特征与全局上下文特征进行加权融合对于模型的性能提升是十分重要的。3大模块相互补充, 可以显著提升模型总体的语义表达能力。

4.5 参数分析

4.5.1 损失函数权重系数

LCPM模型引入了一个额外的超参数 ϵ , ϵ 是损失函数权重系数。为了探讨不同的 ϵ 对模型性能的影响, 本文在Restaurant、Laptop数据集上开展了实验, 设置了不同的 ϵ 值, 以分析 ϵ 对模型性能的影响, ϵ 从0开

始逐渐增加至 0.9, 步长为 0.1. 如图 5、图 6 所示, 结果表明, 当 $\epsilon = 0.3$ 时模型性能最好.

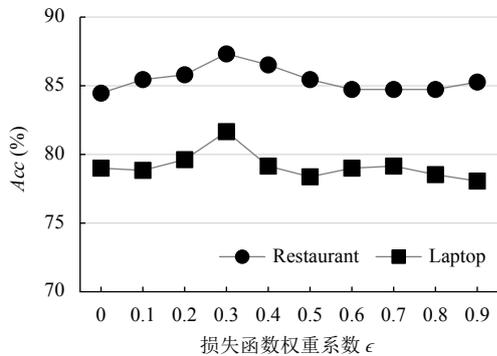


图 5 损失函数权重系数对准确率的影响

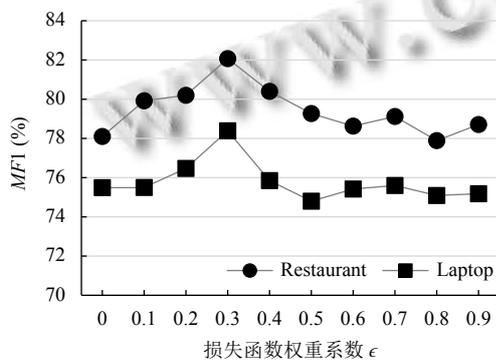


图 6 损失函数权重系数对 MF1 的影响

4.5.2 局部上下文阈值

局部上下文阈值 d_i 是影响局部关键词特征提取的重要因素. 为了找到合适的阈值, 本文通过实验讨论了 d_i 对 Restaurant、Laptop 数据集的影响. 从图 7、图 8 中可以看出, 当 d_i 为 1、2 时, 模型的准确率和 MF1 值比较低. 显然, 当 d_i 值较小时, 模型没有充分学习到局部上下文关键词的特征. 当 $d_i=3$ 时, 模型在以上数据集的准确率和 MF1 值达到最佳, 这是因为模型提取到与方面词相关的信息, 避免全局上下文冗余信息的干扰, 增强了特征表示. 而随后随着 d_i 的增加, 模型性能也较差, 呈现振荡下降趋势, 这说明当局部上下文阈值 d_i 一旦超出最佳值, 继续增加时, 会给模型带来额外的噪声信息, 影响模型的性能. 因此, 本文将局部上下文阈值 d_i 设置为 3.

4.5.3 词性筛选

为了进一步说明本文提出的词性筛选机制的有效性, 本文对比了不加词性筛选的效果, 即当句法依赖树

距离小于等于局部上下文阈值时保留该词的特征, 相当于提取全部的局部上下文特征, 模型的输出结果对比如表 6 所示. 显然, 模型增加词性筛选后, 在 Restaurant、Laptop 数据集上准确率分别提升了 1.34%、2.19%, MF1 值分别提升了 2.12%、2.53%. 这说明了挖掘局部上下文中关键词的语义信息有助于提升方面级情感分析的性能. 因为在局部上下文中, 名词、形容词、动词这类词往往包含着评论者的语气和情绪. 本文重点关注局部上下文中的动词、名词、形容词, 与其他类别的词进行区分, 有效提升情感分类模型的性能, 减少噪声词的影响.

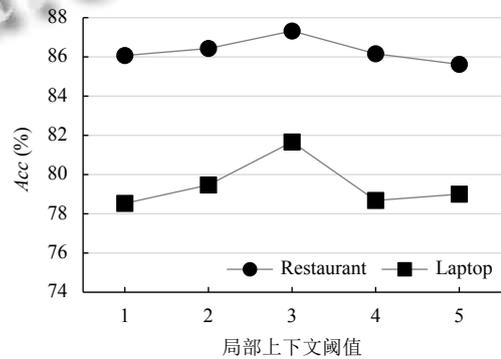


图 7 局部上下文阈值对准确率的影响

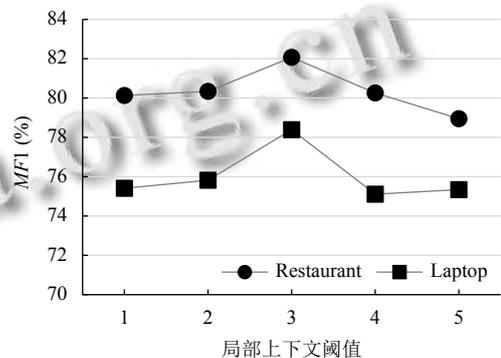


图 8 局部上下文阈值对 MF1 的影响

表 6 词性筛选的影响 (%)

模型	Restaurant		Laptop	
	Acc	MF1	Acc	MF1
无词性筛选	85.98	79.95	79.47	75.86
有词性筛选	87.32	82.07	81.66	78.39

注: 最佳效果用粗体标出.

4.6 案例分析

LCPM 模型重点关注方面词周围重要词的特征.

为研究 LCPM 模型的有效性,将该模型的注意力权重可视化以便于分析.如图 9 所示,颜色越深表示注意力权重值越大.以句子“but the staff was so horrible to us.”为例,这句话方面词是“staff”.在图 9 中,“staff”“was”“horrible”的颜色较深,表明这些词对方面词“staff”的影响较大,这些词往往能够决定方面词对应的情感极性,而其他词颜色较浅,表明这些词对方面词的影响较小.这验证了 LCPM 模型重点关注局部上下文中关键词的特征,加强文本中重要词汇对情感分析的影响,避免冗余信息的干扰.



图 9 注意力权重可视化

5 总结

针对方面级情感分析任务,本文创新地提出了一种基于局部上下文关键词特征提取及增强的方面级情感分析模型 LCPM,模型重点关注局部上下文中形容词、名词、动词等特征,避免全局上下文冗余信息的干扰.为了进一步让模型注重与方面词有关联的局部上下文关键词特征,本文对损失函数进行修改,把预测局部上下文关键词掩码矩阵作为辅助信息,准确地提取局部上下文中与方面词强相关的重要词的特征.为了得到更全面的情感特征,本文设计了一种门控机制,模型可以动态学习权重系数,给局部上下文关键词特征和全局上下文特征分配不同权重.在 4 个公开数据集实验结果表明,与其他常见的方面级情感分析模型对比,本文提出的 LCPM 模型在方面级情感分析任务上性能更优,能够准确提取用户所表达的情感信息.未来工作可以增加一些辅助信息确定局部上下文阈值 d_i ,也可以进一步探索损失函数优化方式,同时把 LCPM 模型应用到方面级情感分析的其他子任务上.

参考文献

1 Zhang C, Li QC, Song DW. Aspect-based sentiment classification with aspect-specific graph convolutional networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language

Processing. Hong Kong: ACL, 2019. 4568–4578. [doi: 10.18653/v1/d19-1464]

2 Ma DH, Li SJ, Zhang XD, *et al.* Interactive attention networks for aspect-level sentiment classification. Proceedings of the 26th International Joint Conference on Artificial Intelligence. Melbourne: IJCAI.org, 2017. 4068–4074. [doi: 10.24963/ijcai.2017/568]

3 Sun K, Zhang RC, Mensah S, *et al.* Aspect-level sentiment analysis via convolution over dependency tree. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: ACL, 2019. 5679–5688. [doi: 10.18653/v1/d19-1569]

4 Wang K, Shen WZ, Yang YY, *et al.* Relational graph attention network for aspect-based sentiment analysis. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. ACL, 2020. 3229–3238. [doi: 10.18653/v1/2020.acl-main.295]

5 Bai XF, Liu PB, Zhang Y. Investigating typed syntactic dependencies for targeted sentiment classification using graph attention neural network. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2020, 29: 503–514. [doi: 10.1109/taslp.2020.3042009]

6 Zhou XT, Zhang T, Cheng C, *et al.* Dynamic multichannel fusion mechanism based on a graph attention network and BERT for aspect-based sentiment classification. Applied Intelligence, 2022: 1–14. [doi: 10.1007/s10489-022-03851-3]

7 Liu BW, Blasch E, Chen Y, *et al.* Scalable sentiment classification for big data analysis using naive Bayes classifier. Proceedings of the 2013 IEEE International Conference on Big Data. Silicon Valley: IEEE, 2013. 99–104. [doi: 10.1109/bigdata.2013.6691740]

8 Kaji N, Kitsuregawa M. Building lexicon for sentiment analysis from massive collection of html documents. Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Prague: ACL, 2007. 1075–1083.

9 Kim Y. Convolutional neural networks for sentence classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Doha: ACL, 2014. 1746–1751. [doi: 10.3115/v1/d14-1181]

10 Cho K, Van Merriënboer B, Gulcehre C, *et al.* Learning phrase representations using RNN encoder-decoder for statistical machine translation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Doha: ACL, 2014. 1724–1734. [doi: 10.3115/v1/

- d14-1179]
- 11 Tang DY, Qin B, Feng XC, *et al.* Effective LSTMs for target-dependent sentiment classification. Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers. Osaka: The COLING 2016 Organizing Committee, 2016. 3298–3307.
 - 12 Sukhbaatar S, Szlam A, Weston J, *et al.* End-to-end memory networks. Proceedings of the 28th International Conference on Neural Information Processing Systems. Montreal: MIT Press, 2015. 2440–2448.
 - 13 Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. 5th International Conference on Learning Representations. Toulon: OpenReview.net, 2016.
 - 14 Huang BX, Ou Y, Carley KM. Aspect level sentiment classification with attention-over-attention neural network. Proceedings of the 2018 International Conference on Social Computing, Behavioral-cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation. Washington, DC: Springer, 2018. 197–206. [doi: [10.1007/978-3-319-93372-6_22](https://doi.org/10.1007/978-3-319-93372-6_22)]
 - 15 Fan FF, Feng YS, Zhao DY. Multi-grained attention network for aspect-level sentiment classification. Proceedings of the 2018 Conference on Empirical Methods in Natural Language. Brussels: ACL, 2018. 3433–3442. [doi: [10.18653/v1/d18-1380](https://doi.org/10.18653/v1/d18-1380)]
 - 16 Song YW, Wang JH, Jiang T, *et al.* Targeted sentiment classification with attentional encoder network. Proceedings of the 2019 International Conference on Artificial Neural Networks. Munich: Springer, 2019. 93–103. [doi: [10.1007/978-3-030-30490-4_9](https://doi.org/10.1007/978-3-030-30490-4_9)]
 - 17 Nguyen TH, Shirai K. PhraseRNN: Phrase recursive neural network for aspect-based sentiment analysis. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Lisbon: ACL, 2015. 2509–2514. [doi: [10.18653/v1/d15-1298](https://doi.org/10.18653/v1/d15-1298)]
 - 18 Chen P, Sun ZQ, Bing LD, *et al.* Recurrent attention network on memory for aspect sentiment analysis. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen: ACL, 2017. 452–461. [doi: [10.18653/v1/d17-1047](https://doi.org/10.18653/v1/d17-1047)]
 - 19 Nguyen HT, Le Nguyen ML. Effective attention networks for aspect-level sentiment classification. Proceedings of the 2018 10th International Conference on Knowledge and Systems Engineering (KSE). Ho Chi Minh: IEEE, 2018. 25–30. [doi: [10.1109/kse.2018.8573324](https://doi.org/10.1109/kse.2018.8573324)]
 - 20 Xu H, Liu B, Shu L, *et al.* BERT post-training for review reading comprehension and aspect-based sentiment analysis. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis: ACL, 2019. 2324–2335.
 - 21 Huang BX, Carley KM. Syntax-aware aspect level sentiment classification with graph attention networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: ACL, 2019. 5469–5477. [doi: [10.18653/v1/d19-1549](https://doi.org/10.18653/v1/d19-1549)]
 - 22 宋威, 温子健. 基于特征双重蒸馏网络的方面级情感分析. 中文信息学报, 2021, 35(7): 126–133. [doi: [10.3969/j.issn.1003-0077.2021.07.015](https://doi.org/10.3969/j.issn.1003-0077.2021.07.015)]

(校对责编: 孙君艳)