

分类号 TP391.1
U D C

密级
编号 10741

兰州财经大学

LANZHOU UNIVERSITY OF FINANCE AND ECONOMICS

硕士学位论文

论文题目 一种基于双通道信息融合的方面级情感
分类模型

研究生姓名: 黄建民

指导教师姓名、职称: 李强 教授

学科、专业名称: 管理科学与工程

研究方向: 信息管理与信息系统

提交日期: 2023年6月6日

独创性声明

本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文作者签名： 黄建民 签字日期： 2023年5月20日

导师签名： 李强 签字日期： 2023年5月20日

关于论文使用授权的说明

本人完全了解学校关于保留、使用学位论文的各项规定， 同意（选择“同意” / “不同意”）以下事项：

1. 学校有权保留本论文的复印件和磁盘，允许论文被查阅和借阅，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文；

2. 学校有权将本人的学位论文提交至清华大学“中国学术期刊（光盘版）电子杂志社”用于出版和编入 CNKI《中国知识资源总库》或其他同类数据库，传播本学位论文的全部或部分內容。

学位论文作者签名： 黄建民 签字日期： 2023年5月20日

导师签名： 李强 签字日期： 2023年5月20日

**An Aspect-level Sentiment Based on
Two-channel Information Fusion
Classification Model**

Candidate : Huang JianMin

Supervisor: Li Qiang

摘 要

方面级情感分类是情感分析中的细粒度任务,其目标是识别句子中某一方面对应的情感态度,这类针对具体方面的情感提取能够在个人、企业和相关部门作出决策时提供更加精准的数据支撑,因此具有非常大的现实意义。目前大多数情感分类模型在构建时往往只提取文本信息的某一类特征,而忽略了将多种特征融合也能给模型带来效果的提升。为解决以上问题,本文提出一种基于双通道信息融合的 GCNN 模型。模型构建主要思路如下:

(1) 在方面级情感分类任务中,传统情感分类模型往往将词语作为单一嵌入,而忽略词语的词性信息和位置信息,因此在模型预测表现上并不理想。本文用 BERT 预训练模型增强传统词嵌入的情感语义表征能力,同时将词性信息和位置信息融入得到最终的词向量表示,缓解词向量表征不足的问题。

(2) 卷积操作能够实现文本特征的有效提取,本文设计了基于多粒度卷积操作的 MulCNN 模块用于提取文本的语义特征。不同于常用的单个卷积核,模块能同时采用三个不同粒度的卷积核提取文本序列的 N-gram 特征,粒度不同的卷积核提取到的特征矩阵并不相同,相较于单粒度卷积操作,本文提出的 MulCNN 模块能够提取到句子中更丰富的表征信息。

(3) 尽管 N-gram 特征能够很好的帮助模型判断方面词的情感极性,但是单一特征的深度学习模型并未考虑到句法结构的重要性,本文设计了基于句法关系的 DepGCN 模块。模块首先通过依赖解析模型学习句子依赖关系得到词语间的逻辑距离,接着以词语和依赖关系分别作为图卷积神经网络的初始节点和边,得到包含丰富句法特征的向量表示。

(4) 传统模型为融合特征往往对特征向量进行简单的拼接操作,但是这类操作并没有进一步挖掘信息间的关联性。本文同时兼顾语法特征相关性和语义特征互补性,基于注意力机制构建了特征融合层用于双通道的特征融合,达到了增强特征向量表达的能力。

经过 BERT 预训练后的词嵌入表示蕴含着更为丰富的特征信息,这些高质量的词向量经过双通道架构能够同时学习句子的语义和语法特征,并在特征融合层实现特征的有效交互,充分发挥语义和语法特征的互补优势,增强了模型的鲁棒性。当句子意思表达充分时,GCNN 模型可以依靠包含语义信息的 MulCNN 模

块实现情感分类；当句子句法关系明显时，GCNN 模型可以依靠包含句法信息的 DepCNN 模块实现情感分类。通过将 GCNN 模型与选取的 12 个代表模型在三个公开数据集上做对比实验可知，本文提出的模型在预测准确率和 F1 值方面表现出色，同时本文还设置了一系列实验证明本文提出的 GCNN 模型能够有效挖掘并融合特征信息，证实了 GCNN 模型的可靠性和有效性。

关键词: 情感分析, BERT, 多粒度卷积, 图卷积神经网络, 双通道, 特征融合

Abstract

Aspect-level sentiment classification is a fine-grained task in sentiment analysis, and its goal is to identify the sentiment attitude corresponding to a certain aspect of a sentence. This kind of aspect-specific sentiment extraction can provide more accurate data to support individuals, enterprises, and related departments in making decisions, and therefore has great practical significance. Most of the current sentiment classification models are constructed by extracting only one type of feature of text information, ignoring the fact that the fusion of multiple features can also improve the model's effectiveness. To solve the above problems, this paper proposes a GCNN model based on two-channel information fusion. The main idea of model construction is as follows:

(1) In aspect-level sentiment analysis tasks, traditional sentiment classification models tend to treat words as a single embedding and ignore lexically and location information of words, and thus are not satisfactory in model prediction performance. In this paper, we use a BERT pre-training model to enhance the sentiment-semantic representation of traditional word embeddings, while incorporating lexical and location information to obtain the final word vector representation to alleviate the problem of insufficient word vector representation.

(2) In this paper, we design a MulCNN module based on multi-granularity convolutional operations to extract semantic features of the text.

Unlike the commonly used single convolutional kernel, the module can use three different granularity convolutional kernels to extract N-gram features of text sequences at the same time, and the feature matrices extracted by the different granularity convolutional kernels are not the same.

(3) Although N-gram features can well help the model to judge the sentiment polarity of aspect words, the deep learning model with single features does not consider the importance of syntactic structure, and the DepGCN module based on syntactic dependencies is designed in this paper. The module first learns the sentence dependency relationship to get the logical distance between words by the dependency resolution model, and then uses words and dependencies as the initial nodes and edges of the graph convolutional neural network, respectively, to get a vector representation containing rich syntactic features.

(4) Feature information is the key to the model's ability to accurately predict sentiment polarity. Traditional models often perform simple splicing operations on feature vectors to fuse features, but such operations do not further explore the correlation between information. In this paper, we take into account both syntactic feature relevance and semantic feature complementarity and build an information fusion layer for two-channel feature fusion based on the attention mechanism, which achieves the ability to enhance feature vector representation.

The word embedding representation after BERT pre-training contains

richer feature information, and these high-quality word vectors can learn both semantic and syntactic features of the sentences through the dual-channel architecture, and realize effective feature interaction in the feature fusion layer to give full play to the complementary advantages of semantic and syntactic features and enhance the robustness of the model. When the sentence meaning is adequately expressed, the GCNN model can rely on the MulCNN module containing semantic information to achieve sentiment classification; when the syntactic relationship of the sentence is obvious, the GCNN model can rely on the DepCNN module containing syntactic information to achieve sentiment classification. By comparing the GCNN model with the selected 12 representative models on three public datasets, it can be seen that the model proposed in this paper performs well in terms of prediction accuracy and F1 value, and a series of experiments are also set up in this paper to prove that the GCNN model proposed in this paper can effectively mine and fuse feature information, which confirms the reliability and validity of the GCNN model.

Keywords: Sentiment Analysis, BERT, Multi-granularity Convolution, GCN, Dual-channel, Feature Fusion.

目 录

| | |
|-------------------------------------|-----------|
| 1. 绪 论 | 1 |
| 1.1 研究背景及意义 | 1 |
| 1.1.1 研究背景 | 1 |
| 1.1.2 研究意义 | 2 |
| 1.2 国内外研究现状 | 2 |
| 1.2.1 基于情感词典的方法 | 3 |
| 1.2.2 基于机器学习的方法 | 3 |
| 1.2.3 基于深度学习的方法 | 4 |
| 1.3 论文研究内容 | 7 |
| 1.4 创新点 | 7 |
| 1.5 论文组织结构 | 8 |
| 2. 相关理论和技术 | 9 |
| 2.1 情感分析概述 | 9 |
| 2.2 词向量表示 | 9 |
| 2.2.1 One-hot 编码 | 9 |
| 2.2.2 Word2Vector | 10 |
| 2.2.3 GloVe | 12 |
| 2.2.4 BERT | 13 |
| 2.3 循环神经网络 | 15 |
| 2.4 卷积神经网络 | 18 |
| 2.4.1 普通卷积神经网络 | 18 |
| 2.4.2 图卷积神经网络 | 19 |
| 2.5 注意力机制 | 22 |
| 2.6 依存句法分析 | 23 |
| 2.7 特征融合方法 | 25 |
| 2.8 本章小结 | 27 |
| 3. 基于双通道信息融合的方面级情感分类模型 | 28 |

| | |
|--------------------------------|-----------|
| 3.1 词嵌入层..... | 29 |
| 3.2 Bi-GRU 层..... | 30 |
| 3.3 多粒度卷积层..... | 31 |
| 3.4 句法关系及图卷积网络层..... | 32 |
| 3.5 特征融合层..... | 34 |
| 3.6 输出层..... | 36 |
| 3.7 本章小结..... | 36 |
| 4. 实验与分析 | 37 |
| 4.1 数据集介绍..... | 37 |
| 4.2 评价指标..... | 38 |
| 4.3 实验环境与超参数配置..... | 39 |
| 4.3.1 实验环境..... | 39 |
| 4.3.2 参数设置..... | 40 |
| 4.4 对比模型设置..... | 42 |
| 4.5 实验结果及分析..... | 44 |
| 4.6 相关实验验证..... | 46 |
| 4.6.1 卷积对比实验..... | 46 |
| 4.6.2 消融实验..... | 47 |
| 4.6.3 GCN 层数对模型性能的影响..... | 49 |
| 4.6.4 注意力可视化..... | 50 |
| 4.6.5 实例分析..... | 52 |
| 4.7 本章小结..... | 53 |
| 5. 总结与展望 | 55 |
| 5.1 工作总结..... | 55 |
| 5.2 未来展望..... | 56 |
| 参考文献 | 57 |
| 致 谢 | 64 |
| 攻读硕士学位期间发表的论文及科研情况..... | 65 |

1. 绪论

1.1 研究背景及意义

1.1.1 研究背景

语言文字作为文明社会产生的标志,时刻记录着人类思想的进步和历史的变迁。受限于器皿、竹简等载体固有的缺陷,文字往往只记录着科学、历史等客观性内容。西汉时期造纸术的发明,使文字大篇幅记载成为可能,越来越多的个人思想和情绪也体现在文字之中,为他人了解记录者的情感态度提供巨大的帮助。近三十年来,以计算机、新材料为代表的新兴技术引起了第三次科技革命,文字有了新的载体,这极大的拓宽了情感分析的数据来源渠道。生活中,无论是个体或者大型团体,想要作出正确的选择都需要充分参考相关人员的态度,而蕴含在文字中的观点则是人们进行情感分析的重要途径。

自 2020 年新冠疫情爆发以来,各行各业都受到了大幅度的冲击和影响,实体店经济出现日趋下降的现象,而互联网依靠线上模式愈加融入到每个家庭的日常生活中。据中国互联网络信息中心(China Internet Network Information Center, CNNIC)第 50 次《中国互联网络发展状况统计报告》显示^[1],我国互联网网民规模为 10.51 亿,互联网普及率达 74.4%,较上次统计时新增网民 1919 万,互联网普及率提升 1.4 个百分点。在疫情背景下,中国电子商务行业迎来了新机遇,以抖音为代表的“直播+电商”的新零售业全面爆发。在每个直播带货短视频中都会留下成千上万条评论,如果能从这些评论中获得消费者对商品评价,就可以给潜在消费者提供购买意见,保障消费者权益。

情感分析也被称作观点挖掘,是自然语言处理(Natural Language Processing, NLP)领域众多研究方向中的一个分支,旨在从主观性文本中找到人们对某一事件、产品、服务及其相关属性的情感态度。近年来,数据量的爆发式增长以及计算机性能的显著提升使越来越多人开始研究情感分析任务,研究成果也大量涌现并应用到实际生活中。

1.1.2 研究意义

目前,情感分析研究已经涉及多个领域。在电商平台中,情感分析可以帮助消费者提前了解商品的质量和服务体验等属性,帮助消费者在对商品或服务有了全方位了解后再决定是否购买,防止出现因商家过度宣传而被骗的情况;另一方面,情感分析对于商家而言也大有益处,商家可以从消费者的评价中明白哪些商品属于消费者喜欢的产品,哪些商品又是因为价格、质量等因素而导致消费者购买意愿低迷,进而为调整营销策略提供依据。在舆情管控中,人们对当前社会热点事件的看法和观点都包含在网上发表的评论中,对这些评论进行情感判别可以了解人们对整个事件的态度,相关部门据此能够提前做出干预措施进行积极化的舆情引导,确保社会处于良好的秩序。在预测领域,情感分析可以根据当前人们的情绪状态提前预测未来的走势,从而给预测者做出最优的选择。例如在电影票房预测中,可以通过预先了解电影上映前和上映中观众的情感态度分布预估电影的最终票房。

各领域对情感态度的需求促使情感分析技术热点化,作为 NLP 领域的一个分支,情感分析任务主要有实体抽取^[2]、情感极性判断^[3]、情绪识别^[4]和意图分析^[5]。根据所分析文本粒度的不同又可以分为方面级、句子级^[6]和篇章级^[7]情感分析。本文研究对象为评论文本,通过建立情感分析模型对商品评论的情感极性进行分析,最终得出评论语句的情感态度。

1.2 国内外研究现状

当前情感识别方法可分为三类:基于情感词典、基于机器学习和基于深度学习。第一类方法利用已有的情感字典,通过对语篇中的情绪词汇进行匹配并赋予相应情感分值,从而获得情绪态度;基于机器学习的方法通过算法学习手工标注的情感特征,最终对文本情感进行预测;基于深度学习的方法是目前使用最多的方法,该方法通过构建深度学习模型自动学习给定训练集的特征从而对测试集作出预测分析。

1.2.1 基于情感词典的方法

基于情感词典的方法是最早用于情感分析的方法,该方法将给定文本进行分词,接着依照构建好的情感词典进行词语的匹配,然后赋予一定的情感得分,最后进行情感分值计算实现分类。显然,情感词典的质量对于最终的预测结果十分重要,目前研究人员常用的情感词典有 HowNet^[11]和 NTUSD^[12]。使用情感词典来进行情感态度的分析判别,首先要对目标文字进行预处理,其中包含去标点、无效字符等操作,接着对经过处理的文字进行分词,再将表达情绪的词与程度词相对应,最后按照设定好的情绪评分规则来获得情感得分。Hu 等人^[13]将 WordNet 作为情感词典,将目标文本的词语与情感词典匹配,同时考虑词语的近义和反义关系,实现目标文本的情感极性判断。Basant 等人^[14]在提取评论语句的情感时,将领域内独有的情感色彩词语加入到情感词典中,结果表明扩充后的情感词典对结果有进一步的提升。常用的情感词典能适应一般化的情感分析任务中,当涉及到特定领域时却不能得到好的效果,罗浩然等人^[15]收集了教育行业专业机器人的常用词汇,构建了教育机器人专用词典,从而给教育机器人行业的消费需求者提供建议。

通过构建情感词典实现情感分类的方法能够较准确的反应文本的非结构化特征,实现起来也相对简单。但是这类方法对情感词典的依赖性过高,构建者经验的丰富程度直接影响词典质量的好坏,且由于时间、语言、领域等因素的影响,相同的情绪词在词汇上的意义可能会发生变化,这就导致了在跨领域、跨语言环境下,基于情感字典的识别算法并不能取得令人满意的结果。目前,基于情感字典的情绪分类方法不能识别文本间的语义关联,这一问题有待于研究人员进一步探究。

1.2.2 基于机器学习的方法

机器学习就是将已有的数据对模型进行训练,接着使用训练好的模型对目标数据进行预测,从而解决实际问题。这种方法现在仍不断有新的突破与创新,在某些方面的研究上也已取得许多令人瞩目的成就。随着计算机视觉技术的发展以

及人工智能理论与实践的不断融合,机器学习已成为当前人工智能的一个重要组成部分,其应用领域也越来越广。最早把机器学习方法运用到情感分析领域的是 Pang 等人^[16],他们采用朴素贝叶斯(Native-Bayes, NB),最大熵(Maximum Entropy, ME)和支持向量机(Support Vector Machine, SVM)三种方法对电影评论实现情感分析。Sharma 等人^[17]以 SVM 为基分类器,通过 Boosting 集成算法提升分类效果,在文本评论数据集的实验结果表明该方法对情感分析效果有明显的提升。濮泽堃^[18]提出了一种基于 SD-LS-SVM 的情感分类算法,该算法引入改进的蚁群算法对 LS-SVM 算法的稀疏度进行优化,最终在电商评论数据集上取得了优于对比模型的分类结果。

提取情感特征和选择合适的分类器是机器学习中至关重要的一步,因为不同的分类器组合会产生不同的情感分析结果,从而影响预测的准确性和效率。这类方法经过多次方案调优最终都能达到不错的效果,而加入领域特定知识和专家经验后模型性能还能有进一步的提升。基于机器学习的方法也存在不足之处,即该方法仍然需要耗费大量的时间和精力对特征进行标注,而在大数据的时代背景下依赖手工构建特征显然不现实。另外,机器学习方法在特定领域和任务中不能通用,训练出来的模型泛化能力差。

1.2.3 基于深度学习的方法

随着计算机性能的提升,研究人员开始探索深度学习模型在各领域的应用。词嵌入(Word Embedding)技术最先被应用到 NLP 领域中。Mikolov 等人^[19]提出了 Word2vec 技术,该技术可以将文字信息转换成带有语义信息的向量,并且具有很高的计算效率和准确率。由于词嵌入技术只包含了局部的信息,并不包含全局的语义和语法关系,Tang 等人^[20]通过捕获上下文和情感水平信息学习更强大的连续单词表示,在三个情感分析任务中证明了情感嵌入的有效性。目前,搭建一个基于深度学习的情感分类模型常用的架构包括:卷积神经网络(Convolutional Neural Networks, CNN)、循环神经网络(Recurrent Neural Network, RNN)、注意力机制、门控机制、预训练语言模型等。

2014 年, Kim^[21]针对 CNN 的输入层做出调整并提出了 TextCNN 模型用于

情感分析任务，TextCNN 构建了一个向量矩阵用于表示文本信息，矩阵的每一行都代表着组成文本的词，接着通过一维的卷积操作对向量矩阵进行特征识别，实验结果证明了 CNN 在情感分析任务中的有效性。TextCNN 是一个 CNN 为网络的文本分类模型结构简单，训练速度快，效果也不错，但是 TextCNN 的可解释型不强，很难去评估每个特征的重要度，Lai 等人^[22]采用循环神经网络代替了 TextCNN 的特征提取过程，构建了 TextRCNN 分类模型，改进的模型最终在实验中取得了更好的效果。郑诚等人^[23]在 CNN 基础上融入词性信息，丰富了 CNN 的特征向量，文章的消融实验证明了融入其他模块也能够提升情感分析准确率。

基于 CNN 的情感分类模型可以避免人工处理特征造成的不足，但是这类模型不能确定长文本中句与句之间的依赖关系。考虑到文本语句间有着时间上的先后顺序，研究人员尝试用 RNN 处理文本特征，RNN 能够在记忆特征的同时实现参数间的共享，保证了参数的可靠性，但是 RNN 在反向传播时，参数连乘容易出现梯度消失和梯度爆炸问题。为此，学者们提出了长短时记忆网络(Long Short Term Memory,LSTM)^[24]和门控循环单元(Gated Recurrent Unit, GRU)^[25]。门控机制能有效缓解梯度问题从而保证 RNN 能学习到句子间长距离的依赖关系。Zhang 等人^[26]认为仅仅依靠一个方向的传播是能完全表达语义关系的，于是提出了双向的 LSTM 同时从前向和反向对特征进行提取，实验结果表明双向 LSTM 效果要优于单向 LSTM。由于 LSTM 和 GRU 能够将相关信息传送到长序列文本中进行预测，许多情感分析模型都基于 LSTM 或 GRU 建模。

在真实语境中，只根据一句话中某些重要的单词就可以完成指定方面的情感分类任务，而神经网络不能自动学习到文本中不同句子、单词对给定方面的重要程度，这限制了模型准确率的提升。一些学者将注意力机制引入到情感分析中，以此获得上下文词对方面词的重要度。Wang 等人^[27]将 LSTM 与注意力机制结合建立情感分类模型，将给定的方面词分别拼接到句子不同部分，学习其他词对方面词的重要性权值，结果表明，注意力机制能够显著提升方面级情感分类模型的性能。胡等人^[28]提出用多层自注意力机制获得更新的局部特征权重，中文数据集上的实验表明模型能取得不错的效果。注意力机制能够有效捕捉方面词的权重信息，但是注意力机制也容易忽略否定副词的重要性从而增加数据噪声。He 等人

[29]引入注意力机制后加入句法约束条件, 预先判断否定副词从而减少噪声对模型的影响。Huang 等人^[30]将方面和句子联合建模, 明确地捕捉上下文和句子间的交互信息, 通过自己构建的 AOA 模块自动关注句子中的重要词, 最终取得较好的结果。随着学者们对神经网络的深入研究, 研究人员尝试将 CNN 与图嵌入技术相结合, 以此丰富特征信息。Scarselli 等人^[31]提出图神经网络 (graph neural network, GNN)模型, 成功将神经网络使用在图结构数据上。进一步的, 学者们在 GNN 基础上加入特征表示学习以此应用到自然语言处理领域^{[32][33]}, Basting 等^[34]对 GCN 进行泛化处理以此加入句法依存关系, 为编码器提供了丰富的语法信息, 最终在机器翻译任务中获得优秀的成绩。巫浩盛等人^[35]用 GCN 对文本中上下文语义关系、语法间距离权重进行情感特征提取, 提出基于图卷积神经网络和句法距离的方面级情感分类模型。由于 GCN 忽略了有序词语的上下文语义关系, 谢珺等人^[36]用双指导注意力网络加强了上下文信息, 并在情感分析常用数据集上验证了方法的可行性。

随着计算机性能的提升, 预训练模型开始大量应用到自然语言处理领域。由于 Word2vec 只能表示文本的局部信息, Pennington 等^[37]提出一种 GloVe(global vectors for word representation)算法, 该算法引入共现概率矩阵将语料库的全局统计特征和滑动窗口特征合并, 实现了全局信息的利用。虽然 GloVe 一定程度上丰富了词嵌入信息, 但仍是一种静态的词嵌入方式, 而在日常生活中, 一词多义的现象是极为常见的, GloVe 无法识别不同场景下字词含义的变化。Peters 等^[38]提出可以对多义词建模的 ELMo (embedding from language models)方法来解决传统词向量无法正确区分多义词的问题, ELMo 模型首先通过大量的语料进行学习, 接着针对具体的下游任务对每个词向量微调以适应要求。2017 年, 谷歌团队提出了完全基于注意力机制的 Transformer 模型^[39], 该模型具有强大的特征抽取能力, 在精度和性能上都要远远高于之前流行的 RNN。Radford 等人^[40]在 Transformer 的解码器结构上提出了生成式预训练模型 (generative pre-training, GPT) 模型, 该模型采用二段式的训练, 模型首先进行无监督式预训练, 接着对下游具体任务进行有监督式的微调。同年, 谷歌团队提出了基于 BERT^[41]的预训练语言模型, BERT 模型采用 Transformer 的编码器结构, 实现了对上下文关系

的双向抽取，该模型一经提出，就刷新了自然语言处理领域的多个记录。如今，许多优秀的预训练模型都是在 BERT 基础上加以改进的。

1.3 论文研究内容

如何准确且高效地识别方面词对应的情感态度是方面级情感分析任务的关键，本文主要对情感分类模型进行研究改进，针对前人研究成果中存在的不足和缺陷进行补足，为情感分析领域贡献更好的模型。具体表现为：

传统的词嵌入方式只简单的满足了将文本转成向量的要求，但是在语义信息表达方面并不充分，且无法有效识别日常生活中常用的多义词。针对此问题，本文引入了最先进的 BERT 预训练语言模型用于丰富特征信息，并在词嵌入层融入词性和位置信息，进一步加强词嵌入的信息表示。在特征提取层时，基于 CNN 和 RNN 的模型不能识别词语间长距离的语义信息，且考虑到句法关系能够增强模型理解文本的能力，本文构建了基于句法关系的图卷积网络通道 (DepGCN)，DepGCN 首先对词向量进行依存句法分析，再将词和词语间的关系分别表示为 GCN 网络中的节点和度长，实现句子中句法信息的全面表示。显然，只包含句法特征的向量矩阵是不完全的，本文还构建了基于语义的多粒度卷积网络通道 (MulCNN)，MulCNN 采用了不同规格的卷积核提取文本序列的 N-gram 特征，进一步丰富特征表示。最后，通过注意力机制将学习到的特征进行融合，一方面减少了特征的冗余度，另一方面也增强了特征，进而更加准确的预测情感态度。

本文在 SemEval 2014 中的 Restaurant 和 Laptop 数据集及 Twitter 数据集上验证模型的有效性。通过与基准模型做比较判断本文模型的优劣，同时设计对比实验、消融实验、实例分析等方式验证模型各模块有效性。

1.4 创新点

(1) 近年来，方面级情感分类模型的构建多以聚合多种方法的混合策略模型为主，基于句法依存树结合图卷积神经网络的建模方法在方面级情感分类模型中成为已经主流。这类模型可以有效地挖掘文本的句法信息特征，但是情感极性

的判断特征并不具有唯一性,还包括语义信息、位置信息等多种特征。本文构建了双通道模型,在学习句法信息的基础上,额外构建语义特征提取通道,通过多粒度卷积操作获得评论文本的语义特征,接着对图卷积神经网络的输出向量和多粒度卷积网络的输出向量进行特征融合,达到增强特征向量表达的效果。

(2) 特征融合是深度学习领域常用的方法,最常见的融合方式包括按点逐位相加(point-wise addition)和向量拼接(concatenate),而这些向量融合方式得到的新特征向量只是实现了特征合并,并没有进一步挖掘特征间的关系。本文用注意力机制代替了向量拼接操作,同时兼顾语法特征相关性和语义特征互补性,突出特征向量中对方面情感分析更有帮助的部分,进而实现特征加强,增加模型预测准确率。

1.5 论文组织结构

本文总共分为五章,各章主要内容为:

第一章:绪论。绪论部分对本文的研究背景及意义、方面级情感分析领域的研究现状做出阐述,其中研究现状对实现情感分析的三大方法做了详细介绍,最后简要概括本文的研究内容。

第二章:相关理论和技术。本章介绍了完成情感分类任务所需要的相关理论和技术,包括方面级情感分析概述、预处理技术、词向量表示技术、卷积神经网络、循环神经网络、图卷积网络、注意力机制和特征融合方法。

第三章:基于双通道信息融合的方面级情感分类模型。本章详细介绍了如何构建基于双通道信息融合的方面级情感分类模型,包括用于学习句法信息的 DepGCN 通道、用于丰富文本序列特征的 MulCNN 通道、用于将双通道特征信息融合进而实现特征增强的注意力机制及最后的输出层。

第四章:实验与分析。本章讲述了模型性能验证过程,包括对评价指标、实验数据集和实验环境的介绍,超参数和对比模型的选择,对比实验、消融实验和实例验证的分析。

第五章:总结与展望。本章对全文工作总结同时指出下一步研究方向。

2. 相关理论和技术

2.1 情感分析概述

情感分析按粒度大小可以划分为篇章级^[8]、句子级^[9]和方面级^[10]。方面级情感分析可以从句中的具体方面找到对应的情感态度,从而让用户更加细致了解特定目标的属性,因此是一个细粒度的任务。例如句子“Yesterday's event was very successful, but everyone's enthusiasm was not high”,这句话存在两个目标对象,对象“event”的相应情感态度为“successful”,体现的是积极的评价,对象“enthusiasm”的相应情感态度为“not high”,体现的是消极的评价,方面词的不同,产生的评价也存在差异。

利用深度学习模型进行方面级情感分析主要包括以下步骤:数据预处理、特征提取、向量表示和情感输出。其中数据预处理是对给定的任务文本文字进行分词得到包含文字信息的词序列,利用一个预先训练好的模型,将其转化成一个词向量矩阵嵌入模型;特征抽取是对输入的词向量矩阵进行语义、语法、位置等特征进行挖掘;向量表示在于把得到的特征信息用向量矩阵表示出来,情感输出即根据任务目标对得到的表示向量进行标签化处理。

2.2 词向量表示

词向量表示又称词嵌入,即将语言文字通过一定技术转成为向量矩阵,以便输入电脑端进行后续任务。目前词向量表示方经历了长足的发展,出现了许多词向量表示技术,其中具有代表性的方法有: One-Hot 编码、Word2vec、GloVe、BERT。本节将详细介绍这四种词嵌入模型。

2.2.1 One-hot 编码

为了让机器也能理解语言文字信息,人们用汇编语言的“0”和“1”表示文字特征,用“0”表示特征不存在,用“1”表示特征存在。例句:我每天都努力,首先将句子划为{“我”,“每天”,“都”,“很”,“努力”}5个词汇,将状态

寄存器的数量设为 5，那么对应的特征向量的表示可为：{我：[1,0,0,0,0]}、{每天：[0,1,0,0,0]}、{都：[0,0,1,0,0]}、{很：[0,0,0,1,0]}、{努力：[0,0,0,0,1]}。下面将举例对 One-hot 编码的实际应用作出解释。

假设语料库中包含“我每天都很努力”，“爸爸每天努力工作”两个语句，首先将语句分词分别得到{“我”，“爸爸”，“每天”，“都”，“很”，“努力”，“工作”}，最终得出句子“我每天都很努力”的特征向量为：[1,0,1,1,1,0,1]；句子“爸爸每天努力工作”的特征向量为：[0,1,1,0,0,1,1]。具体的特征向量提取过程如图 2.1 所示。

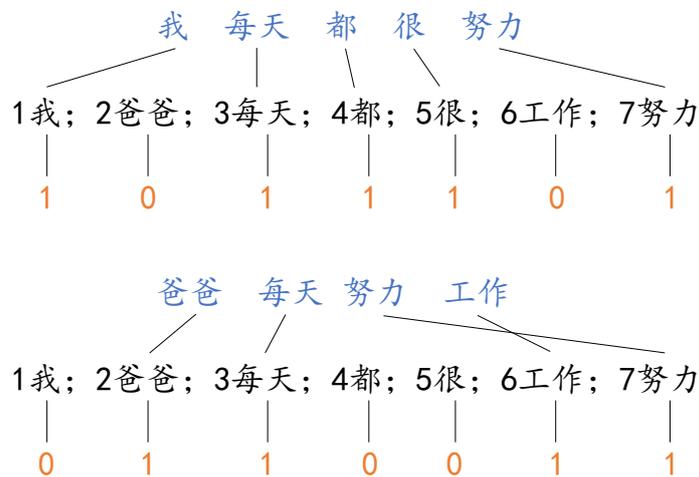


图 2.1 One-hot 提取特征向量过程

从实例中可以看出 One-hot 编码简单且易于理解，一定程度上表示了语言特征信息，但 One-hot 编码也存在明显的缺点。一方面，当本文信息量过大时，该编码方法的向量表示维度会很大且稀疏，容易出现维度灾难；另一方面，One-hot 编码对于容易影响句子意思表的词间顺序信息并不敏感，忽略重要信息。

2.2.2 Word2Vector

研究人员基于分布式表达提出了 Word2Vector 方法用于代替结构复杂的神经网络语言模型^[42](Neural Network Language Model, NNLM)，以此获得更高质量的词向量表示。Word2Vector 方法没有拼接层，减少了训练参数，提升了训练速度，也可以不通过激活函数就直接在隐藏层输出结果。具体来看，Word2Vector 可分

为连续词袋模型 (Continuous Bag Of Words,CBOW)和跳词模型(Skip-gram)两种类型。

(1) CBOW

CBOW 模型可以通过已知的上下文信息预测中心词，CBOW 模型结构如图 2.2 所示。

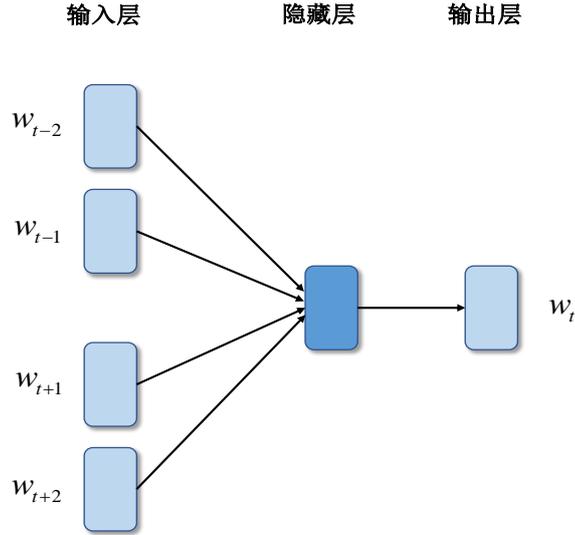


图 2.2 CBOW 模型结构

对于一个含有 T 个词的文本 $\{w_1, w_2, w_3, \dots, w_T\}$ 来说，模型中的输入层就是单词基于文本的 One-hot 表示，是基于中心词的上下文词 w_{t+1} 、 w_{t-1} 等；隐藏层是一个 $T \times N$ 的矩阵， N 为词向量的维度；通过隐藏层的学习，最终在输出层用 softmax 函数实现对中心词 w_t 的预测。CBOW 模型采用最大化平均对数概率作为训练目标，具体训练公式和目标函数 L_{CBOW} 如式 (2-1)、(2-2) 所示。

$$P(w_t | w_{t+j}) = \frac{\exp(\langle v_t, v_{t+j} \rangle)}{\sum_{k=1}^T \exp(\langle v_k, v_{t+j} \rangle)} \quad (2-1)$$

$$L_{CBOW} = \frac{1}{T} \sum_{t=1}^T \sum_{-c \leq j \leq c, j \neq 0} \log P(w_t | w_{t+j}) \quad (2-2)$$

式中 c 为基于中心词的上下文词向量数量， v_t 表示 w_t 对应的 One-hot 向量，

$\langle v_t, v_{t+j} \rangle$ 表示向量 v_t 和 v_{t+j} 的点积运算。

(2) Skip-gram

与 CBOW 模型功能相反, Skip-gram 模型旨在用中心词预测附近的上下文信息, 具体模型结构如图 2.3 所示。

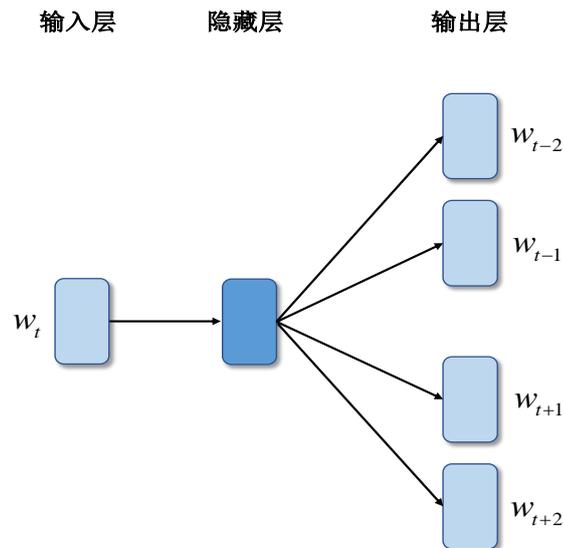


图 2.3 Skip-gram 模型结构

模型输入层为中心词, 输出层则为上下文词对于给定中心词的概率。Skip-gram 模型同样采用最大化平均对数概率作为训练目标, 但是目标函数与 CBOW 的目标函数存在差异, 训练公式和目标函数 L_{Skip} 如式 (2-3)、(2-4) 所示。

$$P(w_{t+j} | w_t) = \frac{\exp(\langle v_t, v_{t+j} \rangle)}{\sum_{k=1}^T \exp(\langle v_k, v_t \rangle)} \quad (2-3)$$

$$L_{Skip} = \frac{1}{T} \sum_{t=1}^T \sum_{-c \leq j \leq c, j \neq 0} \log P(w_{t+j} | w_t) \quad (2-4)$$

2.2.3 GloVe

2014 年, 斯坦福大学 NLP 组在自然语言处理顶级会议 EMNLP 上发表了论文《GloVe: Global Vectors for Word Representation》, 这篇文章提出了一种全新的预训练模型 GloVe。正如论文名字所言, GloVe 是一种基于全局词频统计的词向

量表示方法，该方法将单词转换成包含一些语义特征的向量，并通过欧几里得距离或余弦相似度计算语义相似性。具体实现步骤如下：

对于给定的语料库，将词 i 和其上下文词 j 共同出现在规定范围内的次数构建成一个对称矩阵，对于中心词 k ，计算 k 对 i 和 j 词的相关性公式如式 (2-5) 所示。

$$F(\mathbf{w}_i, \mathbf{w}_j, \mathbf{w}_k) = \frac{P_{ik}}{P_{jk}} \quad (2-5)$$

一般而言，次数最小值为 1，但是作者加入了衰减函数，该函数用于计算两个单词在窗口的距离，用距离的倒数表示衰减函数的值，两个词距离越远，那么两个词的在总词之间占的权重就越低。计算到词之间的相关性后构建的损失函数如式 (2-6) 所示。

$$Loss = \sum_{i,j=1}^V f(X_{ij}) (w_i^T w_j + b_i + b_j - \log X_{ij})^2 \quad (2-6)$$

GloVe 在均方误差的基础上加入权重函数 $f(x)$ 作为最终的损失函数，用于优化预测值和真实值之间的误差，其中 V 为语料库词表的大小，为确保一个词的权重达到一定程度后不会再增加，作者采用了分段函数 $f(x)$ 实现上述目标，分段函数如式 (2-7)。

$$f(x) = \begin{cases} (x/x_{\max})^\alpha & \text{if } x < x_{\max} \\ 1 & \text{otherwise} \end{cases} \quad (2-7)$$

式中 α 和 x_{\max} 为超参数，文中的所有实验， α 的取值都是 0.75，而 x_{\max} 取值都是 100。

2.2.4 BERT

BERT 预训练语言模型于 2018 年由谷歌团队提出，论文^[41]发表在国际顶级学术会议 NAACL 上。BERT 模型一经提出，就给 NLP 领域带来了巨大的变化，多个子领域包括文本分类^[43]、问答系统^[44]、序列标注^[45]，特别是方面级情感分析领域都涌现出许多基于 BERT 构建的模型，这些模型不断刷新着领域内的记录。

BERT 模型的第一步是对大量的文本库进行预学习，从而获得主模型的参数。如果要将其用于下游任务，那么在主模型之后，就会在其后面接个与其相匹配的模块，之后再将这些预学习的参数运用到主模型上，再对其进行随机初始化，最后再按照这些数据对其进行精细调整。BERT 模型结构如图 2.4 所示。

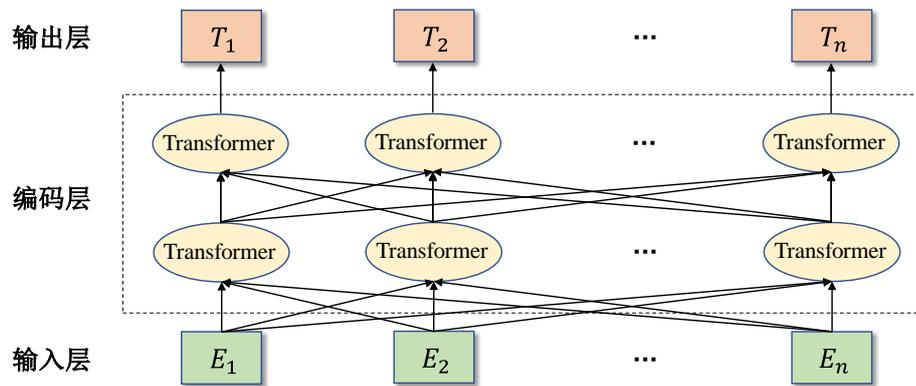


图 2.4 BERT 模型结构

1. 输入层融合向量表示

输入层的融合向量由位置嵌入 (Position Embeddings)、词嵌入 (Token Embeddings) 和分段嵌入 (Segment Embeddings) 相加而成。位置嵌入用于记录输入本文中词与词的位置信息；词嵌入用于将各个词转换成固定维度的向量；分段嵌入用于区分两个句子的向量表示，可用于判断两个句子先后顺序的分类任务。文本信息在经过词嵌入层时需要进行标记，即在输入序列的开头添加 [CLS]，在尾部添加 [SEP]，该做法目的在于为后面的分类任务和划分句子对提供依据。输入层的结构如图 2.5 所示。

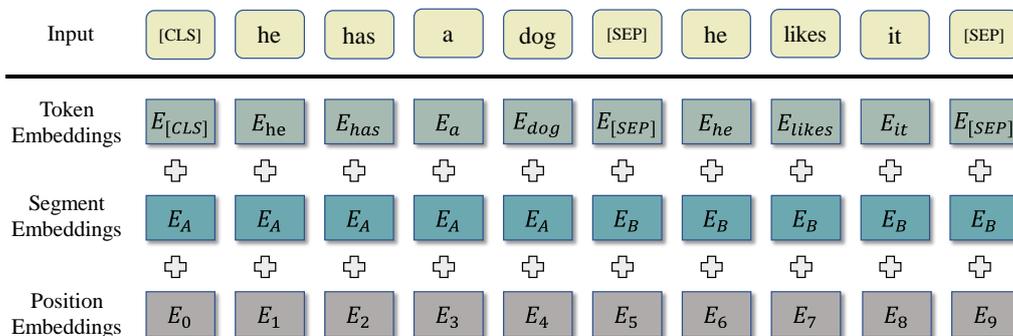


图 2.5 BERT 模型输入表示

2. 编码层 Transformer

Transformer 的编码器结构 (Encoder) 是 BERT 的核心架构, 编码器结构在多头自注意力上添加了三种关键操作。这三种操作分别是残差连接 (Residual Connection)、层标准化 (Layer Normalization) 和线性转换。残差连接用于融合模块的输入信息和输出信息, 进而作为下一个模块的输出, 如此通过改变输入的方式可以更轻松地构建整个输出, 进而降低网络训练复杂度。层标准化将神经网络中的节点进行标准化, 使其具有 0 均值和 1 方差。线性转换则是将得到的本文特征向量做与原始向量长度一致的线性变换

3. BERT 预训练过程

BERT 预训练有两种并行策略, 这两种策略分别为掩码语言模型 (Mask Language Model, MLM) 和下一句预测 (Next Sentence Prediction, NSP)。本文使用 BERT 模型的目的在于获取高质量的词向量表示, 不需要利用 BERT 预测下一句, 因此只介绍 MLM。

BERT 会对输入的单词序列进行一定的信息模糊处理, 即随机抽取出 15% 的词, 并将这些词用符号 [MASK] 替换。在此之后, 该模型会尝试基于上下文关系和未被替换的单词来预测被替换成 [MASK] 的原始单词。研究人员考虑到所有被选中的词如果都采取同一种方式进行遮盖的话, 模型学习到的知识并不丰富, 因此对于被选中的词采取三种处理方式, 一种是进行单词的随机更换, 一种是保持元单词不变, 一种是用 [MASK] 遮盖, 这三种方式按 1: 1: 8 的处理。

基于以上的策略, BERT 预训练语言模型训练时对计算机性能要求较高并且需要耗费大量时间, 但是只要模型训练完成, 便可以直接使用到具体的下游任务中。针对不同的运用场景, 谷歌团队截至目前已经在 Github 代码仓库发布了 24 个精简的 BERT 模型, 研究者们可以根据自身实际使用需求, 选择合适参数的模型进行训练。

2.3 循环神经网络

循环神经网络是一个随着时间推移能够不断重复的结构, 它可以处理具有序列特性的数据。语言文字作为经典的时间序列数据, 因此也能使用循环神经网络

进行序列特征的提取。具体 RNN 结构如图 2.6 所示。

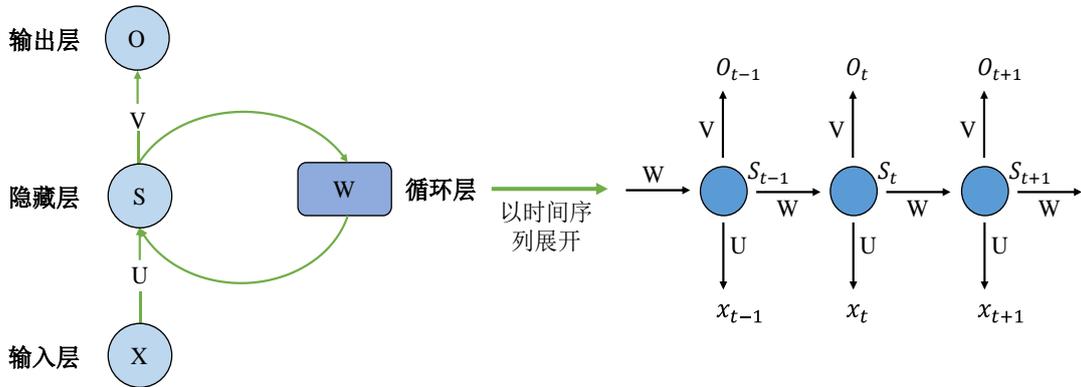


图 2.6 RNN 结构图

图中 X 、 S 、 O 是对应层的向量表示， U 是输入层到隐藏层的参数矩阵， V 是隐藏层到输出层的参数矩阵， W 是每个时间点之间的权重矩阵，下标 t 是某一时刻下的状态。具体实现公式如 (2-8) 和 (2-9) 所示。

$$O_t = g(V \cdot S_t) \tag{2-8}$$

$$S_t = f(U \cdot X_t + W \cdot S_{t-1}) \tag{2-9}$$

在 RNN 的基础上，Hochreiter 等^[46]提出了长短时记忆网络 (LSTM)。普通的 RNN 会记忆所有时刻的信息，而 LSTM 更加智能化，LSTM 引入门控机制可以有选择的过滤信息，对于有用的信息，LSTM 会保留到下一时刻，对于无用的信息，LSTM 则会将其遗忘，以便减少信息储存量。LSTM 结构如图 2.7 所示。

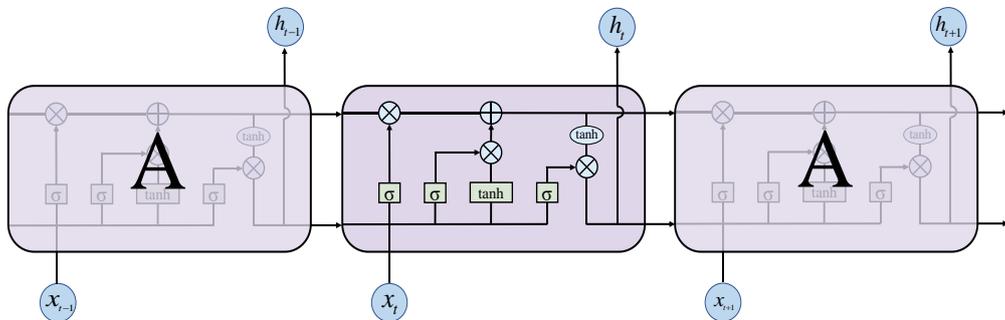


图 2.7 LSTM 结构图

作为 RNN 的一种变体，LSTM 也由结构完全一样的模块组成。LSTM 由 4

个相互作用的神经网络层构成,在图中用绿色矩阵表示,蓝色圆圈表示逐点操作,线条表示向量在节点间的传递。LSTM 模块间的计算公式如下:

$$i_t = \sigma(W_i[x_t \oplus h_{t-1}] + b_i) \quad (2-10)$$

$$f_t = \sigma(W_f[x_t \oplus h_{t-1}] + b_f) \quad (2-11)$$

$$o_t = \sigma(W_o[x_t \oplus h_{t-1}] + b_o) \quad (2-12)$$

$$\tilde{c}_t = \tanh(W_c[x_t \oplus h_{t-1}] + b_c) \quad (2-13)$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tilde{c}_t \quad (2-14)$$

$$h_t = o_t \otimes \tanh(c_t) \quad (2-15)$$

门控机制的遗忘门、输入门、记忆单元和输出门和分别用 f 、 i 、 c 和 o 表示,其中, f 用于确定要删除的数据, i 用于确定要新加入的数据, o 用于确定数据的最终输出。 \oplus 和 \otimes 表示矩阵的拼接和点乘运算,下标 t 为某一时刻对应的状态。 σ 为 sigmoid 函数, \tanh 为双曲正切函数。

相较于传统的 RNN, LSTM 降低了梯度消失和梯度爆炸的风险,并且拥有更强的长距离依赖能力。但是 LSTM 每个模块的 4 个神经网络层决定了 LSTM 在训练时需要花费很多的硬件算力和时间成本,Cho 等人^[47]提出了门控循环单元 (Gate Recurrent Unit,GRU), 论文实验结果表明,在参数调优情况下,GRU 和 LSTM 达到效果相同,但是由于 GRU 只有更新门和重置门,模型结构更简单,花费的时间比 LSTM 更少。GRU 结构如图 2.8 所示。

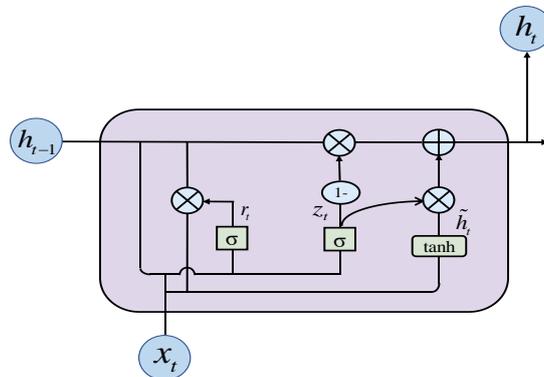


图 2.8 GRU 结构图

GRU 与 LSTM 的输入输出结构一样，通过当前输入的 x_t 和包含之前节点所有信息的隐藏状态 h_{t-1} 相结合，GRU 便可得到当前节点输出的 \tilde{h}_t 和传递给下一个节点的隐状态 h_t 。具体 GRU 实现公式如 (2-16) ~ (2-19) 所示。

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (2-16)$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (2-17)$$

$$\hat{h}_t = \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h) \quad (2-18)$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot \hat{h}_t \quad (2-19)$$

上述公式中 r_t 和 z_t 分别是重置门和更新门计算函数， σ 为 sigmoid 激活函数，该函数可将对应数据输出值控制在 $[0,1]$ 范围内，以此实现门控机制， $(1 - z_t) \odot \hat{h}_t$ 含义是将候选状态 \hat{h}_t 中无效的信息进行忽略。 $z_t \odot h_{t-1}$ 表示的是记忆上一时刻信息中有效的信息记忆。

2.4 卷积神经网络

2.4.1 普通卷积神经网络

卷积神经网络是针对全连接网络的不足而提出的一种前向神经网络结构。卷积神经网络是最早用于手写体字符识别的一种方法，近年来，其在语音识别、目标识别、动作分析等方面取得了长足的进步。基本结构如图 2.9 所示。

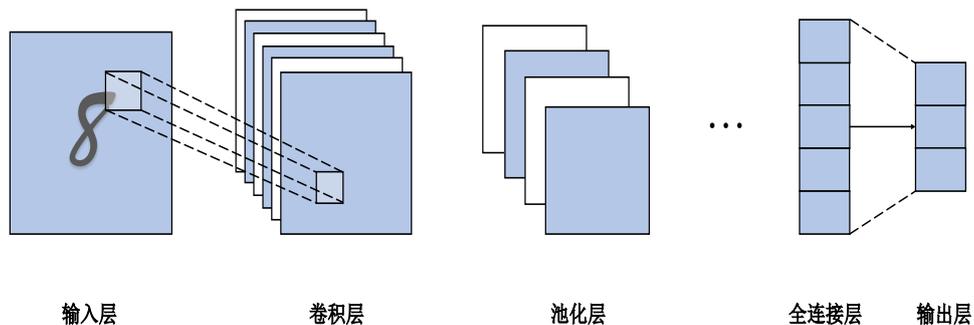


图 2.9 CNN 结构图

输入层即为网络的输入，在 NLP 领域中的输入为词向量矩阵。卷积层由滤波器和激活函数构成，通过卷积操作可以获取输入层的局部区域信息，要设置的超参数包括滤波器的数量、大小、步长等。池化层的主要任务是对从卷积图像中提取出的样本进行压缩，从而实现对重要样本的有效采集。其中，最大池化用来扩大特性，而平均池化则是将特性的平均值取出来。全连接层是将学习到的特征量表达式到类别标记空间的一种方法，其功能相当于一个分类单元。在情感分类任务中，情感类别包括积极、中性和消极三类。

由于卷积神经网络可以有效提取局部特征，对于情感分析任务而言，知道语句中某个重要词语的信息就能准确识别句子情感，因此许多情感分析模型都有卷积神经网络的身影。当卷积神经网络应用到情感分析领域时，CNN 都是采用一维卷积，通过滤波器在由词语组成的向量矩阵上上下下滑动，然后生成临近词语间的 n-gram 特征从而实现特征提取。

2.4.2 图卷积神经网络

卷积神经网络所处理的图片和文字都属于规则性的欧式空间数据，但是在实际生活中，不规则数据占据了大多数，例如知识图谱、拓补结构、分子内部结构等等。利用规则的方法显然不能完全表示非规则数据间的联系，研究人员据此提出了图卷积神经网络（graph convolutional networks, GCN）用于处理能表示为图结构的数据。

图卷积神经网络的实现依赖于卷积算子和池化算子的构建，卷积算子用于学习图数据节点间的局部结构特征，主要用于节点级别的任务，池化算子任务是学习图信息的层级结构表示，主要适用于图级别的任务。文本图作为节点级别的任务，卷积算子的节点间特征表达是最为重要的，而池化算子作用在于感受野的调节和参数简化。目前构建图卷积神经网络有基于谱域和基于空间两种方法。

（1）基于谱域的图卷积神经网络

基于谱域的图卷积神经网络通过卷积定理实现图卷积，卷积定理^[48]指出函数卷积的傅里叶变换是函数傅里叶变换的乘积。其公式如式（2-20）和（2-21）所示。

$$F(f \times g) = F(f) \cdot F(g) \tag{2-20}$$

$$f * g = F^{-1}(F(f) \cdot F(g)) \tag{2-21}$$

其中， F 函数为傅里叶变换， f 和 g 均为原始信号， $*$ 为图卷积算子， \cdot 为乘积算子，式 (2-21) 由式 (2-20) 两端逆变换得到。

对图结构做傅里叶变换则需要在拉普拉斯矩阵上进行，图上的拉普拉斯矩阵如图 2-10 所示。

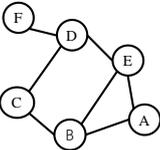
| 原始图数据 | 度数矩阵D | 对角矩阵A | 拉普拉斯矩阵L | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|-------|---------|----|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----|---|---|----|---|----|---|----|---|----|---|---|----|---|----|---|---|---|---|----|---|----|----|----|----|---|----|---|---|---|---|---|----|---|---|
|  | <table border="1"> <tr><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr> <tr><td>2</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>3</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>2</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>3</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>3</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr> </table> | A | B | C | D | E | F | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | <table border="1"> <tr><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr> <tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> </table> | A | B | C | D | E | F | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | <table border="1"> <tr><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr> <tr><td>2</td><td>-1</td><td>0</td><td>0</td><td>-1</td><td>0</td></tr> <tr><td>-1</td><td>3</td><td>-1</td><td>0</td><td>-1</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>2</td><td>-1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>-1</td><td>3</td><td>-1</td><td>-1</td></tr> <tr><td>-1</td><td>-1</td><td>0</td><td>-1</td><td>3</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>-1</td><td>0</td><td>1</td></tr> </table> | A | B | C | D | E | F | 2 | -1 | 0 | 0 | -1 | 0 | -1 | 3 | -1 | 0 | -1 | 0 | 0 | -1 | 2 | -1 | 0 | 0 | 0 | 0 | -1 | 3 | -1 | -1 | -1 | -1 | 0 | -1 | 3 | 0 | 0 | 0 | 0 | -1 | 0 | 1 |
| | A | B | C | D | E | F | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 2 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0 | 3 | 0 | 0 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0 | 0 | 2 | 0 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0 | 0 | 0 | 3 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 3 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | 0 | 0 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| A | B | C | D | E | F | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | 0 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | 0 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | 1 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | 0 | 1 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | 1 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | 1 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| A | B | C | D | E | F | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | -1 | 0 | 0 | -1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -1 | 3 | -1 | 0 | -1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | -1 | 2 | -1 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | -1 | 3 | -1 | -1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -1 | -1 | 0 | -1 | 3 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | -1 | 0 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

图 2.10 图的拉普拉斯矩阵

图中原始图数据之间的连线表示数据间存在联系，将图数据间的关系用矩阵表示可以更加方面的存储图的结构，实现用线性代数方法研究图的问题。度数矩阵 D 用于表示每个点的度数，即由该点发出的边之数量，因此 D 是一个对角矩阵，对角矩阵 A 用于表示节点之间的联系，用“1”代表节点之间存在边连接，用“0”表示节点间没有连接。拉普拉斯矩阵定义为 $L=D-A$ ，从图中可以看出 L 是一个对称的矩阵，并且每一行每一列的元素之和都为 0。

若用 U 表示图上拉普拉斯矩阵的 n 维特征向量， U^T 为特征矩阵 U 的转置矩阵， \odot 表示 Hadmard 算法。则两个原始信号 f 和 g 傅立叶变换可以重新表示为式 (2-22)：

$$f * g = U \left((U^T f) \odot (U^T g) \right) \tag{2-22}$$

进一步的，定义一个对角矩阵 $g_\theta = U^T y$ 可以将 Hadmard 算法转为矩阵乘法，则式 (2-22) 又可表示为式 (2-23)。

$$f * g = U g_\theta U^T x \tag{2-23}$$

式 (2-18) 中 g_θ 就是图卷积神经网络的卷积核。从上述计算过程可以看出基于谱域的方法可以虽然可以实现图数据的处理, 但是图的拉普拉斯矩阵在计算量上偏大, 并且计算复杂度随着图数据的增大呈几何倍数增长, 无法应用到图节点数量多的情景。

(2) 基于空间的图卷积神经网络

为改进基于谱域的图卷积方法缺陷, Kipf 等人^[49]提出了基于空间的图卷积神经网络模型, 模型通过聚合每个中心节点和其邻近节点实现图卷积, 并在每一层隐藏层后都进行归一化处理, 有效避免了过拟合现象。该模型用切比雪夫一阶展开项作为图神经网络的卷积核, 以此避免了计算量大的问题。对于 GCN 层数为 n 时的输出表达式如 (2-24) 所示:

$$Z^{(n+1)} = F\left(\theta D^{-\frac{1}{2}} A D^{-\frac{1}{2}} Z^{(n)}\right) \quad (2-24)$$

其中 A 表示图的邻接矩阵, D 表示图的度数矩阵, θ 表示切比雪夫系数的向量, F 表示 ReLU 激活函数。具体计算流程可从图 2.11 看出。

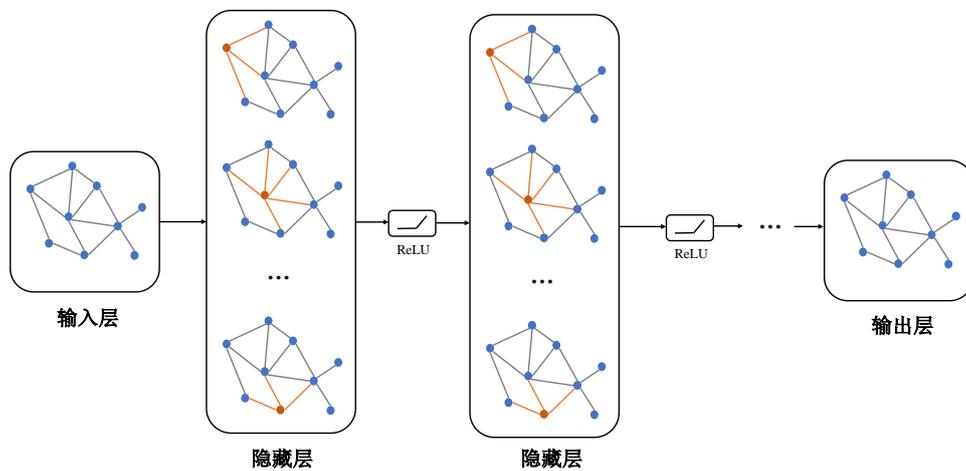


图 2.11 GCN 计算流程

在情感分析领域中, 图神经网络中的节点可以表示词向量, 节点间的连边可以用来表示词之间的共现关系。相较于之前采用普通卷积神经网络来处理序列化模型, 图卷积神经网络能够挖掘更深层次的非线性的语义信息, 进一步提升模型特征提取能力。

2.5 注意力机制

注意力机制源于人类的视觉原理，当一幅图像摆在我们眼前时，我们往往会将注意力放在图像中某个突出显眼的部分，而忽略其他区域信息。注意力机制依赖Encoder-Decoder通用框架实现，Encoder-Decoder框架实质是一个知识转换再输出的过程，例如我们在接触新的理论知识时，需要将晦涩难懂的知识进行理解、掌握，然后再通过自己的表述将知识传授给其他人。

整个框架的核心为Encoder和Decoder，框架首先将任意一段文字、图像或者音频信息作为输入信息，这些信息在Encoder模块进行编码操作，转成包含输入信息的向量，接着进入Decoder模块根据任务需求输出最终结果。两个模块内部可以根据任务具体情况自由选择神经网络，所以基于Encoder-Decoder框架可以设计出针对领域需求的的应用算法。Encoder-Decoder框架如图2.12所示。

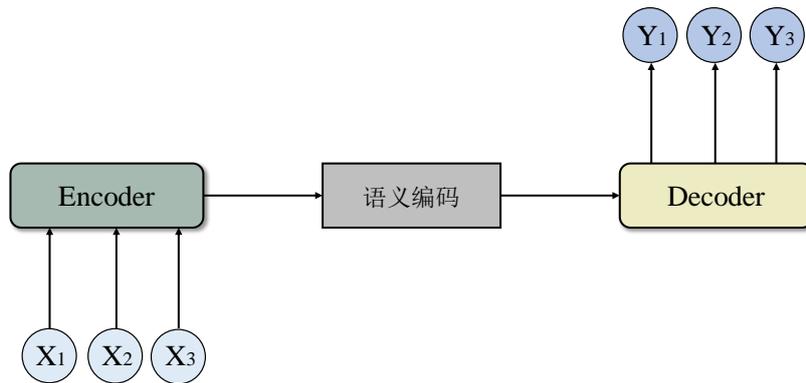


图 2.12 Encoder-Decoder 框架图

Encoder-Decoder框架是一个端到端的学习算法，在自然语言处理领域最先用于机器翻译任务，基于此架构建立的模型也被称为Sequence to Sequence模型(Seq2Seq)。以机器翻译为例，将待翻译的语言作为输入序列 $X = \{x_1, x_2, \dots, x_n\}$ ，Encoder部分对输入序列 X 进行语义编码转成 $C = f(x_1, x_2, \dots, x_n)$ ，接解码器对语义编码 C 进行解码操作，最终输出将目标语言 $Y = \{y_1, y_2, \dots, y_m\}$ 。

Encoder-Decoder框架能有效的表示和输出信息，但也存在一定的缺点。在进行文本处理任务时，如果输入的文本太长，语义编码 C 的无法根据信息量大小自

动调整，导致新的输入信息覆盖旧的输入信息，造成信息丢失问题。2014年，德国雅各布大学的Bahdanau提出了一种注意力模型用于机器翻译任务^[50]，图2.13为融入注意力机制的模型结构图。

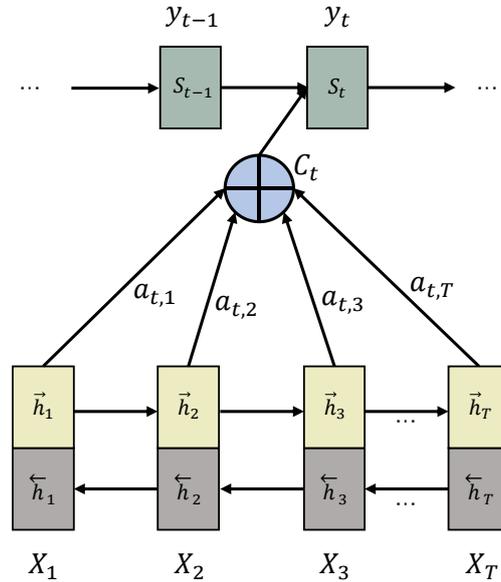


图2.13 引入注意力机制的模型结构图

图中编码器部分由前向LSTM和后向LSTM拼接而成，中间向量 C 由注意力权重 $a_{t,j}$ 和当前时刻隐藏层向量 h_j 加权求和得到，其中 j 为编码器的词序号， t 为解码器生词的词序号，具体实现可由公式（2-25）和（2-26）可得。

$$y_t = g(y_1, y_2, \dots, y_{t-1}, C_t) \quad (2-25)$$

$$C_t = \sum_{j=1}^T a_{t,j} h_j \quad (2-26)$$

在进行情感分析任务时，海量的文本数据中对任务起到关键作用的往往只占极少部分。虽然注意力机制需要对输入进行遍历，会提升模型训练时间，但是这并不影响注意力机制在情感分析领域的流行。

2.6 依存句法分析

句子的词语与词语之间存在一定的联系，依存句法分析就是用确切的关系将

词之间的联系表示出来。具体来说，先要制定好词之间的语法规则，根据规则判断出动作发出者和动作接受者，动作发出者在两者间处于支配者（Head），动作接受者在两者间处于被接受者（Dependency），两者形成的联系就是依存关系，经过多个词的关系标注，这些关系互相交织就形成了图结构。

1959年，著名语言学家泰尼埃尔提出了一种具有普适性的结构句法一般理论，该理论提出每个句子都存在核心词语，这个核心词语不受句子中其他词语的约束，并且在句子中担任了约束其他词语的作用。在此基础上，计算机语言学家Robison提出了依存句法的四条公理^[51]，四条约束性公理奠定了依存句法分析的基础。研究人员根据公理开始制定标注规格实现对句子成分的关系标注，目前常用的依存句法分析工具有百度的DDParser (Baidu Dependency Parser)和哈工大的LTP (Language Technology Platform)，依存句法标签体系如表 2.1 所示：

表 2.1 依存句法分析关系标注表

| 关系类型 | 标注 | 描述 | 举例 |
|-------|-----|-----------------------|--------------|
| 核心关系 | HED | head | 句子的核心词 |
| 主谓关系 | SBV | subject-verb | 我爱中国（我←爱） |
| 间宾关系 | IOB | indirect-object | 我借他一支笔（借→他） |
| 动宾关系 | VOB | verb-object | 我借他一支笔（借→笔） |
| 状中结构 | ADV | adverbial | 特别努力（特别←努力） |
| 定中关系 | ATT | attribute | 大西瓜（大←西瓜） |
| 介宾关系 | POB | preposition-object | 在学校里（在→里） |
| 动补结构 | CMP | complement | 看完了课本（看→完） |
| 前置宾语 | FOB | fronting-object | 他什么菜都吃（菜←吃） |
| 左附加关系 | LAD | left adjunct | 江河与湖泊（与←湖泊） |
| 右附加关系 | RAD | right adjunct | 学子们（学子→们） |
| 独立结构 | IS | independent structure | 两个单句在结构上相互独立 |
| 并列关系 | COO | coordinate | 我和你（我→你） |
| 兼语 | DBL | double | 他送我回家（送→我） |

基于以上句法标注规则,可以用带有方向的弧线表示给定的语句中词语间的关系。以“公司总经理张三视察工作环境时提出,鼓励员工积极提出新想法。”为例,“张三”是例句中谓语动词“提出”的主语,“提出”是整个句子的核心词,两者存在主谓关系(SBV),“公司总经理”为“张三”的修饰语,“提出新想法”是“鼓励”的宾语。依据上述关系,将关系的支配者作为有向弧的出发点,我们可以得到依存关系图如图 2.14 所示。

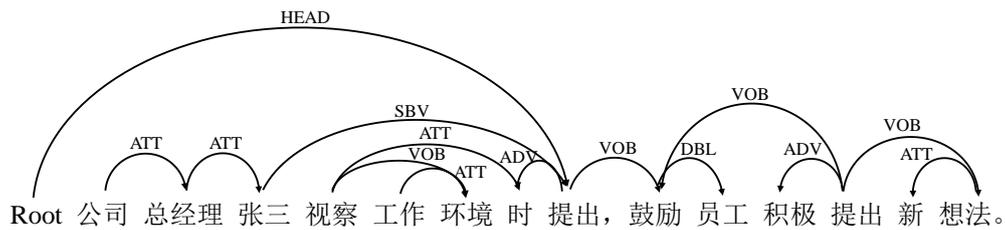


图 2.14 依存句法分析图

无论是英语、法语、中文或者其他语种,句子中词语间的依赖关系是普遍存在的,因此依存句法分析是自然语言里的通用分析工具。目前常用的依存句法分析工具有 Python 文本预处理库 spaCy¹和百度自研的 DDParse²等。

2.7 特征融合方法

在 NLP 领域,如何深度挖掘本文潜在的特征是准确高效完成任务的关键,而如何将学习到的特征进行融合同样也不可忽视。当前特征融合的基本方法主要分为两类,即:按点逐位相加(point-wise addition, add)和向量拼接(concatenate, concat)。其中前者属于线性处理方式,后者则属于非线性处理方式。两种途径有相同之处,也有不同之处,下面将做一个简单的讲述。

(1) add

按点逐位相加也叫逐个位相加,用数学表达为: $v_1 \in \mathbb{R}^n$, $v_2 \in \mathbb{R}^n$, 为得到融

¹ <http://pypi.org/project/spacy/>

² <https://github.com/baidu/DDParser>

合向量 v ，直接进行对应元素的相加，既是：

$$v = v_1 + v_2 = \{x_i | x_i = v_1[i] + v_2[i], i = 1, \dots, n\} \quad (2-27)$$

在这种特征融合条件下，要求两个向量的维度全部一致，如果是不同维度，则可以通过线性变换转换成同维向量，接着再进行逐位相加操作。

(2) concat

向量拼接是特征融合时最为普遍的办法，该方法本质为多个特征的堆叠。在向量拼接下，要把两个矩阵向量在某个维度叠加起来，在这个连接的维度上两者可以不同，但是在其他维度上必须相等。叠加后，某个维度会增加，表现为两个矩阵上的某个维度之和。

假设特征 1 的维度为： $B_1 * C_1 * H_1 * W_1$ ，特征 2 的维度为 $B_2 * C_2 * H_2 * W_2$ ，则两者在 C_1 维度的 concat 的结果为 $B_2 * (C_1 + C_2) * H_2 * W_2$ 。以 $26 * 26 * 256$ 的特征为例，两种特征融合方式可从图 2.15 看出。

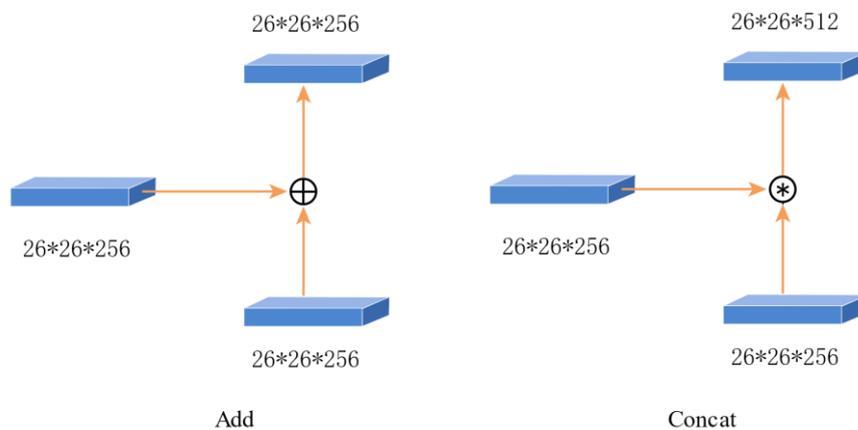


图 2.15 特征融合方式图

concat 的每一个通道都与相应的卷积核相关联，而 add 方法是将相应的特征图叠加起来，然后进行下一步的卷积运算，就像是给它加上了一个先验，所以，后者可以当做前者的一种特别形态，它是一种增加了先验知识的特征融合方式。另一方面，add 会降低数据的维数和丢失一些信息，在后续处理过程中，add 的两个分量会保持一致，导致参数不能更新。而 concat 会容许不同的参数，所以在 NLP 领域中大多采用 concat 操作。

2.8 本章小结

本章主要对情感分析领域的相关理论和技术做出介绍,首先对情感分析作出概述,对基于 Word2Vector、GloVe 和 BERT 的词向量表示方法进行详细阐述;接着分别对循环神经网络、卷积神经网络、注意力机制、依存句法分析和特征融合方法作出进一步介绍。其中循环神经网络重点讲述了长短时记忆网络,卷积神经网络讲述了卷积原理和图卷积神经网络,依存句法分析介绍了句子间普遍存在的依赖关系,特征融合介绍了两种常用的融合方式,为下一章构建情感分类模型奠定了理论和技术基础。

3. 基于双通道信息融合的方面级情感分类模型

基于句法关系的图卷积网络模型能够很好的表示文本这类非欧式数据，因此被广泛应用到方面级情感分类任务中，但是这类模型只考虑到了文本信息的句法关系，忽略了其他同样能对情感分类有帮助的特征。本文提出了一种基于双通道信息融合的方面级情感分类模型（An aspect-level sentiment classification model based on two-channel information fusion, GCNN），模型在基于句法关系的图卷积网络模型基础上额外增加一通道用于提取文本上下文语义特征，接着将双通道的句法和语义特征融合，达到特征加强的目的，进而提升模型情感极性判别能力。GCNN模型结构如图3.1所示。

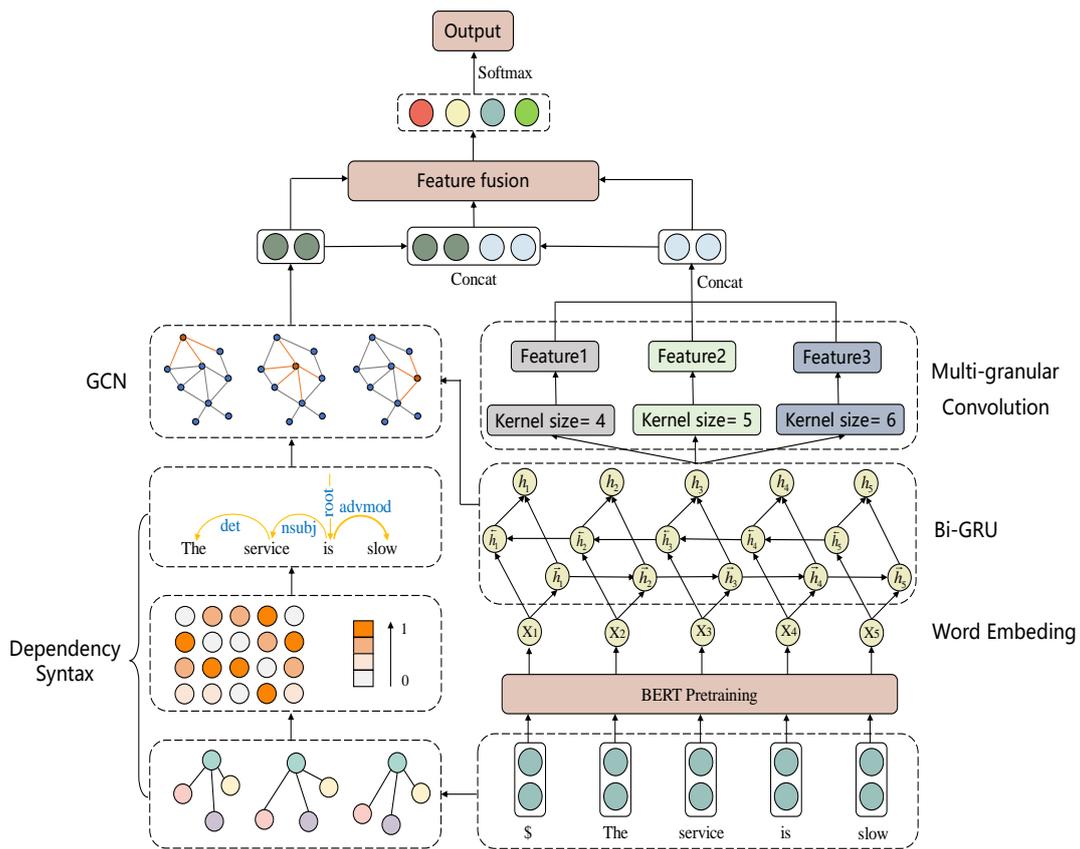


图 3.1 GCNN 结构图

模型可分为基于语义关系的多粒度卷积网络（Multi-convolutional neural networks based on Semantic information, MulCNN）和基于句法关系的图卷积神经

网络 (Graph Convolutional Networks based on Dependency Syntax, DepGCN)。

MulCNN由BERT预训练模型、词嵌入层 (Word Embedding) 层、Bi-GRU层、多粒度卷积层 (Multi-granular Convolution) 组成, 该通道用于学习句子的语义特征信息。DepGCN由BERT预训练模型、词嵌入层、Bi-GRU层、句法关系层 (Dependency Syntax)、GCN层组成, 该通道用于学习句子句法关系特征。两通道的特征在特征融合层 (Feature fusion) 进行注意力计算, 得到包含语义及句法信息的融合特征, 最后在输出层输出情感预测值。

3.1 词嵌入层

词嵌入层用于将输入序列转换成后续深度学习结构可以识别的词向量。一方面本层采用 BERT 预训练模型获取词向量表示, BERT 模型版本选择基础的 BERT-Base, Uncased 作为预训练模型, 模型包含 12 个堆叠的 Transformers, 768 个隐藏层和 12 个头数的多头自注意力机制。另一方面, 为进一步丰富语义特征, 在词向量中融入了词性信息和位置信息。

在本文情感分类任务中, 词性为形容词、副词和语气词的词语往往承担着表达个人观点的任务, 本模型在词嵌入层引入词性信息, 以此加强模型对输入序列的理解能力。词性标注 (Part-of-Speech Tagging) 用于为输入文本中的每个词性标注词分配词性标记, 标记算法的输入是一系列标记化的单词和标记集, 输出是一系列标记。对于输入序列 $\{t_1, t_2, \dots, t_n\}$, 采用词性标注工具进行标注后生成标记集 $\{pos_1, pos_2, \dots, pos_n\}$, 其中 $pos_i \in \mathbb{R}^{d_v}$, d_v 为词嵌入维度。

对于给定的输入序列, 情感词往往出现在方面词附近, 而位置信息可以表示词之间的位置关系。位置嵌入 (position embedding) 的方式有相对位置嵌入和绝对位置嵌入 (Sinusoidal Position Embedding, PE) 两种方式, GCNN 模型采用绝对位置编码, 编码公式如式 (3-1) 和 (3-2) 所示。

$$p_{(i,2j)} = \sin\left(\frac{i}{10000^{\frac{2j}{d_v}}}\right) \quad (3-1)$$

$$P_{(i,2j+1)} = \cos\left(\frac{i}{10000^{\frac{2j}{d_v}}}\right) \quad (3-2)$$

其中 i 为目标词在输入序列的位置， $j \in [0, d_v / 2]$ 为整数计数值。

将预训练好的词向量、嵌入位置信息和词性信息的向量三者融合得到新的词向量，该词向量作为词嵌入层的最终向量输出参与后续训练，最终向量序列 $X = \{x_1, x_2, \dots, x_n\}$ ， x_i 计算公式如 (3-3) 所示。

$$x_i = t_i \oplus pos_i \oplus p_i \quad (3-3)$$

3.2 Bi-GRU 层

考虑到 Bi-LSTM 在参数数量和模型训练时间上要多于 Bi-GRU，且两者在参数调优的情况下特征提取能力一致，GCNN 模型选择 Bi-GRU 作为特征提取层。Bi-GRU 层由两个完全相同、方向相反的 GRU 组成，双向门控循环单元结构简单，模型训练时间短，能够有效避免循环网络的梯度爆炸和梯度消失问题。使用 Bi-GRU 可获得每个词前向和后向的隐藏输出表示，将学习到的两个方向的输出表示联合可以得到当前位置的状态，从而提取到文本更深层次的特征。模型更新公式如式(3.4)~(3.6)所示。

$$\vec{h}_t = GRU(x_t, \vec{h}_{t-1}) \quad (3-4)$$

$$\overleftarrow{h}_t = GRU(x_t, \overleftarrow{h}_{t-1}) \quad (3-5)$$

$$h_t = w_t \vec{h}_t + v_t \overleftarrow{h}_t + b_t \quad (3-6)$$

其中，GRU 函数为非线性的数据标准化激活函数， x_t 为词嵌入的输出向量， \vec{h}_t 表示在 t 时刻词向量经过前向 GRU 编码后的隐层状态， \overleftarrow{h}_t 表示在 t 时刻词向量经过后向 GRU 编码后的隐层状态， w_t 表示在 t 时刻前向 GRU 相对应的权重，而 v_t 表示在 t 时刻经过后向 GRU 相对应的权重， b_t 表示 t 时刻双向 GRU 的综合数据偏置； h_t 则为联合双向门控循环单元计算得到最终的序列输出。

3.3 多粒度卷积层

在图像处理领域，研究人员将图像分为多个通道，每个通道都采取多个维度的特征提取方法，以便得到全方位的图像特征。在 NLP 领域中，文字的信息由二维向量表示，但特征表示仍是一维的，此时若对向量做二维卷积便失去了意义。比如“论文”对应的向量 $[0, 0, 0, 1]$ ，按窗口大小为 1×2 从左到右滑动依次得到 $[0,0]$, $[0,0]$, $[0, 1]$ 三个向量，但向量对应的都是“论文”这个词汇。这种横向遍历并没有提取到词语间的特征，因此情感分析任务中只需要用不同粒度的卷积核在纵向上从上到下遍历词向量从而丰富 N-gram 特征即可。

多粒度卷积层将 Bi-GRU 的输出 $H = \{h_1, h_2, \dots, h_n\}$ 作为输入，设卷积核 $W \in \mathbb{R}^{h \times d}$ ，其中 h 为滑动窗口大小， d 为 Bi-GRU 层输出向量的维度，本文将 d 设为 1536。

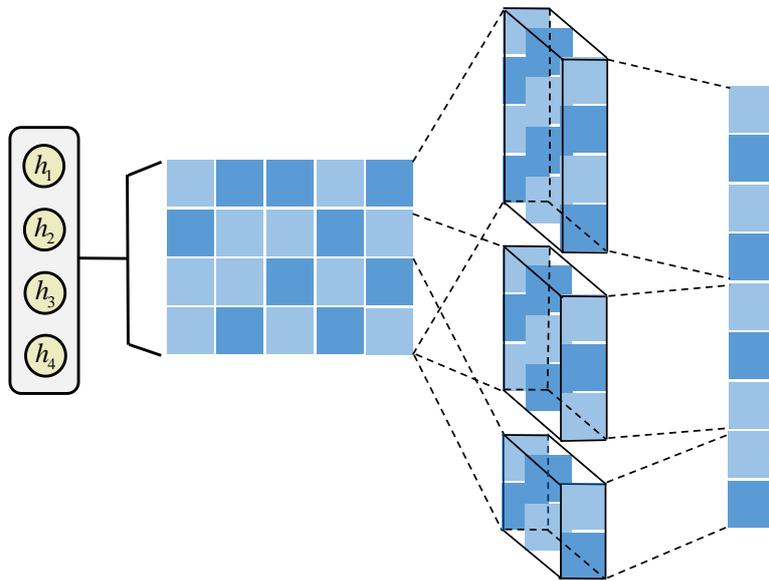


图 3.2 多粒度卷积流程图

卷积核 W 的卷积流程如图 3.2 所示，图中矩阵为学习了序列特征的词向量表示，由于矩阵的宽度就是词向量的宽度，因此不需要 padding 操作。卷积核 W 从矩阵第一行开始依次向下滑动，同子矩阵 $k_{i:i+h-1}$ 做点积运算后得到取局部特征 z_i ，其计算公式如式(3-7)所示。

$$z_i = f(k_{ii+h-1} \odot W + b) \quad (3-7)$$

式中， f 为 ReLU 激活函数， b 为偏置向量。

使用不同粒度的卷积核在相同的词向量矩阵上做自上而下的遍历操作可以得到不同特征的矩阵，通过卷积运算将特征矩阵映射为宽度为 1 的矩阵，这个映射过程就是特征提取过程，其结果就是特征图。不同卷积核得到的特征图蕴含着本文序列中不同位置的 N-gram 特征，这些特征进一步丰富了上下文局部特征。

3.4 句法关系及图卷积网络层

对于给定的评论语句，表达情感态度的词语在句中往往担任修饰语或者在距离上情感词更靠近方面词，这些修饰信息可以通过依存句法分析识别依赖关系得到。依存句法树将每个词都通过节点之间的连边建立了联系，对于无法直接建立联系的词则可以通过关联节点建立间接联系，这就解决了一些评论文本因太长而无法建立联系的问题。本文使用 2019 年 Mrini 等人^[52]提出的句法解析模型 Label Attention Layer Parser (LAL-Parser)，该句法解析模型能够充分利用 constituent 和 head 的 label 信息，从而在句法依存分析中有更好的表现。

句法分析是所有工具性 NLP 任务中较为高级、复杂的一种任务，句子的句法联系可通过句法依存树表示。例如句子“Delicious fruits but the price was expensive!”，通过 LAL-Parser 可以得到一颗依存句法树如图 3.3 所示。

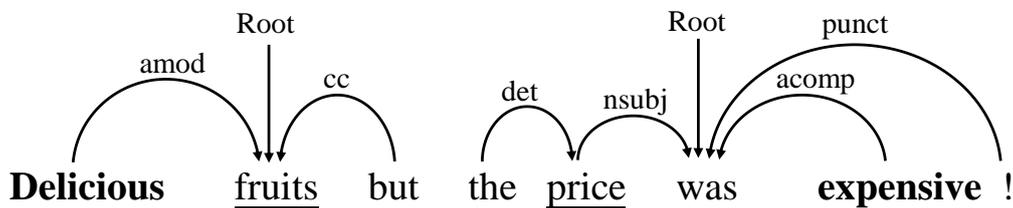


图 3.3 依存句法树关系图

图中关系标签 amod 表示形容词，Root 所指的词被称为根，该成分不依赖于其它成分，cc 表示并列关系，det 为限定修饰词，nsubj 为名词主语，punct 为标

点, **acomp** 为形容词的补充。从树形结构图中的依赖关系可以看出方面词“**fruits**”对应的情感词为“**delicious**”, 两者存在直接的联系, 而方面词“**price**”对应的情感词“**expensive**”并不存在直接依赖关系, 两者通过“**was**”建立了间接联系。这些句法结构信息在深度学习模型中通过邻接矩阵表示, 邻接矩阵的行和列都是输入语句, 因此该矩阵为一个对角线为 1 的稀疏矩阵。矩阵内的数字只有“0”和“1”, 其中“1”表示行中的词语与列中的词语存在依赖关系, “0”表示对应词语无联系。上述例句的邻接矩阵可用表 3.1 所示。

表 3.1 句子的邻接矩阵

| 邻接矩阵 | Delicious | <u>fruits</u> | but | the | <u>price</u> | was | expensive | ! |
|------------------|------------------|---------------|-----|-----|--------------|-----|------------------|---|
| Delicious | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| <u>fruits</u> | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| but | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| the | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| <u>price</u> | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| was | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| expensive | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| ! | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

图卷积神经网络可以学习到非欧几里得结构的空问特征, 更好的将词之间的依赖关系整合到模型中, 因此本文在句法关系层后构建到图卷积网络层, 用于学习到包含深层句法关系的特征表示。在图卷积网络层中, 每个节点都能够通过相邻节点的信息来更新自身节点的表示, 本文将依存句法信息层的邻接矩阵作为语法编码, 将 Bi-GRU 层的输出向量 $H = \{h_1, h_2, \dots, h_n\}$ 作为图卷积网络层的初始节点表示, 将图卷积神经网络层数 L 设置为 3, 每一层节点更新公式如式 (3-8) 所示。

$$h_i^l = \sigma \left(\sum_{j=1}^n A_{ij} W^l h_j^{l-1} + b^l \right) \quad (3-8)$$

其中 n 为输入文本的长度， $l = \{1, 2, 3\}$ ， A_{ij} 为从句法依赖解析器中得到的邻接矩阵， $1 \leq i \leq n$ 和 $1 \leq j \leq n$ 是节点的特征数。 h_j^{l-1} 是 Bi-GRU 层输出向量在图卷积神经网络 $l-1$ 层的节点结果表示，用于网络在 l 的输入信息表示，当 $l=1$ 时， h_j^0 为 Bi-GRU 层的输出向量 H 。 W^l 和 b^l 是第 l 层图卷积神经网络的权重矩阵和偏置向量， σ 为 ReLU 激活函数。将所有节点聚合最终得到图卷积网络的输出表示 $H^{dep} = \{h_1^{dep}, h_1^{dep}, \dots, h_n^{dep}\}$ 。图 3.4 为 GCN 的一个示例。

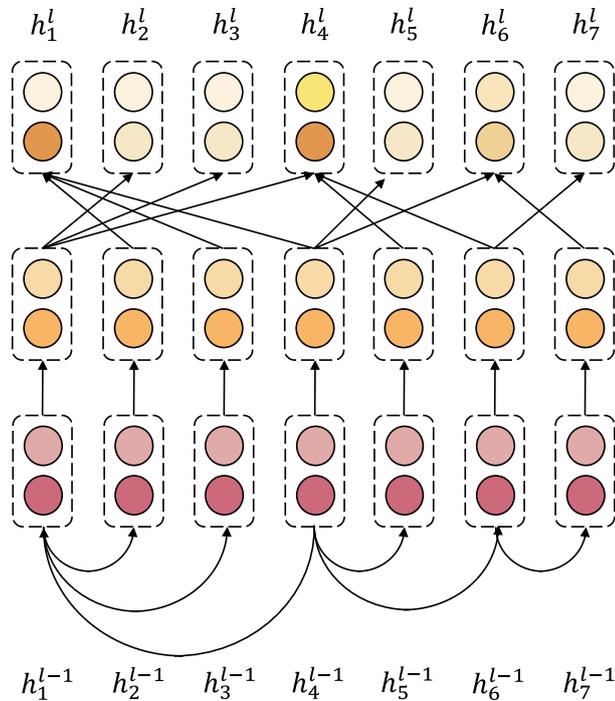


图 3.4 GCN 示例

3.5 特征融合层

图卷积网络层和多粒度卷积层提取的特征并不相同，如何有效利用这两个层提取到的特征十分关键，由于 Concat 操作只是简单的将特征进行堆叠，在特征融合时并没有将句法关系特征和语义信息特征的关联性进一步挖掘。本文基于注意力机制构建了特征融合层，对学习到的两个特征进行信息提取并得到注意力权

值，这些融合特征相较于 **Concat** 操作得到的融合特征有着更丰富的表征信息。

注意力机制可以有选择性的关注句子中对情感分析任务有贡献的单词，提升其在句中的权重值，忽略没有帮助的信息。例如对于“The food is delicious.”，“food”作为本句的方面词，在特征融合层时就会在向量中编码进更多的“food”信息。其结构如图 3.5 所示。

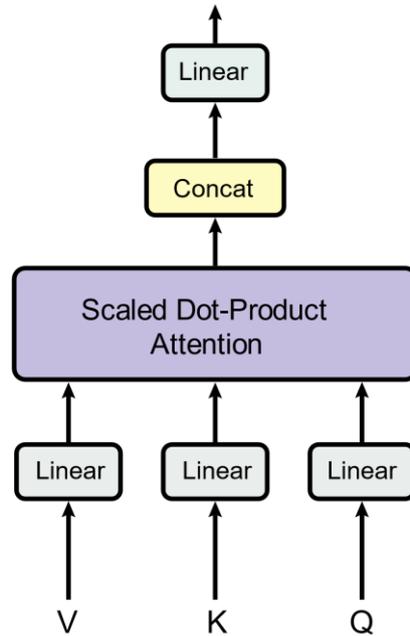


图 3.5 注意力机制结构图

注意力机制要确定给定目标序列的查询 (query, Q)，关键字 (key, K)和关键字权重 (value, V)三个指定的输入，然后通过计算公式得到注意力结果，在本文中，将句法关系层得到的向量作为 K ，将多粒度卷积层拼接的向量作为 Q ，将两个层拼接的向量作为 V 。计算公式可表示为： $F=F(Q, K^T)$ ，用于计算 Q 和 K 的相似度， s 为计算出来的相似度值。 W 为对应的注意力系数。具体计算过程可分为三个阶段：

阶段 1：根据 Q 和 K 计算二者的相似度，计算公式如 (3-9) 所示。

$$s = F(Q, K_1) = \frac{Q \cdot K^T}{\sqrt{d_k}} \quad (3-9)$$

阶段 2: 对阶段 1 得到的注意力得分进行数值转换, 一方面可以进行归一化, 另一方面可以用 softmax 函数的特性突出重要元素的权重。计算公式如 (3-10) 所示。

$$W = \text{softmax}(s) \quad (3-10)$$

阶段 3: 根据权重系数 W 对关键字权重 V 进行矩阵乘法, 计算公式如(3-11)。

$$\text{Attention}((K, Q), V) = W \cdot V \quad (3-11)$$

3.6 输出层

GCNN 模型采用一个全连接层作为输出网络进行情感分类。将带有权重和融合语义及句法特征的向量 O 作为全连接层的输入, 然后利用 softmax 分类算法将该向量映射到与预设情感极性空间维数相同的概率空间, 最后得到模型的预测结果 \hat{y} , 计算公式如式 (3-12) 所示:

$$\hat{y} = \text{softmax}(WO + b) \quad (3-12)$$

其中, W 代表权重, b 代表偏置。模型通过最小化交叉熵损失函数对来优化训练。用 C 表示情感类别, N 表示训练数据量, y 表示真实标签, 损失函数计算公式如式 (3-13) 所示:

$$\text{loss} = -\sum_{n=1}^N \sum_{i=1}^C y_n^i \log \hat{y}_n^i \quad (3-13)$$

3.7 本章小结

本章提出了一种基于双通道信息融合的方面级情感分类模型。从层次结构上来看, 本章用 6 个小节分别对词嵌入层、Bi-GRU 层、多粒度卷积层、句法关系及图卷积网络层、特征融合层和输出层进行介绍。从模型功能上来看, 该模型由 DepGCN 和 MulCNN 两个通道组成, DepGCN 用于学习句子的句法关系, MulCNN 用于学习句子的语义信息, 接着在特征融合层进行句法信息和语义信息的向量融合, 得到具有丰富表征信息的向量表示, 实现情感分类任务。

4. 实验与分析

4.1 数据集介绍

为验证上一章 GCNN 情感分类模型的有效性,本文在 SemEval 2014 Task4^[53] 中的餐馆评论 (Restaurant) 及笔记本电脑评论 (Laptop) 数据集, Twitter 短文本评论数据集^[54]上实验分析。数据集中每条评论语句都由目标方面和方面极性组成, Restaurant 数据集目标方面包括餐厅服务、菜品味道、餐厅位置等, Laptop 数据集目标方面包括电脑价格、电脑质量、使用体验、售后服务等。两个数据集都以正向评论居多,但 Laptop 数据集的情感表达方式更为隐秘,且数据量相对更小。Twitter 数据集共有六千多条评论数据,每个评论只标记了一个方面词且大多都为中性,相比 SemEval 2014 数据集, Twitter 样本质量更低,在情感分析时难度也相应加大,下图为 Restaurant 数据集的样例展示。

```

▼<sentence id="870">
  <text>In addition, the food is very good and the prices are reasonable.</text>
  ▼<aspectTerms>
    <aspectTerm term="food" polarity="positive" from="17" to="21"/>
    <aspectTerm term="prices" polarity="positive" from="43" to="49"/>
  </aspectTerms>
  ▼<aspectCategories>
    <aspectCategory category="food" polarity="positive"/>
    <aspectCategory category="price" polarity="positive"/>
  </aspectCategories>
</sentence>

```

图 4.1 Restaurant 数据集样本

为更清晰了解方面级情感分析任务,以 Laptop 数据集样本“I charge it at night and skip taking the cord with me because of the good battery life”为例。例句方面词为“cord”和“battery life”,可以看出两者对应的情感极性分别为“netural”和“postive”。但是相同情感词语在不同语境中经常出现情感态度不一致的情况,例如“电脑电池用了很长时间”和“排队就餐等了很长时间”,两个句子都用“长”表示了情感态度,但是当方面词不同时,表达的情感态度完全相反。当方面词为电脑电池时,情感词“长”意味着耐用,表示积极的态度;当方面词为排队时间时,“长”意味着排队花费了很长时间,表示消极的态度。如何区别语义在语境中要表达的

意思就尤为重要，因此本文选取的三个数据集可用于验证模型的性能。

由于 SemEval2014 Task4 标准数据集包括积极 (Positive)，中性 (Neutral)，消极 (Negative)，矛盾 (Conflict) 四种标签，而矛盾这一类别的样本量大幅少于其他类别的样本量，为了防止出现数据不平衡的问题，后续测试中将删除矛盾这一类别的数据。具体的数据统计如表 4.1 所示。

表 4.1 数据集统计

| 数据集 | Positive | | Neutral | | Negative | |
|------------|----------|------|---------|------|----------|------|
| | Train | Test | Train | Test | Train | Test |
| Laptop | 976 | 337 | 455 | 167 | 851 | 128 |
| Restaurant | 2164 | 727 | 637 | 196 | 807 | 196 |
| Twitter | 1507 | 172 | 3016 | 336 | 1528 | 169 |

4.2 评价指标

情感分析任务可看为一个二分类问题，因此可以将样例划分为真正例 (True Positive, TP)、假正例 (False Positive, FP)、真反例 (True Negative, TN)、假反例 (False Negative, FN) 四种情形。分类结果的混淆矩阵 (Confusion Matrix) 如表 4.2 所示。

表 4.2 混淆矩阵表

| 真实情况 | 预测结果 | |
|------|------|----|
| | 正例 | 反例 |
| 正例 | TP | FN |
| 反例 | FP | TN |

其中真正例表示标签为正类的样本在模型预测时也为正类，假正例表示标签

为正类的样本在模型预测时为负类，真反例表示标签为负类的样本在模型预测时也为负类，假反例表示标签为负类的样本在模型预测时为正类。令 TP、FP、TN、FN 分别表示其对应的样例数，则有 $TP+FP+TN+FN=$ 样例总数。

基于以上参数，用预测正确的正样本占有所有预测为正样本的比例表示准确率 (Precision)，具体计算公式如 (4-1) 所示。

$$Precision = \frac{TP}{TP + FP} \quad (4-1)$$

针对原来的样本，用召回率 (Recall) 表示样本中的正例有多少被预测正确，定义公式如 (4-2) 所示。

$$Recall = \frac{TP}{TP + FN} \quad (4-2)$$

针对所有的样本，本文使用精确度 (Accuracy) 表示分类正确的样本数占样本总数的比例。Accuracy 反应了模型将正样本预测为正例和负样本预测为负例的能力，可以反映出模型的优劣。

理想状态下，Precision 和 Recall 两个指标都高时则说明预测效果越好，但是在实际计算中很难出现这种情况。比如在极端情况下，如果正例只有一个且被预测正确，那么 Precision 为 100%，但 Recall 就很低；而如果把所有结果都返回，比如 Recall 是 100%，但是 Precision 就很低。因此单一使用准确率或者召回率并不能完全评价模型好坏，因此本文选择综合评价指标 F1-Score 来平衡 Precision 和 Recall 相矛盾的情况。F1 值计算公式如下：

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (4-3)$$

由于样本数量不多，为了更好地利用训练集数据，在实验中采用五折交叉验证法，即将训练集划分为五个子样本，将其中 4 份作为训练集，另外一份作为验证集，如此重复五次，最后对五次结果求平均作为最终结果。

4.3 实验环境与超参数配置

4.3.1 实验环境

本文实验环境硬件方面相关配置信息为：Ubuntu 20.04 系统、CPU 为 I9-

10900X、显卡为 NVIDIA GeForce RTX 3080 TI、12G 显存；软件环境方面，使用 Python 版本为 3.8，深度学习框架 Pytorch 版本为 1.12。

4.3.2 参数设置

为使情感分类模型能够准确识别情感态度，本文通过阅读文献和实际调参得到最优的参数。为确保模型得到的参数是实验中获得的最优值，唯一变量为预调参数，其他参数严格保持一致。为充分利用计算资源，在进行参数寻优时不使用 BERT 预训练模型，只在 Restaurant 数据集上进行测试，模型迭代次数设置为 20，学习率设置为 0.003。

(1) BERT 预训练参数

在预训练阶段，BERT 预训练模型使用的版本为 BERT-Base-Uncased，模型大小约 110M，部分参数设置值如表 4.3 所示。

表 4.3 BERT 预训练参数表

| 参数名称 | 参数定义 | 参数数值 |
|-------------------------|------------|-------|
| hidden_act | 激活函数 | gelu |
| hidden_size | 隐藏层维度 | 768 |
| max_position_embeddings | Token 最大长度 | 512 |
| num_attention_heads | 注意力头的个数 | 12 |
| num_hidden_layers | 隐藏层层数 | 12 |
| vocab_size | 字典大小 | 30522 |

(2) batchsize 值

batchsize 即一次训练所抓取的数据样本数量，其数值大小影响模型的训练速度和优化能力。通过查阅资料和阅读文献^[67]可知，随着 batchsize 的增大，模型训练时间逐渐缩短，但是当 batchsize 值大到一定程度时，训练时间的变化不再明显。同时在实验中发现，batchsize 值太大时，模型的收敛速度变慢，导致模型需要迭代更多次才能获得最优值。综合实验硬件条件，最终选择的 batchsize 值

为 32，此状态下的模型能够在最短的时间内获得高的准确率。

(3) 卷积核尺寸

不同大小的卷积核在做卷积操作时得到的特征并不相同，因此需要通过实验得到特征提取效果最好的卷积核参数，具体实验结果如表 4.4 所示：

表 4.4 不同卷积核尺寸对应模型性能表

| 卷积核大小 | Acc | F1 |
|----------------|--------------|--------------|
| [2] | 82.20 | 75.18 |
| [2,3] | 82.71 | 75.64 |
| [2,3,4] | 83.62 | 76.07 |
| [2,4,5] | 82.17 | 73.78 |
| [3,4,5] | 82.35 | 73.99 |
| [3,4,6] | 82.90 | 75.37 |
| [4,5,6] | 83.80 | 76.80 |
| [5,6,7] | 83.07 | 75.51 |
| [6,7,8] | 82.69 | 75.27 |
| [3,4,5,6] | 83.26 | 75.54 |

从表中可以看出卷积核个数为 1 个，2 个或者 4 个时的效果要比个数为 3 时的效果差，这是因为卷积核数量过少时不能提取到关键特征，遗漏关键信息，而当卷积核数量过多时会把一些冗余无关的特征也提取到，这也会影响模型的预测能力。在卷积核为三个的组合中卷积核尺寸分别为 4、5、6 时模型表现最优，这是因为当卷积核尺寸过小时，感受野就偏小，进而无法提取完整的词语表征，如果尺寸过大，就会引入噪声，也会影响模型预测能力。基于以上实验结果，最终选择尺寸分别为 4、5、6 的 3 个卷积核作为多粒度卷积层的参数。

(4) dropout 值

针对大型神经网络训练过程中存在的训练时间长、过度适应等问题，研究人员往往使用 dropout 方法对原网络进行“瘦”处理。所谓的“瘦”，是指在一个深

度网络的训练期间，以某种几率临时舍弃临时舍去固定数量神经元节点，从而减少了神经网络的复杂性。本文在特征融合层中采用了 dropout 减少网络复杂度，模型准确率随 dropout 值变化的结果如图 4.2 所示。

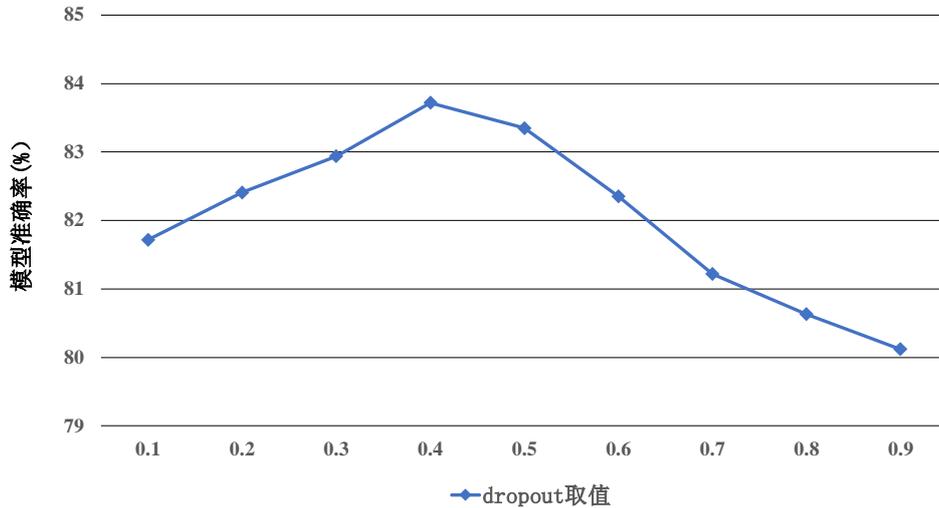


图 4.2 dropout 变化对模型准确率折线图

从折线图可以看出，随着 dropout 值的增大，模型准确率呈现先增后减的趋势，当 dropout 值为 0.4 时，模型准确率达到最高。显然，当 dropout 取值过低时，模型结果过于依赖训练集，过拟合现象无法缓解；当 dropout 取值过高时，神经网络太过简化，容易遗漏关键特征，因此最终确定 0.4 为 dropout 的值。

(3) 其他参数设置

文中其他超参数设置如下：初始化词向量嵌入维度为 300，位置嵌入和词嵌入维度均为 30，GCN 层的层数为 3 层，GCN 层隐藏层维度为 50，学习率设为 0.003，采用 Adam 算法作为优化器，正则化系数为 0.0001，每个数据集均训练 50 个 epoch。

4.4 对比模型设置

为评估 GCNN 模型的性能，将方面级情感分析领域的其他模型与 GCNN 模型在同样的数据集上做对比。

ATAE-LSTM^[55]: 该模型在 LSTM 层对输入信息进行优化, 在词嵌入同时融入了方面词的信息, 接着使用注意力机制给方面词分配情感权重, 实现更精准的情感分析。

IAN^[56]: 该模型分别对方面词和上下文进行建模, 并且在方面词和上下文中都加入了注意力机制, 在对方面词建模过程中, 模型采用上下文作为 Query 向量, 同时采用方面词作为 Query 向量对上下文建模时, 最终得到融合了二者的交互信息的向量表示。

RAM^[57]: 该模型首先进行预训练, 之后利用双向 LSTM 获取了每一个隐藏层的输出, 接着再用多重注意力对每一个隐藏层给予权重, 最后将注意力结果与 GRU 非线性结合, 从而对目标情感进行预测。

TNet^[58]: 该模型首先通过 Bi-LSTM 得到包含上下文信息的向量, 之后构建了多层的 CPT 结构将方面信息融合, 接着用卷积神经网络将特征提取, 最后输出结果。

ASGCN^[59]: 该模型将句法关系和图卷积网络结合, 同时构造方面词和语义信息的表示, 然后利用注意机制进行二者交互, 捕获相关句法约束和远距离单词依存关系。

CDT^[60]: 该模型通过依赖树来建模句子结构, 将 Bi-LSTM 层输出向量作为依赖树的初始点嵌入, 接着通过 GCN 对依赖树学习到的语法信息进行关系建模, 最后采用平均池化聚合方面信息和上下文信息。

BiGCN^[61]: 通过提出一种新的概念层次, 构建层次句法图和层次词汇图, 并利用双层交互图卷积网络来实现这一目标, 有效地利用语料库中的单词共现信息, 并且可以更好地理解不同类型的句法依存关系。

InterGCN^[62]: 该模型首先为依赖项树上的每个句子构造的普通依赖图, 然后通过考虑上下文单词和特定于方面的单词之间的句法依赖性来完善图形, 以得出以方面为中心的图, 接着以方面为中心的图和相应的嵌入矩阵被输入以方面为中心的 GCN, 以捕获关键方面和上下文单词。

DGEDT^[63]: 该模型提出了一种双向 Transformer 网络结构, 首先将 Bi-LSTM 提取到的隐藏信息输入双向 Transformer 模块, 最后通过最大池化操作和注意力

机制来匹配上下文方面词和情感词。

R-GAT^[64]: 该模型首先定义了一种新颖的面向方面的依赖树结构, 然后重构和修剪依赖分析树, 接着利用改进的图注意力网络对新的依赖树进行编码, 实现情感分类。

kumaGCN^[65]: 该模型首次研究面向方面层次情感分类的潜在结构, 构建了统一的自我注意网络用于诱导语义依赖信息, 接着使用 GCN 进行面向方面层次的情感分类, 同时模型引入了门控机制用于更好合并树结构, 显著提升了信息表示能力。

DM-GCN^[66]: 该模型通过依赖树和多头自注意分别生成了语法图和语义图, 并提出了一种动态多通道图卷积网络来有效地从生成的图中学习相关信息, 该网络能有效优化句法信息、语义信息及其组合。

4.5 实验结果及分析

本文在三个公开数据集上分别比较了 GCNN 模型和对比模型的预测准确率, 其中对比模型的实验结果均来自原论文, 最终实验结果如表 4.5 所示。表中 “#” 表示情感分类模型在该数据集上的 SOTA 结果, 下划线表示情感分类模型在表中的次优结果。“Att” 表示基于注意力机制的模型, “Syn+GCN” 表示基于句法依存树的图卷积网络模型, “Syn+Att” 表示句法依存树与注意力相结合的模型, “*” 表示模型引入了 BERT 预训练方法。

表 4.5 模型实验结果

| 模型 | Restaurant | | Laptop | | Twitter | | |
|-----------|------------|-------|--------|-------|---------|-------|-------|
| | Acc | F1 | Acc | F1 | Acc | F1 | |
| ATAE-LSTM | 77.20 | - | 68.70 | - | - | - | |
| IAN | 78.60 | - | 72.10 | - | - | - | |
| Att | RAM | 80.23 | 70.80 | 74.49 | 71.35 | 69.36 | 67.30 |
| | TNet | 80.79 | 70.84 | 76.54 | 71.75 | 74.97 | 73.60 |

续表 4.5

| 模型 | Restaurant | | Laptop | | Twitter | | |
|-------------|-------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| | Acc | F1 | Acc | F1 | Acc | F1 | |
| ASGCN | 80.77 | 72.02 | 75.55 | 71.05 | 72.15 | 70.40 | |
| Syn+GCN | CDT | 82.30 | 74.02 | 77.19 | 72.99 | 74.66 | 73.66 |
| | BiGCN | 81.97 | 73.48 | 74.59 | 71.84 | 74.16 | 73.35 |
| | InterGCN | 82.23 | 74.01 | 77.86 | 74.32 | - | - |
| | DGEDT | 83.90 | 75.10 | 76.80 | 72.30 | 74.80 | 73.40 |
| Syn+Att | R-GAT | 83.30 | 76.08 | 77.42 | 73.46 | 75.57 | 73.82 |
| | kumaGCN* | 86.43 | 80.30 | 81.98[#] | 78.81[#] | <u>77.89</u> | <u>77.03</u> |
| | DM-GCN* | 87.66[#] | 82.79[#] | 80.22 | 77.28 | 78.06[#] | 77.36[#] |
| 本文模型 | GCNN | <u>86.82</u> | <u>80.68</u> | <u>81.47</u> | <u>78.07</u> | 76.80 | 75.76 |

从表中可以看出本文提出的 GCNN 模型在 Restaurant 和 Laptop 数据集上均取得了 Acc 和 F1 值的次优结果。在 Restaurant 数据集中, 与目前表现最好的 SOTA 模型在 Acc 和 F1 值上分别低了 0.84 个百分点和 2.11 个百分点; 在 Laptop 数据集中, 与目前表现最好的 SOTA 模型在 Acc 和 F1 值上分别低了 0.51 个百分点和 0.74 个百分点; 在 Twitter 数据集上的 Acc 和 F1 值比最优模型分别低了 1.26 个百分点和 1.6 个百分点, 但是相较于其他方法的模型仍有较大提升。综合来看, GCNN 模型在各个数据集的表现超过了大多数情感分类模型, 取得了较为先进的情感分类结果, 证明了模型在方面级情感分类任务中的有效性。

ATAE-LSTM、IAN、RAM 和 TNet 作为引入注意力机制的经典模型, 相较于融合了句法信息和将数据在空间上表示的方法仍存在一定的不足, 模型准确率在“Syn+GCN”和“Syn+Att”这两类方法之后。这是因为只依靠注意力机制并不能有效的识别非正式语句中方面词和情感词之间的关联。Twitter 数据集中包含非正式的语言包括符号表情等更为明显, 因此这类模型在 Twitter 数据集上的效

果更差。通过对比 ASGCN、CDT、BiGCN 这三个模型，可以看出“Syn+GCN”方法能够很好的通过句法树提取句子中的依赖关系，准确定位情感词在句中的位置，而图卷积神经网络的树形结构可以很好的表示词向量的共现关系，进而取得更高的预测结果，也证实了 GCNN 模型基于句法关系的图卷积神经网络通道的合理性。

基于“Syn+Att”的方法在模型对比实验中取得了最好的预测结果，其中 R-GAT 和 DGEDT 方法不依赖普通的依赖树关系，而是对依赖树进行改进，降低了依赖关系不准确性带来的误导，接着还进一步构建了增强网络使模型能够更好的学习特征表示，因此模型性能较其他模型有了大的提升。kumaGCN 和 DM-GCN 模型是目前方面级情感分析领域情感分类结果为 SOTA 的模型，两个模型都没有选择图卷积神经网络，因为 GCN 在节点表示过程中存在边关系的权值一直不变且和其他边存在相似的缺点。同时这类模型还有效将注意力与句法关系相结合进行优势互补，降低了单一方法带来的缺点。另一方面在，在模型上加入 BERT 预训练方法后模型性能会得到大幅的提升，可见包含 BERT 模型有着强大的语义信息提取能力。

本文提出的 GCNN 模型一方面构建了学习语法信息的图卷积网络通道，另一方面构建了学习语义信息的多粒度卷积通道，接着将两通道的输出向量在特征融合层进行特征融合，实现特征加强。双通道架构能够同时学习句子的语义和语法特征，当句子句法关系不明显时，可以根据语义关系辅助判断，充分发挥特征融合的优势，增强了模型的鲁棒性，提升了模型预测性能。

4.6 相关实验验证

4.6.1 卷积对比实验

为获得更加丰富的语义信息，本文设置了窗口大小分别为 4、5、6 的卷积核用于提取文本序列的 N-gram 特征。为证明本文提出多粒度卷积操作给模型性能带来的提升要高于单粒度方法，本小节以 Restaurant 数据集为实验数据集，在其他参数不变，只改变卷积核的情况下进行对比实验，实验结果如表 4.6 所示。

表 4.6 卷积对比实验结果

| 卷积核大小及个数 | Acc | F1 |
|----------------|--------------|--------------|
| [4] | 84.03 | 76.80 |
| [5] | 84.82 | 77.35 |
| [6] | 83.62 | 76.54 |
| [4,5] | 85.94 | 79.21 |
| [4,6] | 85.36 | 78.14 |
| [5,6] | 85.58 | 78.86 |
| [4,5,6] | 86.82 | 80.68 |

根据上表，从卷积核个数上看：当卷积核个数为 1 时，卷积核尺寸等于 5 的模型结果表现更优；当卷积核个数为 2 时，卷积核尺寸分别为 4 和 5 的模型结果表现更优；但是准确率都要低于卷积核个数为 3 的模型，说明多粒度卷积效果要比单个和双个卷积核的效果要好。从卷积窗口大小上看，卷积核尺寸为 5 时的模型性能要优于卷积核尺寸为 4 时的模型性能和卷积核尺寸为 6 时的模型性能，即在文本中，对特征提取效果的卷积核窗口大小为 5。综合来看，本文提出的多粒度卷积层能够从不同角度挖掘文本序列在局部的信息特征，进而提升模型预测准确率。

4.6.2 消融实验

为研究 GCNN 模型中各层结构对促进情感分析任务的效果，在只改变研究结构且其他条件不变的情况下进行消融实验。消融实验对象为 BERT 预训练模块、基于句法信息的图卷积网络模块 (DepGCN)、基于语义信息的多粒度卷积模块 (MulCNN)、特征融合模块。通过删减模型的结构评估以上四个模块对模型性能的贡献度，下面对消融实验模型进行说明。

1. “MulCNN”：将句法关系层、图卷积网络层和特征融合层删除，模型在多粒度卷积层学习到特征向量后直接进入全连接层，最终输出结果；

2. “DepGCN”：将多粒度卷积层和特征融合层删除，模型在图卷积网络层得

到特征向量后直接进入全连接层，最终输出结果；

3. “GCNN w/o attention”：仅将模型结构图中的特征融合层删除，模型缺少特征融合层后，将图卷积网络层和多粒度卷积层的特征向量拼接后再进行后续任务；

4. “GCNN w/o BERT”：仅将 GCNN 模型中的 BERT 预训练模型删除。

表 4.7 消融实验结果

| 模型 | Restaurant | | Laptop | | Twitter | |
|--------------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | Acc | F1 | Acc | F1 | Acc | F1 |
| MulCNN | 83.37 | 76.49 | 77.60 | 74.63 | 74.92 | 72.73 |
| DepGCN | 84.71 | 77.96 | 79.31 | 75.67 | 76.03 | 74.44 |
| GCNN w/o attention | 85.04 | 78.33 | 80.32 | 77.25 | 76.19 | 74.71 |
| GCNN w/o BERT | 83.80 | 76.81 | 78.16 | 74.84 | 75.37 | 73.46 |
| GCNN | 86.82 | 80.68 | 81.47 | 78.07 | 76.80 | 75.76 |

从消融实验结果表 4.7 中可以看出，基于多粒度卷积神经网络的 MulCNN 模块在失去句法特征和特征向量融合后，模型在三个数据集上分别下降了 3.45，3.87，1.88 个百分点，模型下降幅度大，可见句法特征信息和特征融合方式对模型性能提升贡献较大，是保证模型预测效果的核心组件。

基于句法关系及图卷积网络的 DepGCN 模块失去语义特征信息和特征融合层后，模型在三个数据集上预测准确率下降幅度较小，可见利用多粒度卷积操作确实对模型性能提升有帮助，但是语义信息的贡献度要比句法关系信息的小。

“GCNN w/o attention”模型没有使用基于注意力机制的特征融合层，在 Restaurant 数据集的准确率下降了 1.68 个百分点，说明特征融合层确实能够有效的代替常用的向量拼接操作，进一步加强特征向量，提升模型性能。

“GCNN w/o BERT”模型去掉了 BERT 预训练模型后较原模型在 Restaurant 数据集的准确率下降了 2.92 个百分点，一方面可以看出即使没有预训练模型也能超过对比模型中的多数基线模型，说明了本文构建的情感分类模型的有效性；

另一方面也说明加入 BERT 预训练语言模型能够大幅的提升模型准确率。

总体上看，所有的消融实验模型的结果对比原模型都出现不同幅度的下降，说明原模型中各个模块都有其存在的合理性。

4.6.3 GCN 层数对模型性能的影响

GCN 层数的多少能给模型预测性能带来很大的变化，是整个模型中非常重要的参数。为研究 GCN 层数对模型性能的影响，在只改变 GCN 层数的情况下，观察模型在三个数据集上准确率和 F1 值的变化情况。实验将层数设置为一至八层，实验结果如图 4.3 和图 4.4 所示。

从下面两幅图可以看出，GCN 层数在 1-3 层间呈现出模型性能随着层数增加而提升的情况，模型的准确率和 F1 值性能均在层数为 3 时达到最高值，可见太浅层的 GCN 网络不能完全获得关键特征信息，从而降低模型预测准确率。随着层数的进一步增加，模型性能开始出现不同程度的下降，在层数为 6 层之前，模型性能下降幅度较小，而随着层数的进一步增加，模型性能开始出现大幅度的下降，可见模型网络层数过多也不能提高模型表现，这是因为模型网络的层级过深导致每个节点能够融合的节点数量更多，包括一些不相关的节点，这些冗余无关的信息最终导致预测效果变差。

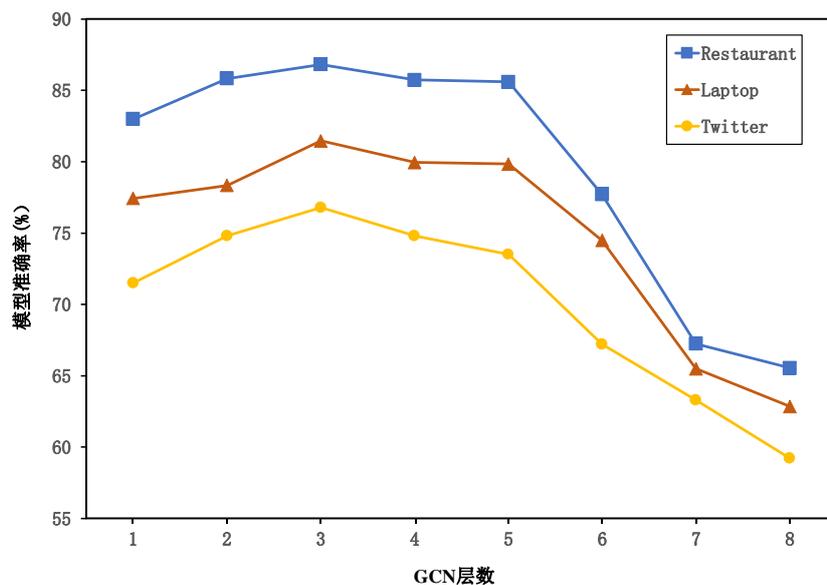


图 4.3 GCN 层数对模型准确率的影响

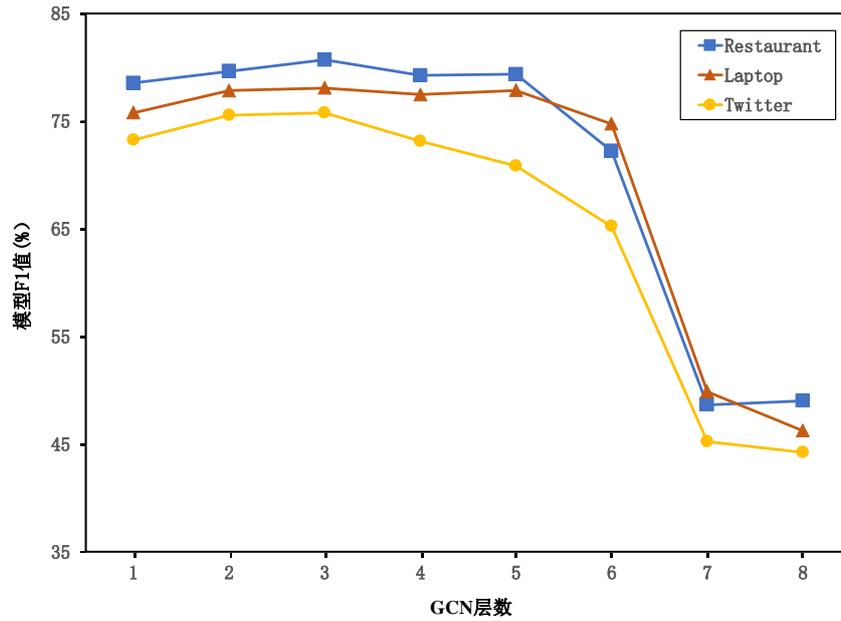


图 4.4 GCN 层数对模型 F1 值的影响

4.6.4 注意力可视化

在消融实验中我们探讨了在去除特征融合模块之后的模型性能变化，从实验中可以看到当模型删除基于注意力机制的特征融合模块后，准确率在分别在 Restaurant 和 Laptop 数据集上下降了 1.78 和 1.15 个百分点。本小节将以数据集中的样本为例，对例句做注意力可视化分析。

在例句“*But the location is very friendly to me.*”中，方面词为“*location*”，方面词对应的情感词为“*friendly*”，显然本句对于“*location*”的情感态度是 *positive*。通过可视化分析获取例句的注意力分布表如表 4.8 所示。

表 4.8 例句注意力权值分布表

| 例句 | But | the | location | is | very | friendly | to | me |
|----|------|------|----------|------|------|----------|------|------|
| 权值 | 0.08 | 0.01 | 0.25 | 0.02 | 0.08 | 0.45 | 0.03 | 0.08 |

在从上表中可以看出注意力得分最高的是情感词“*friendly*”和方面词“*location*”，两者的注意力权重和占句子的 70%，“*the*”和“*to*”等连接词的注

注意力权重则较低,可见注意力可以给句中对情感分析无关和具有干扰性的词语分配更低的权重,而对情感判断有贡献的词赋予更高的权重。

对于句子方面词不止一个的情况,在例句“Although the food was very delicious, the service was poor.”中,方面词为“food”和“service”,方面词对应的情感词为“delicious”和“poor”,对应得情感极性分别为 positive 和 negative。例句中方面词注意力权重分布图如上图 4.5、4.6 所示。

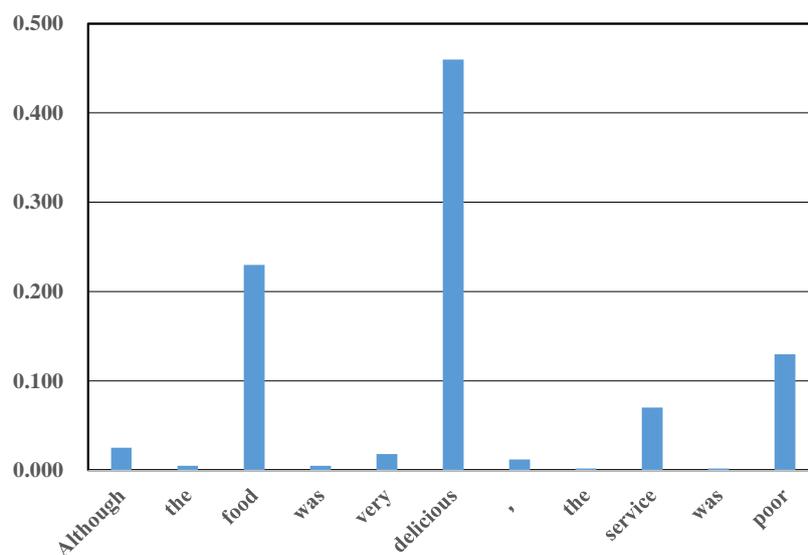


图 4.5 方面词 food 注意力权重分布图

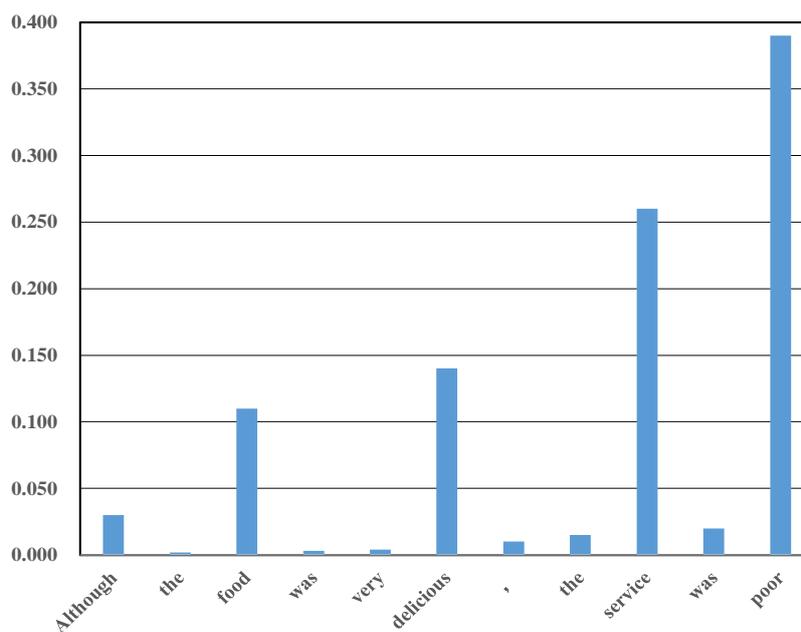


图 4.6 方面词 service 注意力权重分布图

以上两图展示了同一句子面对不同方面词时的各词的权值分布,从图中可以看出在对不同方面词建模时,各成分的权重得分也不一样。当建模方面词为“food”时,情感词“delicious”的权重最高,此时作为干扰词“poor”所占的权重则偏低,模型预测时考虑“delicious”更多;当建模方面词为“service”时,情感词“poor”的权重最高,句子的前半句作为干扰信息,此时赋予了更低的权重。显然,注意力机制能够有效识别对方面词情感判断有作用的词语,进而提供更多的注意力,同时注意力机制能够根据方面词的改变赋予每个词语新的权重。虽然注意力机制获取权重之前要遍历所有的输入信息,这会加大计算机的计算压力,但是由于注意力模块在实验中优秀的表现,已成为自然语言处理领域应用最广泛的“组件”之一。

4.6.5 实例分析

为更直观的体现 GCNN 模型在评论语句中能比其他基线模型有更高的预测准确率,将 ATAE-LSTM、IAN 模型、DepGCN 模块、MulCNN 模块及 GCNN 模型分别在五个样本例句上进行实例验证,用红色和蓝色标记方面词,实验结果如表 4.9 所示。

表 4.9 实例分析

| 序号 | 评论 | ATAE-LSTM | IAN | DepGCN | MulCNN | GCNN |
|----|--|-------------------------|-------------------------|------------------------------------|------------------------------------|------------------------------------|
| 1 | Even though its good seafood , the prices are too high. 【译】: 尽管它的海鲜不错,但价格太高了。 | (N_x, N_{\checkmark}) | (N_x, N_{\checkmark}) | $(P_{\checkmark}, N_{\checkmark})$ | $(P_{\checkmark}, N_{\checkmark})$ | $(P_{\checkmark}, N_{\checkmark})$ |
| 2 | The driver updates don't fix the issue, very frustrating. 【译】: 驱动程序的更新并没有解决这个问题,非常令人沮丧。 | N_{\checkmark} | N_{\checkmark} | O_x | N_{\checkmark} | N_{\checkmark} |
| 3 | The food is usually good but it certainly isn't a relaxing place to go. 【译】: 食物通常很好,但它肯定不是一个放松的地方。 | (O_x, P_x) | (O_x, N_{\checkmark}) | (P_{\checkmark}, O_x) | (N_x, N_{\checkmark}) | $(P_{\checkmark}, N_{\checkmark})$ |

续表 4.9

| 序号 | 评论 | ATAE-LSTM | IAN | DepGCN | MulCNN | GCNN |
|----|--|-------------------------|------------------------------------|-------------------------|-------------------------|------------------------------------|
| 4 | A large is \$20, and toppings are about \$3 each. 【译】: 一个大号的是 20 美元, 配料大约是每个 3 美元。 | O_{\checkmark} | N_x | N_x | N_x | O_{\checkmark} |
| 5 | From the build quality to the performance , everything about it has been sub-par from what I would have expected from Apple. 【译】: 从产品质量到性能, 关于它的一切都不符合我对苹果的期望。 | (N_{\checkmark}, P_x) | $(N_{\checkmark}, N_{\checkmark})$ | (N_{\checkmark}, O_x) | (N_{\checkmark}, O_x) | $(N_{\checkmark}, N_{\checkmark})$ |

对于第一个样本中的方面词“seafood”来说, ATAE-LSTM 和 IAN 模型都是基于注意力的方法, 两个方法都容易注意到“high”这个词, 但是“high”对于方面词“seafood”是干扰信息, 因此模型预测出错。虽然句法依存关系可以在某一方面与某些词之间建立直接的联系, 但在复杂句子中, 方面与意见词之间不存在联系, 在第二个例句中, “driver updates”方面与“frustrating”意见词的句法距离较远, 无法识别两者的关系, 因此造成基于句法关系的 DepGCN 模块对方面词“driver updates”的预测为中性。另外, 在例句 3 中, DepGCN 模块没有捕获关键字“isn't”的特征表示, 造成了该句预测失败, 相比之下, 基于多粒度卷积网络的 MulCNN 模块则能准确把握单词间的语义相关性, 因此能够准确预测。最后两个例句表明, 融合了语义和句法特征的 GCNN 模型能够很好地考虑句法知识和语义信息的互补性, 实现对复杂和非正式语句的高精度预测。

4.7 本章小结

本章为模型实验及测试的具体章节。首先对实验所用到的三个公开数据集进行介绍并对三个数据集进行预处理操作, 接着解释了用于评估和比较模型性能的评价指标, 然后实验的环境和配置作出说明。在确定以上信息后, 对卷积核尺寸及个数、dropout 值进行参数调优实验, 进一步的依据相关文献和资料设定其他

参数值。为验证模型优劣，设置了三大类对比模型进行对比实验，最终实验证明了模型的有效性。后续通过相关实验进一步探究模型结构功能，包括设置消融实验研究模型各层次对模型准确率的贡献情况、GCN 层数对模型性能的影响、注意力机制可视化分析，最后通过实例分析验证了 GCNN 模型在情感分析任务中的有效性。

5. 总结与展望

5.1 工作总结

互联网的快速发展极大的改变人们的生活方式，人们对热点事件的态度、电商平台上的购物、舆论事件的管控等越来越多的人类活动都有互联网的深度参与，如何准确又快速的识别评论语句中蕴含的情感态度已成为自然语言处理领域的热点方向。一方面，情感分析在电商购物中可以给消费者提供购买建议和避雷，也可以给商家提供质量反馈和售卖策略等优点。另一方面，情感分析可以给决策数据提供支撑，保证网络环境和社会生活的良好运作。

针对方面级情感分类任务，本文首先对情感分析理论基础和技术进行说明和介绍，包括常用词向量表示方法的优缺点，基于循环神经网络和卷积神经网络的模型结构及功能，注意力机制和依存句法分析的实现原理及计算方式，向量融合常用的两种方法。

由于目前主流情感分类模型往往提取单一特征作为分类依据，并未尝试过将图卷积网络的特征与多粒度卷积网络的特征融合后进行情感判别。本文提出一种基于双通道信息融合的方面级情感分类模型，模型可分为学习句法信息的图卷积网络模块和学习语义信息的多粒度卷积模块，在特征融合层通过注意力机制进行双通道之间的信息融合，使模型可同时考虑句法结构的互补性和语义相关性，加强了特征表示。通过在三个公开数据集上的实验结果和选取的 12 个具有代表性的对比模型上做比较可以得出，本文提出的模型在准确率和 F1 值上都超过了大多数模型，证明了模型的有效性。

为进一步了解模型中超参数和层次结构对模型性能的影响，在实验前对卷积核个数及尺寸、dropout 值等参数进行寻优实验，之后做了一系列对比分析实验，包括对比多卷积操作下模型性能的改变，调整 GCN 层数探究模型准确率的变化情况，设置消融实验研究各通道和主要层次对模型准确率的贡献情况，对注意力进行可视化研究和设置对比模型对五个案例上的预测结果进行详细分析。

5.2 未来展望

本文提出的 GCNN 模型在 Restaurant、Laptop、Twitter 数据集的预测准确率超过了目前大多数情感分类模型，取得了对比实验中的多个次优结果，但是仍存在进一步完善和改进的空间。

(1) 当代网民在网上表达个人观点时，热衷于在评论语句中加入表情、符号、颜文字等非正式的内容，而这些表情符号往往蕴含着评论者的情感态度。如何用恰当的算法对特征进行表示然后融入到模型中进行端到端学习，是一个对提升模型预测准确率非常有帮助的研究方向。

(2) 多粒度卷积操作能有效的提取本文局部特征，进而丰富特征信息，但也存在天然的局限性，即面对较长的评论文本时，卷积操作只能进行浅层特征提取，忽略长距离间的依赖关系。未来可以考虑采用其他方式补齐这一缺点，提升模型对长评论文本的预测能力。

(3) 本文在提取句子依赖关系时选取了目前效果最好的依赖解析模型，但是模型依然存在一定的依赖解析错误，这些错误信息会影响模型的预测结果。如何降低错误信息的表示和缓解错误信息带来的影响，是一项对提升模型整体效果非常有帮助的研究内容。

参考文献

- [1] 中国互联网络信息中心发布第 50 次《中国互联网络发展状况统计报告》[J]. 国家图书馆学刊,2022,31(5):12.
- [2] Li X L, Zhang L, Liu B, et al. Distributional similarity vs. PU learning for entity set expansion[C]//Proceedings of the ACL 2010 Conference Short Papers. 2010: 359-364.
- [3] Yessenalina A, Yue Y, Cardie C. Multi-level structured models for document-level sentiment classification[C]//Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. 2010: 1046-1056.
- [4] Mower E, Matarić M J, Narayanan S. A framework for automatic human emotion classification using emotion profiles[J]. IEEE Transactions on Audio, Speech, and Language Processing, 2010, 19(5): 1057-1070.
- [5] Engonopoulos N, Lazaridou A, Paliouras G, et al. ELS: a word-level method for entity-level sentiment analysis[C]//Proceedings of the International Conference on Web Intelligence, Mining and Semantics. 2011: 1-9.
- [6] Vinodhini G, Chandrasekaran R M. Sentiment analysis and opinion mining: a survey[J]. International Journal, 2012, 2(6): 282-292.
- [7] Cai G, Lyu G, Lin Y, et al. Multi-level deep correlative networks for multi-modal sentiment analysis[J]. Chinese Journal of Electronics, 2020, 29(6): 1025-1038.
- [8] Chen X, Sun C, Wang J, et al. Aspect sentiment classification with document-level sentiment preference modeling[C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 3667-3677.
- [9] Ma X, Zhu Q, Zhou Y, et al. Improving question generation with sentence-level semantic matching and answer position inferring[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 34(5): 8464-8471.
- [10] Sun K, Zhang R, Mensah S, et al. Aspect-level sentiment analysis via convolution over dependency tree[C]//Proceedings of the 2019 Conference on Empirical

- Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019: 5679-5688.
- [11] Ku L W, Chen H H. Mining opinions from the Web: Beyond relevance retrieval[J]. Journal of the American Society for Information Science and Technology, 2007, 58(12): 1838-1850.
- [12] Chen W T, Lin S C, Huang S L, et al. E-HowNet and automatic construction of a lexical ontology[C]//Coling 2010: Demonstrations. 2010: 45-48.
- [13] Hu M, Liu B. Mining and summarizing customer reviews[C]//Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2004: 168-177.
- [14] Agarwal B, Mittal N, Bansal P, et al. Sentiment analysis using common-sense and context information[J]. Computational Intelligence and Neuroscience, 2015:30-36.
- [15] 罗浩然,杨青.基于情感词典和堆叠残差的双向长短期记忆网络的情感分析[J]. 计算机应用,2022,42(4):1099-1107.
- [16] Pang, B, Lee, L. and Vaithyanathan, S. Thumbs up? Sentiment classification Using Machine Learning Techniques[C]//Proceedings of Annual Conference of the Association for Computational Linguistics. 2002:79-86.
- [17] Sharma A, Dey S. A boosted svm based ensemble classifier for sentiment analysis of online reviews[J]. ACM SIGAPP Applied Computing Review, 2013, 13(4): 43-52.
- [18] 濮泽堃.基于机器学习的电商评论情感分析系统[D].南京邮电大学, 2020.
- [19] Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality[C]//Advances in Neural Information Processing Systems. 2013:3111-3119.
- [20] Tang D, Wei F, Qin B, et al. Sentiment embeddings with applications to sentiment analysis[J]. IEEE Transactions on Knowledge and Data Engineering, 2015, 28(2): 496-509.

- [21] Kim Y. Convolutional neural networks for sentence classification[C] //Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Stroudsburg. 2014: 1746-1751.
- [22] Lai S, Xu L, Liu K, et al. Recurrent convolutional neural networks for text classification[C]//Proceedings of the 29th AAAI Conference on Artificial Intelligence. 2015: 2267–2273.
- [23] 郑诚,魏素华,曹源.结合语法信息的 BG-CNN 用于方面级情感分类[J].计算机工程与应用,2022,58(5):148-155.
- [24] Zhu X, Sobihani P, Guo H. Long short-term memory over recursive structures[C]//International Conference on Machine Learning. PMLR, 2015: 1604-1612.
- [25] Dey R, Salem F M. Gate-variants of gated recurrent unit (GRU) neural networks[C]//2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS). IEEE, 2017: 1597-1600.
- [26] Zhang S, Zheng D, Hu X, et al. Bidirectional long short-term memory networks for relation classification[C]//Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation. 2015: 73-78.
- [27] Wang Y, Huang M, Zhu X, et al. Attention-based LSTM for aspect-level sentiment classification[C]//Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. 2016: 606-615.
- [28] 胡艳丽,童谭骞,张啸宇,彭娟.融入自注意力机制的深度学习情感分析方法[J].计算机科学,2022,49(1):252-258.
- [29] Nguyen H T, Le Nguyen M. Effective attention networks for aspect-level sentiment classification[C]//2018 10th International Conference on Knowledge and Systems Engineering (KSE). IEEE, 2018: 25-30.
- [30] Huang B, Ou Y, Carley K M. Aspect level sentiment classification with attention-over-attention neural networks[C]//International Conference on Social Computing,

- Behavioral-cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation(SBP-BRiMS). 2018: 197-206.
- [31] Scarselli F, Gori M, Tsoi A C, et al. The graph neural network model[J]. IEEE Transactions on Neural Networks, 2008, 20(1): 61-80.
- [32] Chaudhuri U, Banerjee B, Bhattacharya A. Siamese graph convolutional network for content based remote sensing image retrieval[J]. Computer Vision and Image Understanding, 2019, 184: 22-30.
- [33] Schwarzer M, Rogan B, Ruan Y, et al. Learning to fail: Predicting fracture evolution in brittle material models using recurrent graph convolutional neural networks[J]. Computational Materials Science, 2019, 162: 322-332.
- [34] Bastings J, Titov I, Aziz W, et al. Graph Convolutional Encoders for Syntax-aware Neural Machine Translation[C]//Conference on Empirical Methods in Natural Language Processing. 2017:1957-1967.
- [35] 巫浩盛, 繆裕青, 张万桢, 周明, 文益民. 基于距离与图卷积网络的方面级情感分析[J]. 计算机应用研究, 2021, 38(11): 3274-3278
- [36] 谢珺, 王雨竹, 陈波, 张泽华, 刘琴. 基于双指导注意力网络的属性情感分析模型[J]. 计算机研究与发展, 2022, 59(12): 2831-2843.
- [37] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014: 1532-1543.
- [38] Peters M, Neumann M, Iyyer M, et al. Deep Contextualized Word Representations[C]// The Annual Conference of the North American Chapter of the Association for Computational Linguistics. 2018: 2227-2237.
- [39] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]// Annual Conference on Neural Information Processing Systems. 2017: 5998-6008.
- [40] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[EB/OL]. (2018-06-13), [2022-12-08]. <https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf>.

- [41] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[C]// The Annual Conference of the North American Chapter of the Association for Computational Linguistics. 2019: 4171–4186.
- [42] Schouten K, Frasincar F. Survey on aspect-level sentiment analysis[J]. IEEE Transactions on Knowledge and Data Engineering, 2015, 28(3): 813-830.
- [43] Kong J, Wang J, Zhang X. Hierarchical BERT with an adaptive fine-tuning strategy for document classification[J]. Knowledge-Based Systems, 2022, 238.
- [44] Sakata W, Shibata T, Tanaka R, et al. FAQ retrieval using query-question similarity and BERT-based query-answer relevance[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2019: 1113-1116.
- [45] Kaneko M, Komachi M. Multi-head multi-layer attention to deep language representations for grammatical error detection[J]. Computacion Y Sistemas, 2019, 23(3): 883-891.
- [46] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735-1780.
- [47] Chung J, Gulcehre C, Cho K H, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling[J]. CoRR, 2014, abs/1412.3555.
- [48] Shuman D I, Narang S K, Frossard P, et al. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains[J]. IEEE Signal Processing Magazine, 2013, 30(3): 83-98.
- [49] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[C]//Proceedings of the 5th International Conference on Learning Representations. 2017: 24-26.
- [50] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate[J]. Computer Science, 2014, 48(5): 334-340.

- [51]Robinson J J. Dependency structures and transformational rules[J]. *Language*, 1970:259-285.
- [52]Mrini K, Deroncourt F, Bui T, et al. Rethinking self-attention: An interpretable self-attentive encoder-decoder parser[J]. *CoRR*, 2019, abs/1911.03875.
- [53]Pontiki M, Galanis D, Pavlopoulos J, et al. Semeval-2014 task 4:Aspect based sentiment analysis[C]//*Proceedings of the 8th International Workshop on Semantic Evaluation*. 2014:27-35.
- [54]Dong L, Wei F, Tan C, et al. Adaptive recursive neural network for target-dependent twitter sentiment classification[C]//*Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 2014: 49-54.
- [55]Wang Y, Huang M, Zhu X, et al. Attention-based LSTM for aspect-level sentiment classification[C]//*Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2016: 606-615.
- [56]Ma D, Li S, Zhang X, et al. Interactive attention networks for aspect-level sentiment classification[C]//*Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 2017: 4068-4074.
- [57]Peng C,Sun Z,Bing L,et al. Recurrent Attention Network on Memory for Aspect Sentiment Analysis[C]// *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2017: 452-461.
- [58]Li X, Bing L D, Lam W, et al. Transformation networks for target-oriented sentiment classification[C]//*Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. 2018: 946–956.
- [59]Zhang C, Li Q, Song D. Aspect-based sentiment classification with aspect-specific graph convolutional networks[C]//*Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 2019: 4568-4578.
- [60]Sun K, Zhang R, Mensah S, et al. Aspect-level sentiment analysis via convolution over dependency tree[C]//*Proceedings of the 2019 Conference on Empirical*

- Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019: 5679-5688.
- [61]Zhang M, Qian T. Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020: 3540-3549.
- [62]Liang B, Yin R, Gui L, et al. Jointly learning aspect-focused and inter-aspect relations with graph convolutional networks for aspect sentiment analysis[C]//Proceedings of the 28th International Conference on Computational Linguistics. 2020: 150-161.
- [63]Tang H, Ji D, Li C, et al. Dependency Graph Enhanced Dual-transformer Structure for Aspect-based Sentiment Classification[C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 6578-6588.
- [64]Wang K, Shen W, Yang Y, et al. Relational Graph Attention Network for Aspect-based Sentiment Analysis[C]// Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 3229-3238.
- [65]Chen C, Teng Z, Zhang Y. Inducing target-specific latent structures for aspect sentiment classification[C]//Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2020: 5596-5607.
- [66]PANG S, XUE Y, YAN Z, et al. Dynamic and multi-channel graph convolutional networks for aspect- based sentiment analysis[C]//Findings of the Association for Computational Linguistics: ACL-IJCNLP. 2021: 2627-2636.
- [67]Lin T, Kong L, Stich S, et al. Extrapolation for large-batch training in deep learning[C]//International Conference on Machine Learning. PMLR, 2020: 6094-6104.

致 谢

总以为来日方长，却不知时光匆匆，硕士三年求学时光转瞬即逝。始于 2020 年初秋，我离开家乡，第一次来到素有“黄河明珠”之称的兰州市，在这里我感受到西北人民的热情朴素，了解到塞北风光的雄奇壮阔，品尝到西北特有的美食佳肴。兰州财经大学不断见证着我的成长，目光所及，校园里的点点滴滴汇聚心头，失去与收获，遗憾与欣喜，但更多的还是感谢。

盛行千里，不忘师恩。首先我要感谢我的导师李强老师，他在忙碌的教学工作中仍不忘挤出时间关心我的学习和生活状况。他是一位专业知识渊博的学者，给我提供了最好的学习与实验设备，在论文撰写过程中给予了我悉心的指导和宝贵的意见；他也是一位人生路上的良师，在我在生活迷茫时给我指点迷津，阐述道理，让我能够面对一次次的挫折，勇往直前。在此，谨向李强老师致以崇高的敬意和由衷的感谢！

父母之爱，为之深远。其次我要感谢父母的养育之恩，你们一直坚定不移尊重、支持我的选择，并给予我无微不至的照顾。你们日以继夜，不辞辛苦的工作换来了我现在的一切，却也换来了日渐衰老的身体。希望我能尽快成为您依靠的肩膀，为您分担解忧。

山水一程，有幸相遇。我还要感谢 520 实验室的伙伴们，师兄何多魁，师姐王文瑞，学长刘祥强，刘伟，刘学文。当我学习上有不懂地方时你们总会教我正确的思路与方法，是你们让原本枯燥乏味的科研生活变得多姿多彩。感谢同门王雪绒和师弟赵金雨、何博文，师妹李聪聪，我们一起讨论问题，交流经验，一起参加比赛，一起进步。感谢我可爱的室友，从陌生到熟悉，回到寝室后总会有无限的欢乐，从不觉得无趣。

最后，我要感谢百忙之中抽出时间来评审我的论文的各位专家教授以及答辩委员会的老师们，感谢你们对本文的指导与宝贵意见！

攻读硕士学位期间发表的论文及科研情况

已发表论文:

[1]黄建民,李强,王雪绒,李聪聪.基于 BERT 融合多模块的方面级情感分析[J].井冈山大学学报(自然科学版),2021,42(06):64-68.

专利:

[1]李强,黄建民等,一种酱酒糖化堆翻堆处理装置,专利号:ZL2021 2 2906638.X。
[2]李强,黄建民等,一种酱酒糖化堆翻堆入窖管理系统,专利号:ZL 2021 2 2906701.X。

参与课题:

[1]基于大数据的个性化自适应学习模式研究,项目编号:Lzufe2018C-12。