

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：基于双重注意力的方面级情感分析

研究生姓名：李聪聪

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：2024年5月31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： 李陈陈 签字日期： 2024年5月31日

导师签名： 李强 签字日期： 2024年5月31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：

- 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；
- 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内容。

学位论文作者签名： 李陈陈 签字日期： 2024年5月31日

导师签名： 李强 签字日期： 2024年5月31日

# **Aspect level sentiment analysis based on dual attention**

**Candidate: Li CongCong**

**Supervisor: Li Qiang**

## 摘要

自然语言资源是一种以文本形式呈现的资源,其背后往往隐藏着人们对某种事件或商品的看法。如果我们能够对这些资源进行准确分析,找出隐藏在文字背后的情感态度,并对这些数据进行合理高效的利用,将会给企业、消费者以及相关部门带来巨大的利益。文档或句子层面的情感分析任务是对整个文档或句子进行情感分析。然而,由于一篇文章或句子中存在多个具有不同情感极性的主体,这种情感分析方法的准确性就会降低。因此,更细粒度方面的情感分析引起了越来越多的关注。基于此,人们提出了许多方法,例如用于人工设计特征的支持向量机、基于注意力机制的长短期记忆网络、基于依赖解析的图神经网络。虽然这些方法都有很不错的性能,但他们都忽略了一项重要的语法信息:依赖标签。

回顾现有的 ALSA 方法,从最初基于手工标注特征工程的方法,到后来利用深度学习,人们开始使用 LSTM 的各种变体来解决这个问题。随着注意力和最近 GNN 的引入来解析依赖关系的语法特征,方面级情感分析的模型性能正在逐步增强。模型在简单的语义特征提取上已经很难改进了。

但在语法特征的提取上,还有很大的改进空间。通过观察当前在语法特征提取方面性能良好的模型,发现这些模型只是简单地提取依赖弧的连接信息作为语法特征并将其输入到 GCN 或 GAT 当中。但除了依赖弧的连接信息外,“直接宾语”“形容词修饰语”等依赖弧的标签也是提供句法信息非常重要的部分。所以模型可以将这些依赖弧标签转化为嵌入,对句子进行注意力操作,以捕获这句话中的重要部分。此外,考虑到依赖树中单词和依赖弧标签之间的关系,本文使用图神经网络来处理单词嵌入和依赖标签嵌入。使用预训练的 BERT 模型作为编码器,带来了巨大的性能提升。

基于上述想法,本文提出了两个有效的模型来完成方面级别的情感分类任务。一种是基于 Bi-LSTM,一种是基于 BERT。为使模型的性能达到最优,选取了具有代表性的模型在三个公开数据集上进行了比较实验。实验结果显示,本文提出的模型在预测准确率和 F1 值上有不同幅度的提升。为了进一步了解模型中注意力层对模型性能的影响,设置了一个删除了依赖嵌入层、依赖 GCN 和依赖注意力模块的 Non-Dep 模型以及改变了两个注意力模块的运行方式来验证注意力层

对模型性能的提升作用。最后进行实例研究模型的推理过程和两个注意力模块的作用。

**关键词：**情感分析 依赖标签 BERT 注意力机制 句法信息

## Abstract

Natural language resources are a type of resource presented in textual form, often hiding people's views on a certain event or commodity. If we can accurately analyze these resources, identify the emotional attitudes hidden behind the text, and make reasonable and efficient use of this data, it will bring huge benefits to enterprises, consumers, and relevant departments. The task of sentiment analysis at the document or sentence level is to perform sentiment analysis on the entire document or sentence. However, due to the presence of multiple subjects with different emotional polarities in an article or sentence, the accuracy of this sentiment analysis method will be reduced. Therefore, more fine-grained emotional analysis has attracted increasing attention. Based on this, many methods have been proposed, such as support vector machines for manually designing features, long short-term memory networks based on attention mechanisms, and graph neural networks based on dependency analysis. Although these methods all have good performance, they all overlook an important syntactic information: dependency labels.

Looking back at the existing ALSA methods, from the initial approach based on manually annotated feature engineering to the later use of deep learning, people began to use various variants of LSTM to solve this problem. With the introduction of attention and recent GNN to parse the syntactic features of dependency relationships, the performance of

aspect level sentiment analysis models is gradually improving. The model is already difficult to improve on simple semantic feature extraction.

But there is still a lot of room for improvement in the extraction of grammatical features. By observing the current models that perform well in syntax feature extraction, it is found that these models simply extract the connection information of dependency arcs as syntax features and input them into GCN or GAT. But in addition to the connection information of the dependent arc, labels such as "direct object" and "adjective modifier" are also very important parts in providing syntactic information. So the model can convert these dependency arc labels into embeddings, perform attention operations on the sentence, and capture the important parts of the sentence. In addition, considering the relationship between words and dependency arc labels in the dependency tree, I use graph neural networks to handle word embedding and dependency label embedding. The use of pre trained BERT models as encoders has brought significant performance improvements.

Based on the above ideas, this article proposes two effective models to complete aspect level sentiment classification tasks. One is based on Bi LSTM, and the other is based on BERT. To achieve optimal performance of the model, representative models were selected for comparative experiments on three public datasets. The experimental results show that the proposed model has different degrees of improvement in prediction

accuracy and F1 value. In order to further understand the performance of the attention layer in the model, a Non Dep model was set up, which removed the dependency embedding layer, dependency GCN, and dependency attention module, and changed the operation mode of the two attention modules to verify the improvement effect of the attention layer on model performance. Conduct case study on the reasoning process of the model and the role of two attention modules.

**Keywords:** Sentiment Classification; Dependency Label; BERT; Attention Mechanism; Syntactic Information

# 目 录

<b>1 绪论</b>	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	4
1.2.3 基于深度学习的方法	5
1.3 论文研究内容	8
1.4 论文组织结构	9
<b>2 相关理论知识与技术</b>	11
2.1 情感分析概述	11
2.2 数据预处理	11
2.2.1 分词	12
2.2.2 句法分析	14
2.3 词向量表示	14
2.3.1 one-hot 编码	15
2.3.2 Word2Vec	16
2.3.3 GloVe	18
2.3.4 BERT	19
2.4 循环神经网络	21
2.5 卷积神经网络	24
2.5.1 普通卷积神经网络	24
2.5.2 图卷积神经网络	25
2.6 注意力机制	28
2.7 依赖句法分析	30
2.8 本章小结	32

<b>3 方面级情感分类的双重注意力模型</b> .....	33
3.1 任务定义 .....	34
3.2 依赖解析层 .....	34
3.3 嵌入层 .....	35
3.4 编码器层 .....	35
3.5 GCN 层 .....	36
3.6 注意力机制层 .....	37
3.7 Bi-LSTM 层 .....	38
3.8 最终 MLP 层 .....	38
3.9 本章小结 .....	39
<b>4 实验与分析</b> .....	40
4.1 数据集介绍 .....	40
4.2 评价指标 .....	40
4.3 实验环境与超参数配置 .....	41
4.4 对比模型设置 .....	42
4.5 实验结果及分析 .....	44
4.5.1 消融实验.....	44
4.5.2 GCN 层数的影响.....	45
4.5.3 案例研究.....	46
4.6 本章小结 .....	47
<b>5 总结与展望</b> .....	48
5.1 工作总结 .....	48
5.2 未来展望 .....	48
<b>参考文献</b> .....	50

# 1 绪 论

## 1.1 研究背景及意义

### 1.1.1 研究背景

近几年来,随着互联网的飞速发展与网络基础设施的逐步完善使得线上购物、线上社交等新兴生活方式被人所追捧。人们采用文本评论的形式对某些社会热点以及各种消费体验来表达自己的感受和观点。如果我们可以整理收集这些文本评论并了解到用户对他们评论主体所持的情感极性,则会很大程度上帮助那些需要了解大众对于某一事件的态度行业。例如:在政府部门做决策时,可对大众的意见进行参考,大大提高了政府部门做决策的效率。商家也可以将大众对于商品的态度作参考,以此来进行更好的改进,提高商品的质量。如今互联网越来越普及,人们发表的评论的数目也越来越多,则靠人工去对这些评论进行整理归纳费时费力,因此学者们开始研究使用机器学习的方法去进行情感分析。

情感分析(Sentiment Analysis, SA),又称意见挖掘(Opinion Mining, OM)或主观分析<sup>[1]</sup>,是自然语言处理领域中的一项基础任务。情感分析最终是要使用计算机技术挖掘人们在文本评论里的情感倾向<sup>[3]</sup>。情感分析根据其所处理文本粒度的不同可划分为三个级别:篇章级(Document-level)、句子级(Sentence-level)与方面级(Asspect-level)<sup>[7]</sup>。其中,篇章级别的情感分析主要是对文本所传达的情感倾向进行评估。但在实际生活中,除了对事件整体进行评论,人们通常对评价主体的某一方面表达自己的观点。例如:在评论“这家蛋糕店的蛋糕味道很好,但是样式很差”中,用户对蛋糕的“味道”和“样式”两个方面做了评价,我们无法从整体上判断用户对蛋糕的情感极性,因为用户对蛋糕的味道是赞赏的态度但是对样式却是批评的态度。如果在对该句子评论进行情感分析时采用篇章级或句子级的粗粒度情感分析技术,则对不同方面的情感极性难以捕获。而当判断具体某一方面的情感倾向时,更细粒度级的方面级情感分析(Asspect-level Sentiment Analysis, ALSA)可以根据提供的方面词进行更有针对性和灵活性的判断。

### 1.1.2 研究意义

由于互联网使用具有极低的成本,越来越多用户开始使用互联网,这些用户通过网络平台去对某件事情的不同方面表达自己的看法。与关注句子整体情感不同,方面级情感分析技术主要对某一特定方面的情感极性进行获取<sup>[3]</sup>。通常,在 ALSA 任务执行过程中,句子的不同情感可以划分为三类:积极(Positive)、中性(Neutral)与消极(Negative)。

传统的 ALSA 方法主要可以划分为两大类:基于情感词典的方法和基于机器学习的方法。基于情感词典的方法是指构建一个积极、消极和中性情感词汇的词典。然后对评论中包含情感方向的词与情感词典里的词按照一定的规则去进行比对,然后再去计算情感极性。这种方法的适应性较差,一旦出现复杂的评论文本,这种方法就会难以进行处理。相比于前者,基于机器学习的方法就展现出了更好的拓展性。它采用统计学上的相关算法将情感分析任务看成分类任务。但是这种方法需要大量的人力去对特征进行筛选。最近几年,学者们将基于神经网络的深度学习模型应用于自然处理领域的研究中,其中卷积神经网络和循环神经网络是最具代表性的。这些模型都能有效提高分类性能。深度学习模型与传统的机器学习方法在特征提取上有所不同,它能够自动地进行特征提取,有效地补充了传统机器学习技术的缺陷。

虽然学者们认为基于深度学习算法的 ALSA 方法有很大的潜力,在近几年中展示出了很大的优势,但是它仍存在着很多的不足。首先,因为文本特征大都比较复杂,且在不同的语境中各词语所表达的意思也会产生不同,所以在提取文本特征时比较困难。其次,当句子中存在着多个情感词和评价主体时,如何准确地将其进行对应也是目前基于深度学习的 ALSA 模型需要精进的部分。

综上所述,如何将深度学习算法更好地在 ALSA 领域中进行运用具有重大意义,值得我们认真地去进行思考和探索。

## 1.2 国内外研究现状

情感分析是一种旨在挖掘出文本中用户所表达的情感极性,并对一些有情感倾向的评论性文本进行分析、处理、归纳以及推理。基于情感字典、基于机器学

习和基于深度学习是目前三种主要的实现情感分析任务的方法。基于情感词典的方法是将文本内带有情感色彩的词与情感词典进行匹配从而得到情感态度,这种方法的对预先构建好的情感词典具有较大的依赖性;在对特征进行提取时,基于机器学习的方法对手工标注的情感特征使用算法进行学习,该方法的缺点是比较依赖相关模型;在对特征进行提取时,基于深度学习方法允许模型学习特定训练集的特征,然后对测试集进行预测和分析,基于深度学习的方法是目前最受欢迎的方法。

### 1.2.1 基于情感词典的方法

基于情感词典的方法是指根据不同情感词典中的情感词的情感极性,来实现情感极性的划分。不可否认,情感词典的质量对预测结果十分重要,目前研究者们经常使用的情感词典有 HowNet<sup>[11]</sup>错误!未找到引用源。和 NTUSD<sup>[12]</sup>错误!未找到引用源。。

基于情感词典的情感分析是一种简单却很很有用的情感分析方法,具体做法是:第一步,通过人工构建或者自主学习的方法构建一个包含了丰富情感词的词典,情感词的词性应包含积极、中性、消极三个方面。第二步,进行文本分析,将要分析的词和构建的情感词典进行对比匹配,然后计算其情感得分,当该词与情感词典匹配成功时,就可以将该词的情感得分合计到所有词的情感得分里面。第三步:计算情感得分:将每一个词的情感得分汇总起来计算完整句子的情感得分。分值为正的为积极情感,分值为负的为消极情感。最后,通过得到的情感得分来进行情感极性的输出。基于情感词典的情感分析通过构建海量包含情感极性的词来构建情感词典,然后使用情感模型通过计算情感得分的方式来实现情感分析。该方法的优点是很容易理解和操作。但是它的缺点也非常明显,因为出现在社交媒体的情感词是不断增加的,需要不断地去更新扩充情感词典,对人工耗费较大,且无法充分考虑上下文的语境从而实现更准确的情感分析。

基于情感词典的方法可以准确反映文本的非结构化特征,易于分析和理解。但这种方法仍然存在一定的缺陷:基于情感词典的情感分类方法过于依赖手工构建的词典,而构建词典需要花费大量的时间和精力,并且词典的质量好坏与构建者的经验有直接关系;情感词典中的情感词随着其语言环境的改变表达出来的意

思也会发生改变,所以这种方法遇到多领域和多语言环境时不能实现很好地效果;此外在使用这种方法时需要上下文之间的语义关系进行充分考虑,这点也值得我们去深入研究。

## 1.2.2 基于机器学习的方法

机器学习的方法主要分为两大类:监督学习和无监督学习。其中,监督学习方法是一种从数据中提取大量参数的技术。在文献<sup>错误!未找到引用源。</sup>中, Nakagawa 及其团队提出,当作为 Max-Ent 识别器的特征时,可以使用基于情感词典的原始分数和某些参数的度量方式。当这种方法变得可行时,我们可以通过分析每个方面的情感倾向来获取评论的整体信息,这将作为一个额外的信号。在本文中我们利用了一个新的技术——深度神经网络,它可以自动地提取出情感相关的词语并计算它们之间的相似度。然而,在文献<sup>[5]</sup>里,当需要挖掘特定上下文中的“pros”和“cons”情绪时,通常会使用特定的情感字典,然后利用这些挖掘出的信息来训练支持向量机(SVM)。支持向量机能够将情感术语分为正面和负面两个方面。对于这两个方面,在解析树中寻找包含其情感的表达式时,应该将距离控制在五个步骤内。最后,使用 SVM 确定情感方向是积极的还是消极的。本文主要讨论了中文文本的表示问题。Choi<sup>[6]</sup>和 Cardie 提出,在使用基础的 SVM 方法对文本进行建模时,应采用简化的词袋模型,因为这样的模型在代表某一方面的情感表达上可能不够充分有效,且模型过于简单。所以他们使用了组合语义学的原则来使得情感表达更加充分,具体做法是将各个部分自身的情感极性按照一定规则去进行组装。和机器学习方法相比,使用组合推理更加有利一些,但是获得更高准确率的方法还是将两者结合起来。文献<sup>[8]</sup>中提出了一种旨在对标签分配的后验分布进行更好地控制的基于后验正则化的监督情感分析的变分方法,通过实验可以得出结论这种方法提高了原始变分方法的效率,预测的准确率得到了提升。文献<sup>错误!未找到引用源。</sup>提出了将隐式情感表示与该情感表示情感标签相同的表示进行对齐的有监督的对比预训练方法,经过测试后发现该方法具有非常好的预测性能。基于上述分析,我们认为有监督的预训练方式在预测准确性和可扩展性上都表现得更为出色。

还有一种方法被称为无监督学习。是指学习时没有任何数据标签,在学习过

程中,将数据输入到给定模型中,然后模型通过使用这些数据去学习,从而达到对任务的执行。无监督学习的优点是不需要人工去对每一个数据点进行标注,可以在通过大量的数据来得到模式,但是大多数时候它的效果不如有监督学习。Popescu 和 Etzion<sup>错误!未找到引用源。</sup>在方面级情感分析任务上设计了一个无监督学习模型,该模型首先建议使用每一个明确的方面在其周围寻找潜在的情感短语,以更全面地识别潜在的情感短语。在测量其邻近程度时,模型采用了解析句子依赖关系的方法。然后,对每一个找到的潜在情感短语进行检查,只保留那些显示情感极性的短语。这个模型的创新之处在于,在确定短语的情感极性时,采用了计算机视觉领域的一种被称为松弛标记的无监督技术,通过在上下文中查找单词语义来确定短语的情感极性。另外,提出一种结合方向图和语义距离的角度标签抽取技术,通过这种方式可以很好地减少由于句子间相似度计算带来的错误标注。Bagheri<sup>[10]</sup>及其团队设计了一个全新的、独立领域的、无监督的、高效的情感分析模型。在这个模型中,首先需要学习多个方面词,在进行方面词学习时采用一个通用的方法,考虑观点词对检测方面的影响程度时采用一套启发式规则,接下来,他们提出了一个新的指标,该指标基于方面频率和交互信息,并使用一个与无监督的随机种子集合共同训练的引导迭代算法来对这些方面进行评分。然后在不对正确的方面用剪枝的方法来去除。最后,在对带有极性的词进行提取之后,对于词库中的带有观点的词,将其映射到提取的显性方面的词的集合上,以此来对隐性的方面进行识别。最后的实验结果表明,在对情感极性进行识别时,该模型比无监督方法展示出了更好的性能。文献<sup>错误!未找到引用源。</sup>提出,在对带有情感倾向的文本进行信息提取时,使用了无监督学习,例如:利用感性进行评分以及模糊熵的方法。Dai<sup>错误!未找到引用源。</sup>等人在 2021 年提出了一个适用于情感分析的多源无监督领域的两阶段多任务学习框架。这个模型通过明确的建模共享,可以在多个数据集上保持出色的建模能力,从而在一定程度上减少了私有知识的损失。

### 1.2.3 基于深度学习的方法

基于深度学习的方法是在句法上对文本中某个方面所表达的带有极性的情感进行识别的方法。这种方法主要使用了机器学习和自然语言处理技术以此来实现情感分析。例如:当我们想要对某家餐厅的“服务”在线评论进行情感分析时,

可以训练一个能够利用句法关系准确识别出对服务好坏做出评价的词,最后对这些带有情感极性的词做分类。此类方法的优点是通过对方面词和其对应的情感词之间句法关系分析可以发现那些出现频率很少的方面词。虽然它的优势明显但是同时它也存在一些不足。例如:当方面词和多个情感词都存在特定的语法练习,则会导致该方面词真正对应的情感词会被混淆,从而出现情感极性识别错误的情况。又或是当方面词与它对应的情感词之间的语法关系并不明显,导致系统识别不出来,也会导致出现情感极性分析错误的情况。除此之外,在对语法关系进行提取时,需要对所分析的文本进行语法解析,处理期间通常伴随着不可预知的错误且处理过程非常缓慢。

2003年 Nasukawa<sup>错误!未找到引用源。</sup>等人最早开始对使用基于句法的情感分析方法进行研究,相比于之前将整个文本的情感分类为正面或负面,提出了一种新的情感分析方法,该方法从文本中提取与特定正负情感极性相关的情感信息。在对网页和新闻文章的数据进行测试时通过语义分析与句法分析器结合来进行测试,表明这种方法是可行的,并且通过在大量数据集上进行测试也能取得当时最好的效果。不同于以往数据大多来源于对于商品的评论,文献<sup>错误!未找到引用源。</sup>的数据采用了电影评论,且进行情感极性判断时,该方法整合了统计分析、WordNet 以及电影知识,通过实验得知,在进行方面级情感分析时该方法确实展现了其优越性。但是这个方法也有其不足之处,对于联系紧密的方面词和情感词它可以很好地去进行语法关系的描述,但是却没有描述那些联系不紧密的方面和情感词。例如对句子“fruit other than orange is also nice”进行句法分析时会出现如下结果:“orange”的情感倾向会被判断为积极地,因为“orange”和“nice”的距离小于“fruit”和“nice”的距离,而实际上“orange”的情感倾向为消极,这就导致了预测发生了错误,导致准确率降低。那么,如何改善这种情况呢,2008年,Filippova 和 Strube<sup>错误!未找到引用源。</sup>发现对句法进行压缩可以解决此类问题,并提出使用依赖树。紧接着,Xu、Hu 和 Shen<sup>错误!未找到引用源。</sup>发现在提取方面和情感次的关系时也可以使用依赖树,由此可得,依赖树可用于方面级情感分析的句法结构分析。我们使用句法依赖树对上面的那句英文句子进行解析,通过依赖关系树情感词“nice”和方面词“fruit”之间直接进行了连接,而“orange”要连接到“nice”要通过好几个单词。另外我们还可以利用依赖树技术来精确地确定方向词与其相应情感词之

间的最短距离，从而有效地避免上述问题可能导致的判断失误。

依赖树揭示句子中单词间的内在联系，每个节点代表一个单词，边揭示单词间的语法关系，有助于理解句子构造和含义。过去研究中，为捕捉单词间复杂关系，研究者倾向于使用循环神经网络（RNN）<sup>[17]</sup>而非卷积神经网络（CNN）<sup>错误!未找到引用源。</sup>。例如，Tang <sup>错误!未找到引用源。</sup>等学者开发了两个长短期记忆网络（LSTM）模型，其独特的记忆机制有效传递目标信息，在分类任务中表现出色，且不需要句法分析器或外部情感词典，灵活高效。工业界和学术界已经认识到句法关系在方面级情感分析中的重要性，并设计了多种模型来捕捉这种关系。然而，这些模型往往忽视了位置信息在识别方面术语情感极性时的重要作用，可能导致处理复杂句子时表现不佳。为了弥补这一缺陷，文献<sup>错误!未找到引用源。</sup>提出了基于双向 GRU 的位置感知双向注意力网络（PBAN），结合双向 GRU 和注意力机制，既捕捉方面词的位置信息，又建模方面词与句子之间的关系，更全面地理解句子情感极性，并在实际应用中表现出色。在 SemEval2014 数据集上的实验证实，PBAN 性能优于其他相关模型。这一成果不仅证实了 PBAN 的有效性，也为我们进一步探索句法关系在情感分析中的应用提供了新的思路。依赖树为理解句子构造和含义提供新视角，而 PBAN 的提出为捕捉句子情感极性提供新解决方案。随着研究深入，相信情感分析领域的理解和应用将越来越深入和广泛。文献<sup>[21]</sup>介绍了一种基于图结构的可扩展半监督方法，名为图卷积网络（Graph Convolutional Networks, GCN）。GCN 结合卷积神经网络（CNN）的特征提取能力和图理论的结构信息，能深入学习节点隐藏性特征并理解节点间边的依赖关系，处理图结构数据效果显著。该研究同时探索句子句法结构，并创新性融入注意力机制，提出目标相关图注意力网络（TD-GAT）<sup>错误!未找到引用源。</sup>模型。TD-GAT 利用单词间依赖关系，结合依赖图和注意力机制，聚焦于目标相关信息。实验显示，TD-GAT 在传播句子情感特征方面优于传统 Glove 嵌入模型，证明其捕捉复杂情感特征的能力。文献<sup>错误!未找到引用源。</sup>还提出融合句法依赖信息和标签信息的新型关系图注意力网络，提升情感分类准确率。这一研究为图卷积神经网络在自然语言处理领域的应用提供新思路，具有广泛应用前景。

随着科技的进步，计算机性能提升为自然语言处理（NLP）提供了强大的技术支持。预训练模型成为 NLP 领域的重要发展方向。早期，Word2vec 模型虽取

得显著成果，但只能表示文本的局部信息。为克服此局限，Pennington 等人<sup>[24]</sup>提出了 GloVe 算法，结合全局统计特征与滑动窗口特征，丰富词嵌入内容并提高模型性能。然而，GloVe 仍是静态词嵌入，难以识别语境变化。为解决多义词问题，Peters 等人<sup>[25]</sup>提出 ELMo 方法，利用大量语料学习并微调词向量，提高模型准确性。2017 年，谷歌团队提出 Transformer 模型<sup>[26]</sup>，基于注意力机制，精度和性能优于 RNN 模型，为 NLP 带来新突破。Radford 等人<sup>错误!未找到引用源。</sup>在此基础上提出生成式预训练模型 GPT，采用二段式训练方法，先无监督预训练后微调，证明预训练模型的应用潜力。同年，谷歌提出 BERT 模型<sup>错误!未找到引用源。</sup>，基于 Transformer 编码器，实现上下文双向抽取，取得显著成绩。许多优秀模型在 BERT 基础上改进，推动 NLP 持续发展。

随着计算机性能提升，预训练模型在 NLP 领域应用广泛。从 GloVe 到 ELMo，再到 GPT 和 BERT，模型的不断涌现推动 NLP 技术发展，为人工智能进步注入新活力。未来，预训练模型在 NLP 领域的应用将更加广泛和深入。

综上所述，方面级情感分析研究领域正吸引越来越多的研究者投身其中，尤其是基于图神经网络和句法结构的研究方法已经逐渐成为研究的主流方向。然而，这种方法依然有一些明显的缺点。首先，在依赖树进行转换的过程中，有可能会导导致某些句子中原有的语法信息出现流失。因此，如何提高依赖树生成效率仍然是一个亟待解决的难题。其次，现有的技术在利用依赖树展示的词性和节点之间的路径信息上还存在明显的不足。此外，基于依赖树来分析网络特征也需要大量的计算时间和内存消耗。第三点，对于样本量相对较少的数据集来说，该模型通常很难进行高效的训练或实现最优性能。

### 1.3 论文研究内容

方面级情感分析，作为情感分析领域中的一项重要任务，其关键在于如何精准而高效地识别文本中各个具体方面所表达的情感倾向。本文对当前的情感分析模型进行深入研究和优化，在继承前人研究成果的基础上，进一步提升模型的性能表现，为情感分析领域的发展贡献更为优秀的模型。具体表现为：

尽管传统的词嵌入技术可以满足文本向向量的转换需求，但在语义信息的完整呈现上仍有所不足，尤其是在处理常见的多义词问题时，其表现尤为困难。针

对上述问题,本文引入了 BERT 预训练语言模型以丰富特征信息,并在词嵌入层中加入了词性和位置信息,从而进一步提升了词嵌入的语义表示能力。在进行特征抽取的过程中,传统依赖于 CNN 和 RNN 的模型在捕获词语之间的长距离语义信息方面表现得相对不足。鉴于句法关系在增强模型文本解读能力上的关键作用,本文构建了一种基于 BERT 的双注意力模型。使用预训练的 BERT 模型作为编码器对句子的语义信息进行充分学习,带来了巨大的性能提升。且在注意力模块层设置了两个注意力模块,方面注意力模块和依赖注意力模块。不仅能注意到句子中的方面信息,也能对句子中提供句法信息非常重要的标签部分进行注意。最后通过 Bi-LSTM 对句子进行进一步的加工和处理,得到包含语义和语法依赖信息更丰富的句子词向量。

为了验证本文所提模型的有效性,我们在 SemEval 2014 年的 Restaurant 和 Laptop 数据集,以及 Twitter 数据集上进行了全面的实验验证。通过与基准模型进行对比分析,我们评估了本文模型的性能表现。此外,我们还设计了消融实验,以验证模型各模块和网络的能力。

## 1.4 论文组织结构

本文分为五章,各章内容概述如下:

第一章:绪论。在这一章节中,详细阐述了本文的研究背景及其重要性,同时探讨方面级情感分析领域的当前研究状况。重点关注情感分析的三大方法,并对基于深度学习的情感分析技术进行深入探讨。最后,将简要概述本文的主要研究内容。

第二章:相关理论和技术。此章节将介绍情感分析所需的理论知识和技术支撑。我们将对方面级情感分析的核心理念进行简要描述,并探讨数据的预处理方法、词向量的表示技巧,以及在深度学习领域中如卷积神经网络、循环神经网络、图卷积网络和注意力机制等关键技术。

第三章:基于 BERT 的双重注意力模型。在本章中,详细介绍了基于 BERT 的双重注意力模型的各个组成部分。这包括用于深度挖掘语义信息的预训练 BERT 层,旨在全面学习句法信息的 GCN 层,以及模型的核心部分注意力机制层。此外,以及后续的 Bi-LSTM 层以及用于最终分类的 MLP 层。

第四章：实验与分析。此章节详细描述了模型性能的验证步骤。详细描述了评估标准、实验数据集以及实验环境，并对超参数的选取和对比模型的建立进行了深入探讨。除此之外，还展示了消融实验与实例验证的成果，并对其进行了深入的探讨。

第五章：总结与展望。最后一章旨在对全文的研究内容进行全面总结，并展望未来研究方向。

## 2 相关理论知识与技术

### 2.1 情感分析概述

情感分析按照细粒度可以划分为篇章级<sup>[29]</sup>、句子级<sup>[30]</sup>和方面级<sup>[31]</sup>三个粒度。篇章级和句子级情感分析属于粗粒度任务，属于二分类和三分类任务，方面级情感分析属于细粒度任务，具有多分类特点。方面级情感分析是一种针对文本中特定方面进行情感倾向判断的技术。以句子“The quality of the notebook is good, but the price is too expensive”为例，这句话包含了两个方面：“quality”和“price”。通过方面级情感分析，可以发现这两个方面所表达的情感分别是“good”和“expensive”。这意味着用户对于笔记本的质量给予了积极的评价，而对于价格则持消极态度。这样的分析结果有助于我们更准确地把握用户的真实需求，为改进产品或服务提供有针对性的建议。

利用深度学习模型进行方面级情感分析主要步骤包括：数据预处理、特征提取、向量表示和情感输出。具体步骤主要内容如图 2.1 所示：

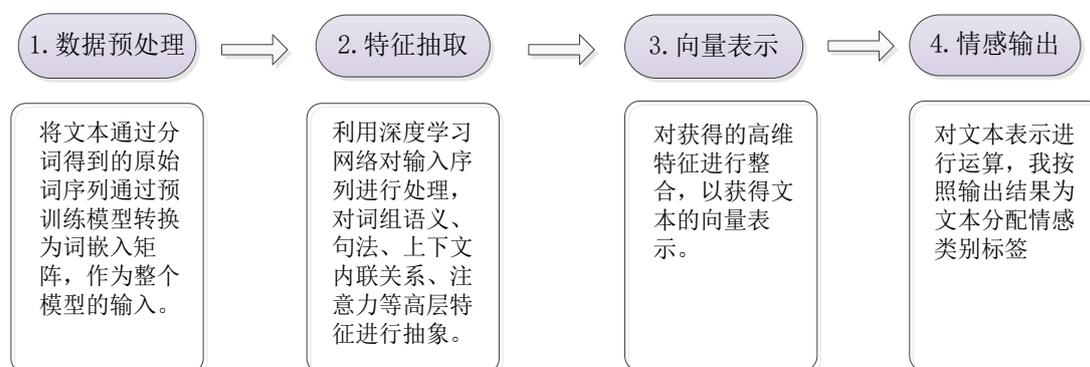


图 2.1 情感分析各步骤任务图

### 2.2 数据预处理

在机器学习中数据质量的好坏对模型的预测和泛化能力有着至关重要的作用，所以在机器学习中进行数据预处理是至关重要的。在本文中，要想得到想要的的数据，就需要对数据进行预处理。

## 2.2.1 分词

在自然语言处理中，为了方便模型进行处理分析，我们需要将句子、段落分解为一个个的词单位，这个过程就叫分词。非结构化的数据可以通过分词的方式转换为结构化的数据，这使得将问题转换为数学问题变得非常简单。在选取语句或文本的度量单位时通常选取单词而不是句子，句子因其过多的信息表达以及相对比较的粒度一般不被作为度量单位。英文句子的构成本来就存在空格，所以在进行分词的时候一般在空格处进行划分，但是在英文中时态不同单词的形态也不同，例如“walk”“walks”“walked”“walking”，当出现不同的形式的时候我们可以用词形还原技术将其还原成单词原形 walk，例如“games”、“boys”这类名词的复数形式，需要使用词干提取技术将其转换为单词原形“game”、“boy”。同时，由于分词是一个很复杂的过程，分词技术的不同也会导致不同的分词效果，分词效果的好坏直接影响模型对句子的理解程度。因此，提升英文数据集分词的效率对于后续实验是否能够高效进行具有至关重要的意义。

至今，存在大约七种主流的英文词汇分类方式，一是基于空格的分词方法，二是基于词典的划分，三是基于规则的划分，四是基于 Penn TreeBank 错误!未找到引用源。 的划分，五是基于 Spacy 的划分，六是基于 Moses 的划分，七是基于子词的划分。

(1) 基于空格的分词方法是几类分词方法中最快速简单的一种方法，它的分割标志是文本中的空格字符，同时它也有显而易见的缺点，例如在进行分词时它一般忽略了单词所在的上下文以及句子的语法，在碰到缩略词以及其他语言时它不能很好地进行处理。

(2) 基于词典的划分主要是依据预先编制的单词词典，对那些需要进行分词处理的英文文本进行科学地分割。

(3) 基于规则的划分主要是用来解决特定问题，例如为特定的语言制定特定的语法规则，当语言不同时，需要创建的规则可能不同，这也使得操作变得相对复杂。

(4) 基于 Penn TreeBank 的划分，该方法是由宾夕法尼亚大学开发，可以用来实现对文本的标记，该方法维护了一个可以反映文本的语法和句法关系的英文的语料库，以此来实现对文本的分割。在对这张表进行设计时规则相对比较严

格，它提出的文本标记方法在语法上是准确的，充分地考虑到了单词的上下文，对解析语法树起到了很大的帮助。然而，该方法也存在一些明显的不足。首先，当文本为英语时，其效果表现得最为出色；文本是其他语言的时候达不到那个效果，其次它需要对文本进行人工注释，这是一项耗时耗资源的过程。

(5) Spacy 的划分能够对句子、单词、空格、标点符号、特殊字符、数字以及整数都是能够进行标记的，缺点就是并不支持所有语言。

(6) Moses Tokenizer 错误!未找到引用源。 是一个开源英语的文本标记器，它将基于规则和使用正则表达式的简单方法结合起来进行文本标记。其主要优势在于能够处理各种文本和语言，无论是正式的还是非正式的或是带有噪音的文本。在对文本进行处理时，我们首先使用空格将文本分隔开来，接着按照既定规则和正则公式进行更深入的分割和合并标识。但是它的缺点也是很显而易见的，首先它在英语语言上效果最好，在其他语言上效果不如英语好，其次它对正则表达式的依赖性较强，在面对比较复杂的自然语言时无法进行准确灵活的处理。

(7) 基于子词的分词方法 (Subword tokenization) 主要是识别文本中出现频率最高的词汇，并为其分配一个特定的 id，而将出现频率低的词拆分成子词。以单词 “long” 为例，若其在文本中出现频率最高，则为其分配一个独特的标识符。相较之下，“longer” 与 “longest” 的出现频率较低，因此会将它们拆分成更小的子词单元，例如 “long”、“er” 和 “est”。子词分词相较于传统的分词方法，其单元更为精细，旨在应对测试集中未出现于训练集中的单词，并有效处理单词的形态变化。当前，用于标记子词的方法主要可以划分为三个不同的类别。该方法通常需要在字母键上增加或删除多个字，以实现对于子词的自动标注。首先，我们采用字节对编码 (BPE) 方法，该方法从选定的一组字符开始，然后持续地合并下一组出现频次较高的字符或子词，直至词汇表规模满足预设要求，从而构建一个能够表达训练集外单词的词汇表。此外，通过引入上下文信息以提高分类准确率和召回率，同时也可以有效地减少冗余标注。在 BERT 模型中，WordPiece<sup>[34]</sup> 方法得到了广泛的应用，这种方法使用 unigram 语言模型来标记文本。除了前述方法外，我们还采用了诸如 SentencePiece、Unigram 语言模型以及子词正则化等子词标记防范策略。这些策略在处理测试集中未出现在训练集中的词汇以及单词形态变化等问题时，展现出了显著的优势。同时，它们的运用还有助于减少训练

时间，并增强对稀有词的稳健性，从而提升了整体模型的性能与稳定性。

### 2.2.2 句法分析

句法分析 (syntactic analysis),又被叫做句法解析 (syntactic parsing),在自然语言处理过程中对句子的语法结构进行分析或者单词之间的句法关系进行识别的步骤。句法分析涵盖两个方面,即句法依赖解析与句法成分解析。前者深入剖析句子内部的语法构造,并精确识别各词汇间的关联性;后者对句中词汇成分进行主次分明的划分。接下来对这两个方面展开详尽的探讨。

句法依赖解析采用了基于树结构的方法,展示了如何将句子中的每一个词连接起来,树上的节点用来表示句子中的词,词与词之间的依赖关系用节点之间的边表示,这种关系解析对理解句子含义有很大帮助。下表是常见的依赖句法关系标签:

句子成分分析也叫做成分解析 (Constituency parsing),是自然语言处理中对句子的语法结构和成分进行分析和识别的过程。该方法依据特定规则构建了一个基于树结构的解析树,并依据句子的递归规则形成了树的基础结构。句子的结构通过树的层次显示,可以清楚地表明句子中的成分和各个成分之间的关系。它的好处是能够更加明确地进行句子的层次化表示,与依赖分析方法相比,它展现出更高的稳定性和更广泛的适用性,在进行句法分析的时候可以和依赖解析搭配着使用。如下表所示是部分成分分析标签:

在对句子进行依赖解析和成分解析过程中,其中还需要对单词进行词性标注,词性标注的过程也被叫做语法标记的过程,做法是将语法标签分配给句子中的各个单词,例如名词、动词、形容词、副词等,在进行文本分析任务和情感分析任务时会有很大的帮助。

## 2.3 词向量表示

词向量表示,也称为词嵌入,是一种将语言文本通过特定技术转化为向量矩阵的方法,这样可以方便地将其输入到电脑端进行后续处理。词向量表示是一种有效的自然语言理解工具,它能够对文本进行语义分析和情感分类等功能,因而得到广泛关注与研究。词向量的表示方法已经经历了显著的进步,其中最具代表

性的技术包括：One-Hot 编码<sup>[35]</sup>、Word2vec<sup>错误!未找到引用源。</sup>、GloVe<sup>错误!未找到引用源。</sup>以及 BERT。它们都是基于词频统计和词典学习等传统方式构建出不同类型的特征空间来进行特征提取。下面将对这四个词嵌入模型进行深入的解析。

### 2.3.1 One-hot 编码

由于人类的语言文字无法被计算机识别,为了使机器可以识别人类的语言文字,在对文字特征进行表示时研究人员选取了汇编语言的“0”和“1”,用“0”表示特征不存在的情况,用“1”表示特征存在的情况。每个特征都有属于自己专属的状态寄存器。例句:他每天都很开心。第一步需要将句子分词成{“他”“每天”“都”“很”“开心”},其中包含了5个词汇,则5就被设置成状态寄存器的数目。如果把这些变量分别对应到不同的词中,那么这个词就是该名词的特征向量。特征向量的表示方式如下:特征向量的表示方式如下:{他:[1,0,0,0,0]}、{每天:[0,1,0,0,0]}、{都:[0,0,1,0,0]}、{很:[0,0,0,1,0]}、{开心:[0,0,0,1,0]}。接下来,我们将通过具体的实例来阐述 One-hot 编码在实际场景中的应用。在语料库里,如果有两句话:“小明喜欢篮球”和“小云喜欢跳舞”。在这两种情况下如何提取句子的特征?首先,将句子的分词设置为{“小明”“小云”“喜欢”“篮球”“跳舞”},这样“小明喜欢篮球”的特征向量就可以表示为:[1,0,1,1,0];如果不包括“小明”和“篮球”,那么描述“小云喜欢跳舞”的特征向量是[0,1,1,0,1]。特征向量提取的具体步骤如图 2.2 示。

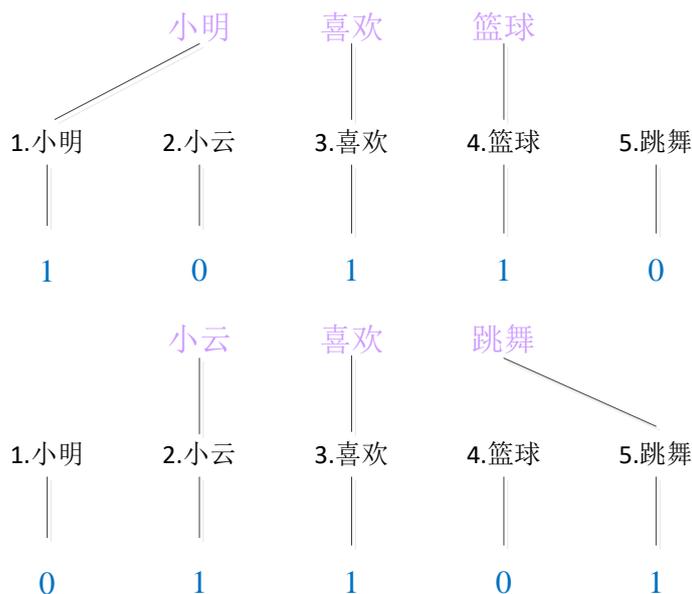


图 2.2 One-hot 提取特征向量

通过以上实例可以发现 One-hot 编码在对语言特征的表达时简单且比较容易理解。但同时它也有着明显的缺点：第一，One-hot 编码只适合文本信息量不太大的情况当文本信息量比较大的时候，该编码方法会出现向量表示时维度大且稀疏的情况，有时候会出现维度灾难；第二，对于词与词之间的顺序问题 One-hot 编码没有考虑，但其实词与词之间的顺序是非常重要的。

### 2.3.2 Word2Vec

2013 年，谷歌地提出了一种名为 Word2Vec 的词向量生成方法。该方法的核心思路在于，通过广泛搜集大量文本语料，运用滑动窗口技术以单词为单位逐步遍历文本内容。在此过程中，采用中心词预测其上下文环境的方式，反之，也可利用上下文词来预测中心词。为实现这一目标，Word2Vec 衍生出两种变体，即 Skip-gram 和 CBOW (Continuous Bag of Words)。在 Skip-gram 模式中，已知中心词，我们通过该词预测上下文词。目前，Skip-gram 方法结合负采样的实践最为普遍。而 CBOW 则是将上下文词向量整合为“词袋”形式进行处理，然后基于这个“词袋”来预测中心词。下列分别对两种模型进行介绍。

#### (1) CBOW

CBOW 模型，也被称为连续词袋模型，拥有使用已知的上下文信息来预测中心词的功能，其模型的具体结构如图 2.3 示。

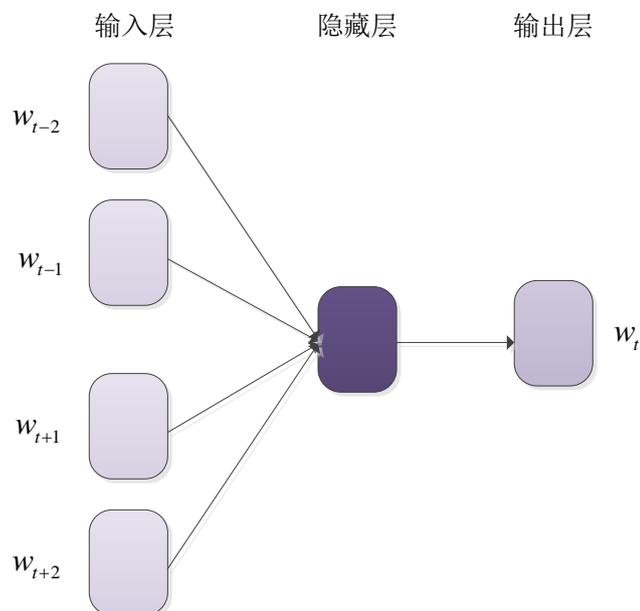


图 2.3 CBOW 模型结构

对于包含  $T$  个词的文本  $\{w_1, w_2, \dots, w_t\}$  来说，CBOW 模型选择了基于文本的 One-hot 编码作为其输入层，其中输入内容是与中心词有关的上下文词汇。模型的隐藏层由一个维度为  $T * M$  的矩阵构成，其中  $M$  代表的是词向量的维度。在隐藏层的学习过程中，输出层运用 softmax 函数来预测中心词。在模型的训练阶段，CBOW 模型的主要目标是提高平均对数概率，从而增强模型的预测准确性。具体公式如式 (2-1) 所示，目标函数如式 (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， $v_t$  为  $w_t$  所对应的 One-hot 向量， $\langle v_t, v_{t+j} \rangle$  为  $v_t$  和  $v_{t+j}$  的点积运算。

## (2) Skip-gram

Skip-gram 模型的主要目标与 CBOW 模型的目标截然不同,它的功能是通过指定的中心词来预测其周边的上下文信息,具体模型结构如图 2.4 所示。

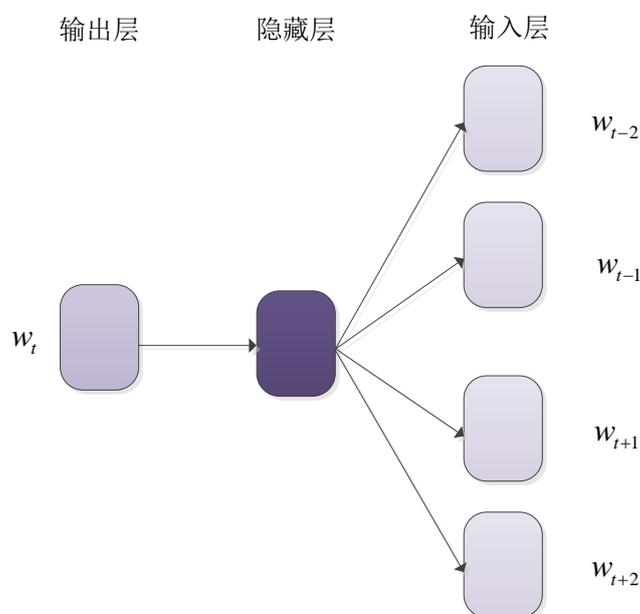


图 2.4 Skip-gram 模型结构

模型的输入层是中心词,而输出层则是上下文词对指定中心词的概率。Skip-gram 模型也以最大化平均对数概率作为其训练目标,但其目标函数与 CBOW 有所不同,具体的目标函数公式如下 2-3 所示。

$$L_{CBOW} = \frac{1}{T} \sum_{t=1}^T \sum_{-c \leq j \leq c, j \neq 0} \log P(w_{t+j} | w_t) \quad (2-3)$$

### 2.3.3 GloVe

在 2014 年的 EMNLP 自然语言处理高级会议上,斯坦福大学 NLP 组发表了一组名为《GloVe: Global Vectors for Word Representation》的研究论文,并进一步提出了 GloVe 模型。GloVe 是一种基于全局词频统计并可以将单词转换成包含一些语义特征的词向量的表示方法,将单词转换为词向量后使用了欧几里得距离或者余弦相似度对语义相似性进行计算,具体的操作步骤如下描述:根据现有的

语料库我们首先需要构建一个共现矩阵  $X$ ，在特定大小的窗口中中心词  $i$  和上下文词  $j$  有一个共同出现的次数  $X$ ，我们用矩阵中的元素  $X_{ij}$  表示，计算给定的词  $k$  对  $i$  和  $j$  词之间相关性公式如下所示。

$$F(w_i, w_j, w_k) = \frac{P_{ik}}{P_{jk}} \quad (2-4)$$

通常情况下，次数的最小值是 1。作者引入了一个衰减函数，用于计算两个单词在窗口间的距离。这个函数的值是用两个单词在窗口间距离的倒数来表示的。两个词在总词中的权重的高低由它们之间距离大小决定，距离越近权重越高，反之则权重越低。上面的步骤计算完之后则构建损失函数如下所示。

$$Loss = \sum_{i,j=1}^V f(X_{ij})(w_i^T w_j + b_i + b_j - \log X_{ij})^2 \quad (2-5)$$

为了计算预测值和真实值之间的欧氏距离，作者选用的损失函数采用了均方误差， $V$  代表了语料库的大小，常出现单词的权重要比那些不常一起出现的单词的权重大，且权重数值达到某一定程度后不会再往上升， $f(x)$  采取了分段函数的形式，公式如下所示。

$$f(x) = \begin{cases} (x/x_{\max})^\alpha & \text{if } x < x_{\max} \\ 1 & \text{otherwise} \end{cases} \quad (2-6)$$

上式中的  $\alpha$  以及  $x_{\max}$  都是超参数， $\alpha$  赋值为 0.75， $x_{\max}$  赋值为 100。

### 2.3.4 BERT

在 2018 年的国际顶级学术会议 NAACL<sup>[38]</sup>上，谷歌的研究团队提出了 BERT 预训练语言模型。该模型通过对输入语料进行预先训练得到一个可用于自然语言处理的分类器来识别和理解用户意图。由于 BERT 模型的引入，NLP 的多个子领域经历了显著的变革，如文本分类<sup>错误!未找到引用源。</sup>、问答系统<sup>错误!未找到引用源。</sup>、序列标注<sup>错误!未找到引用源。</sup>，特别是在层面情感分析领域，由于 BERT 模型的出现，相关记录不断得到更新。

BERT 模型的第一步是要获得主模型的参数，所以我们需要对大量的文本库进行预学习。接下来在主模型之后接一个可以与其匹配的模块来进行下一步的任务，接着在主模型上运用第一步预学习后得到的参数，然后将其进行一个随机初

始化，最后一步则是根据这些收集到的数据，对主模型进行更为细致的调整和优化。如图 2.5 所示是 BERT 的模型结构图。

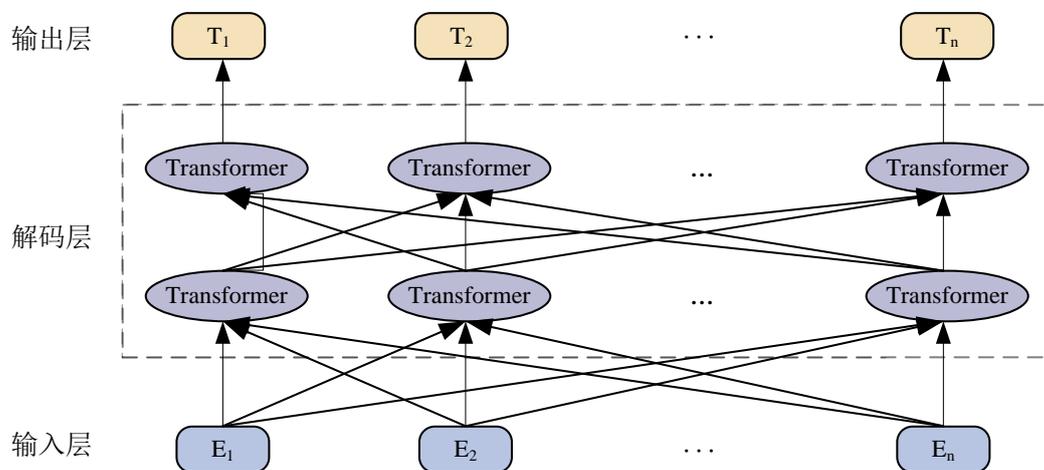


图 2.5 BERT 模型结构

### 1. 输入层融合向量表示

位置嵌入 (Position Embeddings)、词嵌入 (Token Embeddings) 和分段嵌入 (Segment Embeddings) 相加组成了输入层的融合向量。当需要记录文章中单词及其之间的相对位置时，可以采用位置嵌入技术；当需要描述一个特定类别特征时，则通过词嵌入来完成。在把词汇转化为具有固定维度的向量的过程中，需要采用词嵌入的方法；在需要区别两句话的向量表示时，可以使用分段嵌入方法，这样可以在分类任务中清晰地确定两句话的顺序。这种方法为接下来的分类工作和句子的分类提供了关键的参考。如图 2.6 所示为输入层的结构。

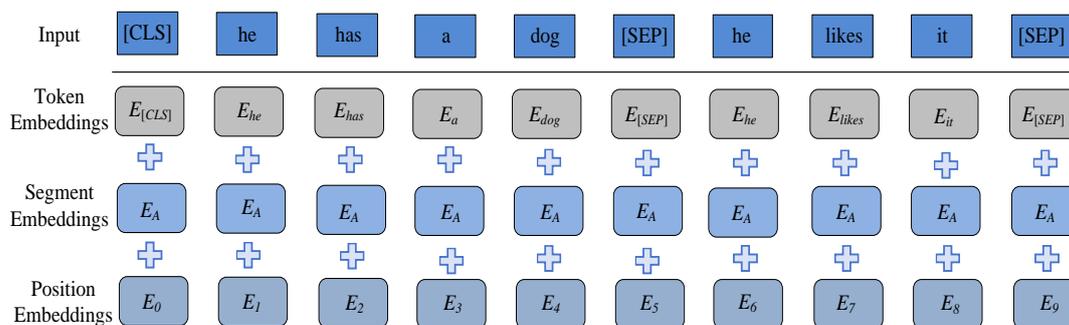


图 2.6 BERT 模型输入表示

## 2. 编码层 Transformer

BERT 模型的核心结构采纳了 Transformer 的编码器部分 (Encoder)，该编码器通过在多头自注意力机制之上融入残差连接 (Residual Connection)、层标准化 (Layer Normalization) 以及线性转换等关键操作，显著提升了模型的性能。具体而言，残差连接的主要职责是将模块的输入和输出信息整合在一起，作为下一个模块的输入数据。这种设计方式简化了输出信息的生成流程，进而也降低了网络训练的复杂性。层标准化则确保神经网络中每个节点的输出均值接近 0，方差接近 1，有效提升了模型的稳定性和收敛速度。此外，本文的特征向量在经过线性转换后，其维度与原始向量保持一致，确保了信息的完整性和一致性。这些改进共同提升了 BERT 模型在自然语言处理任务中的表现。

## 3. BERT 预训练过程

BERT 预训练采用了掩码语言模型 (Mask Language Model, MLM) 和下一句预测 (Next Sentence Prediction, NSP) 这两种并行处理策略。前者是一种基于句法知识的方法，后者是一种基于句法分析的方法，它们都属于文本分类技术。本文采用 BERT 模型的核心目标是提取句子中具有丰富语义含义的词向量，所以这里只介绍 MLM。

在 BERT 模型的处理流程中，当单词被输入后，模型会随机选择其中的 15% 进行信息遮蔽。这一步骤是通过将选定的单词替换为特定符号【MASK】来实现的。为了确保模型能够学习到多样化的知识，研究人员设计了三种不同的处理策略。第一种策略是随机替换选中的单词，第二种是保持原单词不变，第三种则是使用【MASK】符号进行遮蔽。在实际处理过程中，这三种策略按照 1: 1: 8 的比例进行应用。

鉴于上述策略，对 BERT 预训练语言模型的训练过程往往需要花费大量时间，但是，当模型的训练结束后，它可以被直接用于特定的下游任务。截至目前，谷歌的研究团队已在 Github 代码仓库发布了 24 个精简的 BERT 模型，研究者们只需挑选适当的参数，并依据具体的需求进行相应的训练。

## 2.4 循环神经网络

循环神经网络是一个能够处理带有序列特性数据的结构，它的特点是随着时间的推移循环神经网络能够不断地重复。根据循环神经网络的这一特点，它可以对带有时间序列特点的语言文本进行序列特性的抽取。如图 2.7 所示为具体的 RNN 结构。

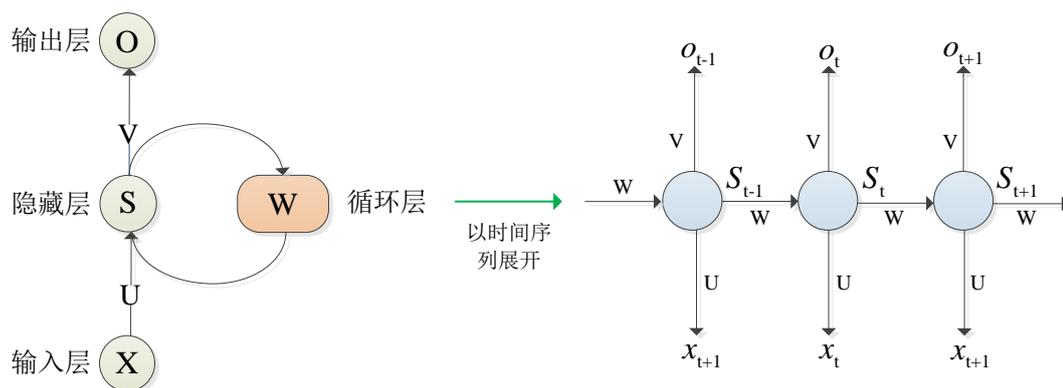


图 2.7 RNN 结构图

Hochreiter 等<sup>[42]</sup>在 RNN 的创新性地提出了长短时记忆网络(LSTM)的概念。这种方法利用了神经元之间的相互连接和相互作用来完成对输入数据的处理。传统的 RNN 能够存储所有时刻的信息，但在加入了门控机制后，LSTM 能够更智能地筛选掉不必要的信息，保留有价值的信息。如图 2.8 所示为 LSTM 结构图。

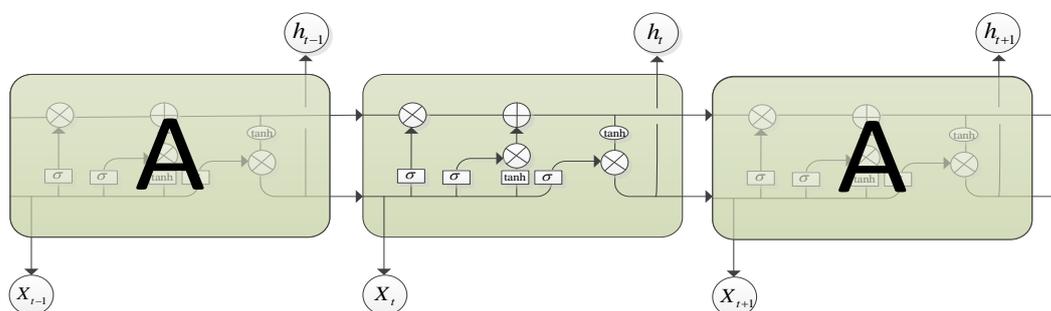


图 2.8 LSTM 结构图

LSTM 在结构上和 RNN 一样，由相同地模块组成。图中的绿色矩阵表示 LSTM 中 4 个相互作用的神经网络层，蓝色圆圈代表逐点进行操作，线条表示节

点间向量之间的传递。如下是各个模块之间的计算公式：

$$i_t = \sigma(W_i[x_t \oplus h_{t-1}] + b_i) \quad (2-7)$$

$$f_t = \sigma(W_f[x_t \oplus h_{t-1}] + b_f) \quad (2-8)$$

$$o_t = \sigma(W_o[x_t \oplus h_{t-1}] + b_o) \quad (2-9)$$

$$\tilde{c}_t = \tanh(W_c[x_t \oplus h_{t-1}] + b_c) \quad (2-10)$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tilde{c}_t \quad (2-11)$$

$$h_t = o_t \otimes \tanh(c_t) \quad (2-12)$$

$f$ 、 $i$ 、 $c$  和  $o$  分别代表了门控机制的遗忘门、输入门、记忆单元和输出门，其中，遗忘门用来确定哪些数据需要被删除，输入门用来确定哪些数据需要被新加进来，输出门用来确定最终的输出数据。在这些过程中，每个变量都有一个值，即为该变量在这一时间间隔内的变化量。矩阵的拼接和点乘运算用  $\oplus$  和  $\otimes$  来表示，特定时刻所对应的状态由下标的  $t$  来表示。 $\sigma$  代表了 *sigmoid* 函数， $\tanh$  代表了双曲正切函数。

与传统的 RNN 方法相比，LSTM 显著降低了梯度消失和梯度爆炸的风险，同时其对长距离目标的依赖性也得到了显著增强。但是在对 LSTM 进行训练时需要耗费大量的时间且要求非常高的硬件能力，基于此 Cho 等人<sup>错误!未找到引用源。</sup>进一步提出了门控循环单元（Gate Recurrent Unit, GRU），经过大量实验验证，GRU 和 LSTM 在参数优化方面能够达到相似的效果，然而，GRU 在模型结构上仅包括更新门和重置门，相比于 LSTM 而言可以大量节约时间成本。如图 2.9 所示为 GRU 的结构图。

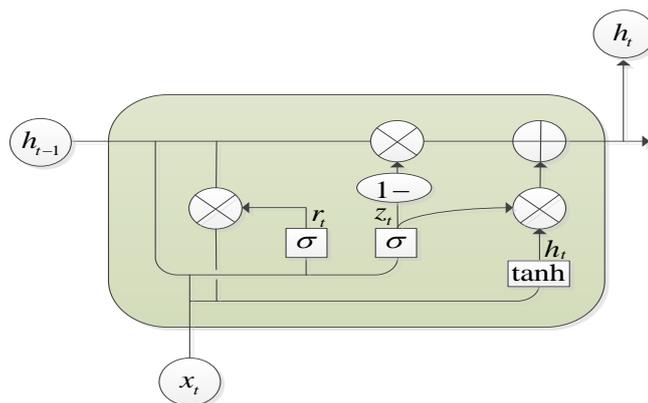


图 2.9 GRU 结构图

GRU 和 LSTM 在输入输出结构方面具有相似性。当接收到当前输入  $x_t$  时，它们都会将这一输入与包含先前节点全部信息的隐藏状态  $h_{t-1}$  进行结合。随后，当前节点会生成一个综合输出  $h_t$  和一个隐藏状态  $h_t$ 。GRU 具体的实现公式如下所示。

$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r) \quad (2-13)$$

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z) \quad (2-14)$$

$$h_t = \tanh(W_h x_t + U_h (r_t \odot h_{t-1}) + b_h) \quad (2-15)$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot h_t \quad (2-16)$$

在上述的公式里，计算重置门和更新门的公式为  $r_t$  和  $z_t$ ， $\sigma$  代表了 *sigmoid* 激活函数，在进行门控机制时，相应的数据输出值经过 *sigmoid* 函数范围控制在  $[0,1]$ ， $(1 - z_t) \odot h_t$  代表了忽略候选状态  $h_t$  中的无效信息， $z_t \odot h_{t-1}$  代表了对上一时刻信息中的有效信息进行记忆。

## 2.5 卷积神经网络

### 2.5.1 普通卷积神经网络

针对全连接网络存在的局限性，研究者们创新性地提出了一种名为卷积神经网络的前向神经网络架构。近年来，卷积神经网络在语音识别、目标识别以及动

作分析等多个领域得到了广泛的推广，并且在这些领域中取得了显著的进步和突破。其基本结构如图 2.10 所示。

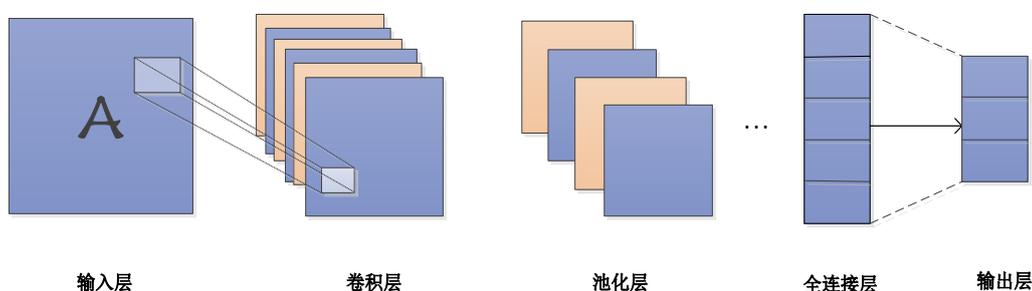


图 2.10 CNN 结构图

在 NLP 领域中，卷积神经网络的输入层输入的是词向量矩阵。从输入层出来后经过卷积层的卷积操作可以获取到输入层的局部区域信息，过滤器和激活函数构成了卷积层，在进行卷积操作时首先要对滤波器的数量、大小、步长等超参数进行设置。样本从卷积层出来后经过池化层进行压缩，这个池化层的主要功能是对关键样本进行高效的收集。通过计算得到每个训练数据集内的权重值，然后根据这个权值来确定是否需要使用全池化机制或者只采用部分池化机制。在这其中，最大池化技术能够扩展样本的各项特性，而平均池化技术则负责提取这些特性的平均数值。全连接层在功能上相当于一个分类单元，它可以将学习到的特征量表达为类别标记空间。在对情感进行分类时，情感类别包括三种：积极、中性和消极。

在对句子进行情感分析时，如果我们准确掌握了句子中某个重要单词的含义，则在识别句子情感时就可以更加准确快捷，卷积神经网络因其可以有效地对局部特征进行提取，所以受到很多情感分析模型的青睐。

## 2.5.2 图卷积神经网络

在处理图片或文字信息时，我们通常会使用普通的卷积神经网络，这些数据类型均属于欧式空间，其特点是数据的规则性。然而，在实际问题中，许多数据并不具备这种规则性，特别是图结构或拓扑结构，例如社交网络、化学分子结构以及知识图谱等。此外，语言本身也是一种图结构，其内部呈现出复杂的树形结

构。因此，在处理这些数据时，我们需要采用相应的方法和技术，以适应其不规则性。

卷积神经网络用来处理规则性数据，形式类似于图片和文字，但是实际生活中都是类似于知识图谱、分子结构等不规则数据。研究人员提出了能够处理图结构等非规则数据的图卷积神经网络（graph convolutional networks, GCN）。

在实现图卷积神经网络的时候，卷积算子和池化算子的构建非常重要。在执行节点级别的任务时，卷积算子被用于学习和提取图数据节点间的局部结构特征。而在执行图级别的任务时，池化算子则负责从图信息中学习和捕捉层级结构表示。现阶段，用于构建图卷积神经网络的技术主要可以划分为两个主要类别：一是基于谱域的方法，二是基于空间的方法。

(1) 基于谱域的图卷积神经网络

在基于谱域的图卷积神经网络中，为了实现图卷积，我们采用了卷积定理，而卷积定理<sup>[49]</sup>则提出了通过函数傅里叶变换的乘积来形成函数卷积的傅里叶变化的方法。具体的公式如式所示。

$$F(f \times g) = F(f) \cdot F(g) \tag{2-17}$$

$$f * g = F^{-1}(F(f) \cdot F(g)) \tag{2-18}$$

其中， $F$  函数表示傅里叶变换函数，而  $f$  和  $g$  则分别代表了原始信号，同时， $*$  和  $\cdot$  分别代表了图卷积算子和乘积算子。图中的拉普拉斯矩阵如图 2.11 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><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr> <tr><td>A</td><td>2</td><td>-1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>B</td><td>0</td><td>3</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>C</td><td>0</td><td>0</td><td>2</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>D</td><td>0</td><td>0</td><td>0</td><td>3</td><td>0</td><td>0</td></tr> <tr><td>E</td><td>0</td><td>0</td><td>0</td><td>0</td><td>3</td><td>0</td></tr> <tr><td>F</td><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	A	2	-1	0	0	0	0	B	0	3	0	0	0	0	C	0	0	2	0	0	0	D	0	0	0	3	0	0	E	0	0	0	0	3	0	F	0	0	0	0	0	1	<table border="1"> <tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr> <tr><td>A</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>B</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr> <tr><td>C</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>D</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr> <tr><td>E</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>F</td><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	A	0	1	0	0	1	0	B	1	0	1	0	1	0	C	0	1	0	1	0	0	D	0	0	1	0	1	1	E	1	1	0	1	1	0	F	0	0	0	1	0	0	<table border="1"> <tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr> <tr><td>A</td><td>2</td><td>-1</td><td>0</td><td>0</td><td>-1</td><td>0</td></tr> <tr><td>B</td><td>-1</td><td>3</td><td>-1</td><td>0</td><td>-1</td><td>0</td></tr> <tr><td>C</td><td>0</td><td>-1</td><td>2</td><td>-1</td><td>0</td><td>0</td></tr> <tr><td>D</td><td>0</td><td>0</td><td>-1</td><td>3</td><td>-1</td><td>-1</td></tr> <tr><td>E</td><td>-1</td><td>-1</td><td>0</td><td>-1</td><td>3</td><td>0</td></tr> <tr><td>F</td><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	A	2	-1	0	0	-1	0	B	-1	3	-1	0	-1	0	C	0	-1	2	-1	0	0	D	0	0	-1	3	-1	-1	E	-1	-1	0	-1	3	0	F	0	0	0	-1	0	1
	A	B	C	D	E	F																																																																																																																																																
A	2	-1	0	0	0	0																																																																																																																																																
B	0	3	0	0	0	0																																																																																																																																																
C	0	0	2	0	0	0																																																																																																																																																
D	0	0	0	3	0	0																																																																																																																																																
E	0	0	0	0	3	0																																																																																																																																																
F	0	0	0	0	0	1																																																																																																																																																
	A	B	C	D	E	F																																																																																																																																																
A	0	1	0	0	1	0																																																																																																																																																
B	1	0	1	0	1	0																																																																																																																																																
C	0	1	0	1	0	0																																																																																																																																																
D	0	0	1	0	1	1																																																																																																																																																
E	1	1	0	1	1	0																																																																																																																																																
F	0	0	0	1	0	0																																																																																																																																																
	A	B	C	D	E	F																																																																																																																																																
A	2	-1	0	0	-1	0																																																																																																																																																
B	-1	3	-1	0	-1	0																																																																																																																																																
C	0	-1	2	-1	0	0																																																																																																																																																
D	0	0	-1	3	-1	-1																																																																																																																																																
E	-1	-1	0	-1	3	0																																																																																																																																																
F	0	0	0	-1	0	1																																																																																																																																																

图 2.11 图的拉普拉斯矩阵

度数矩阵  $D$  被用来量化每个节点发送的边的数目，也就是节点的度数。对于给定顶点数、边数以及所有可能的连接点，如果某个节点的度数小于或等于该

顶点所代表的邻接权最大的顶点个数,那么这个节点就会与其它相邻节点发生连接关系。因此,矩阵  $D$  被构建成为一个对角矩阵,以刻画每个节点的度数信息。节点间的连接关系则通过对角矩阵  $A$  进行表达,其中,当节点间存在边连接时,对应元素值为“1”,在无连接节点间,相应元素值设定为“0”。拉普拉斯矩阵  $L$  是通过公式  $L=D-A$  来定义的,其中  $D$  和  $A$  分别代表度矩阵和邻接矩阵。经分析可知,  $L$  呈现出对称性质,且其每一行与每一列的元素和均为 0。

将 *Hadnard* 算法转换为矩阵乘法时,可以定义一个对角矩阵  $g_\theta = U^T y$ , 则式可以表示为。

$$f * g = U g_\theta U^T x \quad (2-19)$$

在式中,图卷积神经网络中的卷积核用  $g_\theta$  表示。从上述计算过程可以看出随着图数据的增大,图拉普拉斯矩阵的计算复杂度亦呈现出显著的上升趋势,同时计算量也变得相当庞大。因此,基于谱域的图数据处理方法在实际应用中,特别是在涉及大量图节点的场景中,面临着显著的应用局限性。

## (2) 基于空间的图卷积神经网络

Kipf 等人<sup>错误!未找到引用源。</sup>设计了一种基于空间的图卷积神经网络模型,此模型可以有效克服基于谱域的图卷积方法的缺陷。在构建图卷积模型的过程中,该模型将中心节点的数据及其邻近节点的数据进行整合,同时每个隐藏层后进行归一化处理,显著地降低了过拟合的风险。为了提升计算效率,该模型在图神经网络中运用了切比雪夫一阶展开项作为卷积核。当 GCN 的层数设定为  $n$  时,其输出表达式可表述为:

$$Z^{(n+1)} = F \left( \theta D^{-\frac{1}{2}} A D^{-\frac{1}{2}} Z^{(n)} \right) \quad (2-20)$$

其中  $A$  和  $D$  代表了图的邻接矩阵和度数矩阵,切比雪夫系数的向量用  $\theta$  表示,而  $F$  代表了激活函数 *RELU*。计算的具体流程如图 2.12 所示。

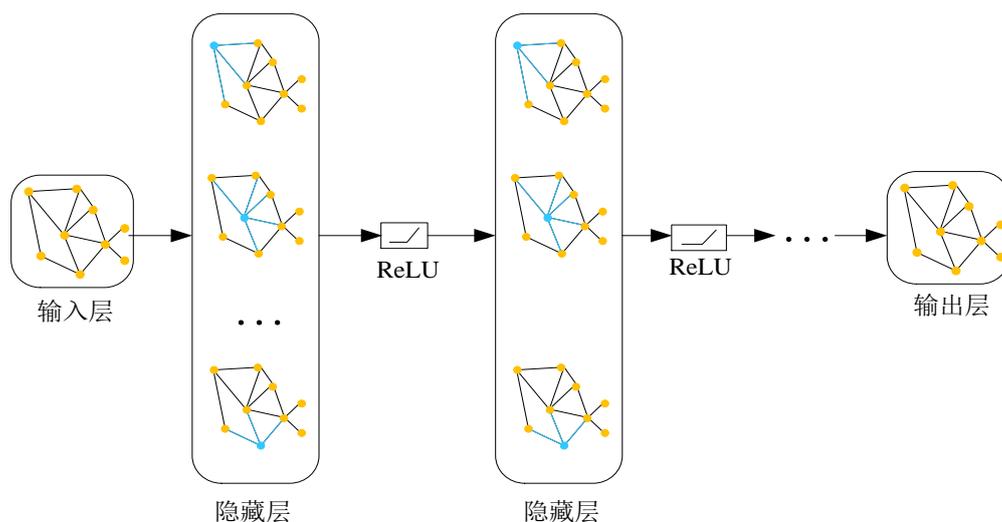


图 2.12 GCN 计算流程

在情感分析的过程中，我们将词向量视作图神经网络中的节点，词与词之间的共现关系则以节点间的连边形式展现。相较于传统序列化模型所采用的普通卷积神经网络，图卷积神经网络在挖掘非线性的语义信息方面展现出更为卓越的深度能力，从而更有效地提升模型的特征提取能力。

## 2.6 注意力机制

2014年，Bahdanau、Cho和Bengio在一篇研究机器翻译的论文中首次提出了注意力机制这一概念。2017年，谷歌发表了一篇名为“Attention is all you need”的研究机器翻译的文章，在这篇文章中作者在Transformer模型中加入了自注意力机制，该模型不再依赖递归或者卷积，根据实验结果来看，加入了自注意力机制的模型的性能优于之前性能最好的模型。2014年，德国雅各布大学的Bahdanau提出了一种注意力模型<sup>[51]</sup>，图2.13所示为融入注意力机制的模型图。

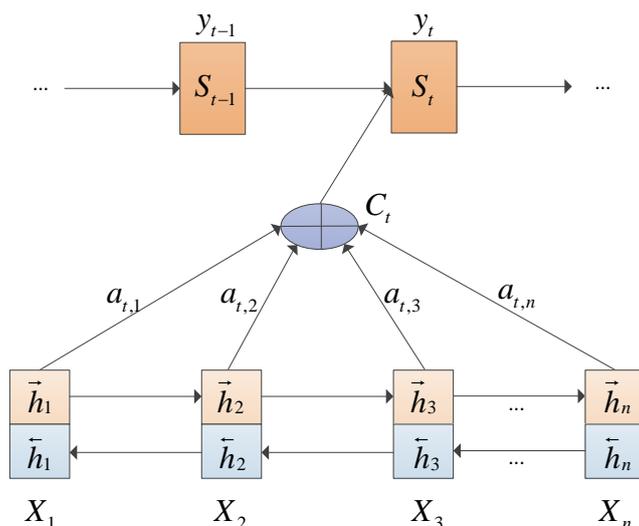


图 2.13 引入注意力机制的模型图

在现实世界中，人类关注某一物体时，往往会对不同方面有所侧重，这便是注意力机制的体现。在深度学习中，注意力机制被视作一个权重向量。通过数学运算，注意力机制能够区分目标元素与其他元素的不同重要程度。这种机制的实现依赖于 Encoder-Decoder 通用框架，该框架模拟了人类的认知过程。以学习新理论知识为例，人类首先需理解并掌握知识，然后通过自己的理解和表述将知识传授给他人。同样，Encoder-Decoder 框架在处理信息时，先通过编码器(Encoder)将输入信息转化为内部表示，再通过解码器(Decoder)根据内部表示生成输出。在此过程中，注意力机制发挥着关键作用，帮助模型更加聚焦于重要的信息，从而提高处理效率和准确性。这中间涉及一个知识转换再输出的过程。Encoder-Decoder 框架如图 2.14 所示。

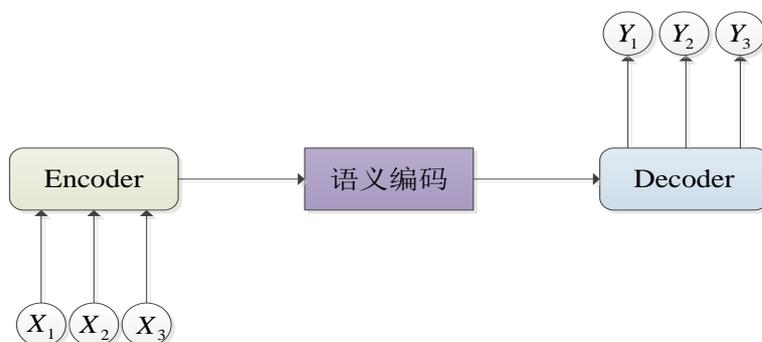


图 2.14 Encoder-Decoder 框架

自注意力机制是属于注意力机制的其中一种变体,对于距离较长的单词,自注意力机制也能够建立起它们之间的依赖关系,而且和之前的卷积神经网络和循环神经网络相比效率更高。自注意力模型所采纳的模式是“查询-键-值”(Q-K-V),Q、K、V以及最后的输出向量H的计算步骤如下式所示。

$$Q = W_q X \in \mathbb{R}^{D_k \times N} \quad (2-21)$$

$$K = W_k X \in \mathbb{R}^{D_k \times N} \quad (2-22)$$

$$V = W_v X \in \mathbb{R}^{D_v \times N} \quad (2-23)$$

$$H = V \text{soft max} \left( \frac{K^T Q}{\sqrt{D_k}} \right) \quad (2-24)$$

除了自注意力机制外,还有一种有效的注意力机制,那就是多头自注意力机制,这一机制通常被应用于 transformer 架构。首先采用不同的线性变换方法对输入序列进行调整,从而获得多组不同的Q、K、V向量。然后多头自注意力机制的不同注意力头对特征进行计算以得到输出向量,最后再线性变换得到的多组输出注意力向量,经过以上步骤后输出最终的多头自注意力。多头注意力机制使得模型的性能大大的提高了。

## 2.7 依赖句法分析

依赖句法分析的目标是将句子中不同词之间的依赖关系进行表示。依赖是指句中不同词语间支配与被支配的关系,其中其支配地位的词称为支配词(Head),被支配的词称为从属词(Dependency),支配词与从属词之间的关系就是依赖关系。句法分析是根据预先制定好的句法规则将语句中的成分关系表示为图结构。

在1959年,著名的语言学者泰尼埃尔在他的《结构句法基础》一书中,提出了一种普遍适用的结构句法通用理论。这一理论主张,每一个句子中都存在一个核心词汇,而这个核心词汇并不受句子中其他词汇的限制,反而在句子中起到了约束其他词汇的功能。这种语义关系称为依赖语法。基于此,计算机语言学专家Robison提出了四个依赖句法的公理<sup>[52]</sup>,这四个约束性公理为依赖句法分析提供了坚实的基础。这四条公理分别是:

1. 根节点的唯一性:有且仅有一个词语独立存在,该词语不依赖句子中任

何词语。

2. 节点的连通性：除了根节点，其他所有的词语都有其依赖的词语。

3. 节点的无环性：句子中每个词语只能依赖其中一个词语，不存在依赖多个词语的情况。

4. 节点的投射性：对于两个存在依赖关系的词语，这两个词语间其他词语的依赖词只能是两个词或者两个词之间的其他词。

基于以上公理，研究人员开始制定标注标准。目前，进行句法分析使用的是DDParser (Baidu Dependency Parser)和LTP (Language Technology Platform) 具体的标签体系如表 2.1 所示：

表 2.1 依赖句法分析关系标注表

关系类型	标注	描述	举例
主谓关系	SBV	subject-verb	我爱中国（我←爱）
动宾关系	VOB	直接宾语, verb-object	我借他一支笔（借→笔）
间宾关系	IOB	间接宾语, indirect-object	我借他一支笔（借→他）
前置宾语	FOB	前置宾语, fronting-object	他什么菜都吃（菜←吃）
兼语	DBL	double	他送我回家（送→我）
定中关系	ATT	attribute	大西瓜（大←西瓜）
状中结构	ADV	adverbial	特别努力（特别←努力）
动补结构	CMP	complement	看完了课本（看→完）
并列关系	COO	coordinate	我和你（我→你）
介宾关系	POB	preposition-object	在学校里（在→里）
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
核心关系	HED	head	句子的核心词

根据前述的句法标记准则，我们可以用带有方向的曲线来描述给定句子之间

的相互关系。在示例句子中，“班主任李某检查作业时提出，鼓励学生大胆提出问题。”，在一个完整的句子结构中，“李某”充当主语，而“检查”则是谓语动词，同时也是句子的核心词。这两个词构成了一个主谓关系。“班主任”这一修饰词用于修饰“李某”，而“提出问题”则作为“鼓励”的宾语使用。从上面提到的关系出发，选择关系的主导者作为有向弧的起始点，我们可以观察到依赖关系如图 2.15 所示。

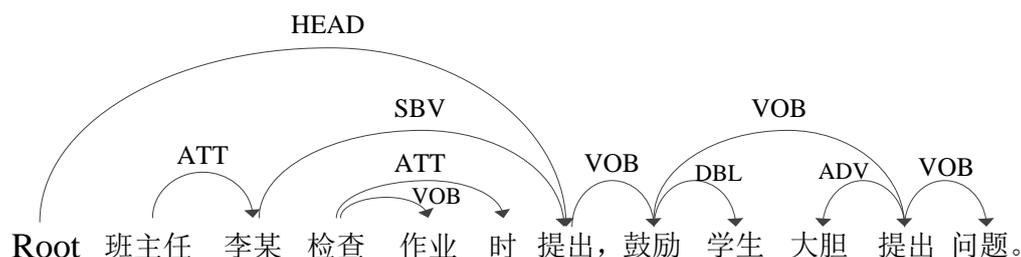


图 2.15 依赖句法分析图

不管是在英语、法语、中文或是其他多种语言中，句子中的词汇之间的依存关系都是普遍存在的。因此，在自然语言处理领域，依赖句法分析已经变成了一种普遍的分析手段。目前，有多种常用的依赖句法分析工具，包括 Python 文本预处理库 spaCy，百度自然语言处理部基于深度学习平台飞桨以及大规模标注数据研发的 DDParse 等。

## 2.8 本章小结

本章对情感分析领域的相关理论与技术进行了详尽的阐述。首先，对情感分析进行了概括性介绍，随后深入探讨了基于 Word2Vector、GloVe 和 BERT 的词向量表示技术。紧随其后，本章深入探讨了如循环神经网络、卷积神经网络、注意力机制和依赖句法分析等关键技术。在循环神经网络的讨论中，特别关注了长短时记忆网络的应用；在卷积神经网络的介绍中，详细解释了卷积原理及图卷积神经网络的原理；在依赖句法分析部分，则对句子间普遍存在的依赖关系进行了深入探讨。这些内容将为后续情感分析模型的构建奠定坚实的理论与技术基础。

### 3 方面级情感分类的双重注意力模型

在对句子的语法进行提取上，除了依赖弧的连接信息外，“直接宾语”“形容词的修饰语”等依赖弧的标签也是提供语法信息的非常重要的组成部分。将依赖弧标签转化为模型的嵌入，对句子进行注意力操作，以此对这句话的重要部分进行捕获。基于上述想法本章构建了基于 BERT 的方面级情感分析的双重注意力模型（A dual attention model for aspect level sentiment analysis based on BERT）。其模型结构如图 3.1 所示。此模型使用预训练的 BERT 作为编码器，在准确率和 F1 值方面带来了巨大的性能提升，双重注意力模型结构图如下图 3.1 所示。

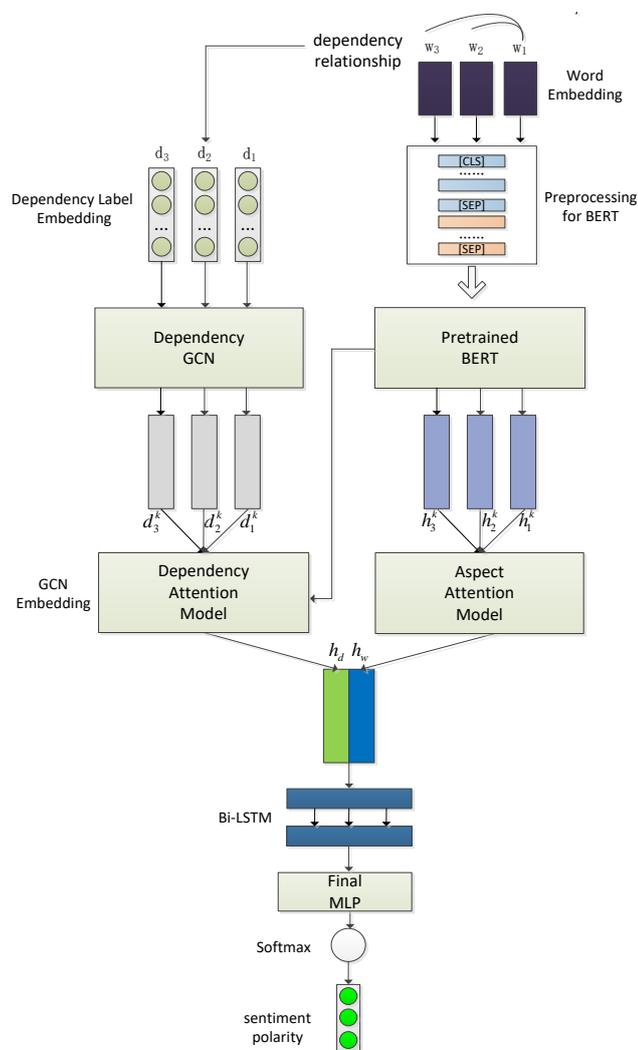


图 3.1 双重注意力模型结构图

模型由依赖解析层(dependency Parsing Layer)、嵌入层(Embedding Layer)、编码器层(Encoder Layer)、GCN 层、注意力机制层(Attention Mechanism Layer)、Bi-LSTM 层以及最终 MLP 层组成。第一步对句子中的词语作分词以及解析出关系依赖树(Dependency Tree),然后将词语本身意思输入到模型的上面 BERT 的部分,将解析出的关系依赖树中的每个词的语义标签(the label of the Dependency Arc)输入到下面的注意力模块(GCN+Attention)。第二步经过预训练的 BERT 将输入的词训练编码为对应的词向量(Word Embedding),此词向量包含了词语在句子中丰富的上下文语义信息。第三步将每个词的语义标签(the Label of the Dependency Arc)使用训练好的 Glove 嵌入转化为标签嵌入向量(Label Embedding),然后通过 GCN 层以及注意力模块得到词语语义标签的带有注意力权重的输出。第四步将二、三步得到的输出  $h_w$  和  $h_d$  首尾相连,得到句子的词向量的表示,然后通过 Bi-LSTM 做进一步的处理,得到包含更丰富的语义和语法信息的句子词向量,最后通过 MLP 层进行最终的情感预测分类。

### 3.1 任务定义

给定一个由  $N$  个单词组成的句子  $s = \{w_1, w_2, \dots, w_N\}$  和一个方面  $a = \{w_{a1}, w_{a2}, \dots, w_{am}\}$ , 方面级情感分类的任务是判断句子对方面的情感极性。虽然情感极性的种类是由数据集决定的,但情感极性总是有积极、中性和消极三种。其中  $w_i$  代表句子中的单词,  $w_{ai}$  代表某个方面的单词,该方面可以由许多单词组成。

### 3.2 依赖解析层

依赖解析层解析句子中单词之间的依赖关系,得到的依赖关系包括依赖弧和依赖弧的标签。依赖弧可以理解为两个词之间的连接关系,并且这种连接关系是有向的。虽然在进行处理时,本模型忽略了依赖弧的方向,只使用了连接关系。根据依赖弧提供的连接信息,本模型设置了一个存储于矩阵  $G$  中的图,并将该图用于词图卷积块。对于这个矩阵,如果一个句子有  $n$  个单词,那么每个单词将

依赖于一个单词，因此存在  $n$  个依赖弧标签和依赖关系。忽略连接关系，如果单词  $w_i$  依赖于单词  $w_j$ ，那么  $G_{ij}$  和  $G_{ji}$  将会被填充，这表示连接关系。

### 3.3 嵌入层

本模型设置了两种嵌入层。一种是词嵌入层，用于将词转化为特定的词向量。另一种是依赖标签嵌入层，用于将依赖关系的标签转化为特定的向量表示。

在词嵌入层，使用 Glove 嵌入将单词转化为词向量。本模型构建了一个词向量矩阵  $M^{V \times \text{dim}_w}$ ，其中  $V$  代表数据集中词的种类数， $\text{dim}$  表示词向量的维度。使用 Glove 嵌入对词向量矩阵进行初始化，通过将输入与词向量矩阵相乘最终得到词向量  $\{v_1, v_2, \dots, v_n\}$ ，其中  $v_i \in R^{\text{dim}_w}$ 。

依赖标签嵌入层和词嵌入层一样。为了将依赖标签转化为特定的向量表示，本模型还将构造一个矩阵  $M^{V' \times \text{dim}_l}$ ，其中  $V'$  是特定数据集中依赖结果中依赖标签的种类数量， $\text{dim}_l$  表示转换后的向量表示的维度。这一层输出的是关于依赖弧标签的向量表示  $\{d_1, d_2, \dots, d_n\}$ ，其中  $d_i \in R^{\text{dim}_l}$ 。

### 3.4 编码器层

为了学习更充分的语义信息，本模型使用预训练的 BERT 作为编码器。在处理句子信息的部分中，BERT 模型的输入是语句  $\{w_1, w_2, \dots, w_n\}$ ，这些符号表示字符串而不是词向量。为了将文本转换为 BERT 可以处理的形式，我们首先使用 BERT Tokenizer 将输入的文本划分成一个个标记。但是此标记不是标记整个单词，而是标记单词的词根或词缀。在处理过程中，对句子的整体和方面分别进行了标记，因此得到了两部分结果。然后对得到的两部分结果进行合并，在两个结果之间添加[SEP]分隔符，在头部添加[CLS]分隔符，在尾部添加[SEP]分隔符。然后就得到了 BERT 模型的一个输入，另一个输入是区分两个分段结果的掩码。将其输入到预训练的 BERT 模型中，BERT 模型通过其强大的双向 Transformer 结构对输入进行处理，得到两个输出。一个是池化向量  $v_{pool} \in R^{\text{dim}}$ ，其中  $\text{dim}$  代表了

BERT 模型的隐藏层大小。另一个输出是模型最后一层句子标记化输入的隐藏状态  $h_{final} \in R^{m \times \text{dim}}$ 。

池化向量  $v_{pool} \in R^{\text{dim}}$  是可以直接输入到最终 MLP 层进行情感分类的特征向量。但这种做法将使模型成为纯粹的 BERT 模型，为了利用丰富的句法的信息，对最终的隐藏状态  $h_{final} \in R^{m \times \text{dim}}$  进行依赖性注意。因为根据 BERT 独特的分词方式，隐藏状态  $h_{final} \in R^{m \times \text{dim}}$  是  $m$  而不是句子中的单词数  $n$ 。所以首先将向量的长度恢复到原始长度，然后得到隐藏状态向量  $h'_{final} \in R^{n \times \text{dim}}$ 。然后做与原始模型相同的依赖注意力。最终将  $v_{pool}$  和  $v'_{final}$  连接起来以获得一个新的特征向量。将特征向量输入到 MLP 层以获得最终的情感极性。其中， $v_{pool}$  和  $h'_{final}$  向量在图中分别为  $h_w$  和  $h_d$ 。

### 3.5 GCN 层

在前面的依赖解析层中，通过依赖关系的连接信息，得到了连接图  $G \in R^{n \times n}$ ，其中  $n$  表示句子中单词的数目，矩阵的项只有 1 和 0，代表了两个词之间是否连接。在这个图卷积网络中，模型将使用图和特征向量进行特殊计算。GCN 中有两部分，即单词 GCN 和依赖关系 GCN。

在单词 GCN 块中，模型把  $h_s \in R^{n \times 2d_n}$  和  $G \in R^{n \times n}$  输入到图卷积网络中，单词 GCN 块的输出是  $h_s^k \in R^{n \times \text{dim}_{wordgcn}}$ ，其中  $k$  表示单词 GCN 的层数， $\text{dim}_{wordgcn}$  表示最终单词 GCN 层的输出维度， $h_s^k$  是  $[h_1^k; h_2^k; \dots; h_n^k]$  集合对应的整体结构图。图卷积网络对应的详细信息如下式所示。其中， $l$  表示层， $H$  代表整个层的隐藏状态， $\sigma$  是激活函数， $w$  是神经网络的参数， $\hat{D}$  是做归一化的度矩阵， $\hat{A}$  是图的邻接矩阵  $A$  和对角矩阵  $I$  的和。简要地说，图卷积网络可以处理图中节点之间的连接关系。如果单词  $w_i$  依赖于单词  $w_j$ ，那么单词  $w_i$  与单词  $w_j$  的信息融合到自身向量中，图卷积网络作为一项强大的技术可以实现此操作。并且利用深层 GCN，单词的特

征可以被广泛传播。

$$H^{(l+1)} = \sigma \left( \hat{D}^{-\frac{1}{2}} \hat{A} \hat{D}^{-\frac{1}{2}} H^{(l)} W^{(l)} \right) \quad (3-1)$$

依赖标签 GCN 块和单词 GCN 块相似，但输入的是关于依赖标签的向量而不是关于单词的向量。在单词 GCN 块中，模型将  $D \in R^{n \times \dim_l}$  和  $G \in R^{n \times n}$  输入到图卷积网络中，其中  $D$  是集合  $\{d_1, d_2, \dots, d_n\}$  且  $d_i \in R^{\dim_l}$ 。依赖标签 GCN 块的输出是  $D^k \in R^{n \times \dim_{dep\text{gcn}}}$ ，其中  $k$  代表 GCN 层的数量， $\dim_{dep\text{gcn}}$  代表依赖标签 GCN 层的输出维度。对于依赖关系，如果  $w_i$  依赖于  $w_j$ ， $w_j$  依赖于  $w_k$ ，则  $w_i$  也依赖于  $w_k$ 。该模型利用依赖标签向量对句子向量进行注意，由于  $w_i$  和  $w_j$  或  $w_j$  和  $w_k$  之间只有一个连接，模型对句子的潜在意义不能准确把握。但是利用图卷积网络可以挖掘这种深层次的依赖标签信息。

### 3.6 注意力机制层

注意力机制模块是双重注意力模型中最重要的部分。该模型中有两种注意力模块，方面注意力模块和依赖注意力模块。这两个模块分别关注不同的部分，方面注意力模块将方面嵌入作为查询（Query），句子特征向量作为键值（Key）和值项（Value），专注用于寻找与特定方面有密切关系的特征。而依赖注意力模块以图卷积网络处理的依赖标签特征作为查询（Query），以句子特征向量作为键值（Key）和值项（Value），着重于在句子中寻找最能代表情感极性的词语。在双重注意力模型中，使用了方面注意力和依赖注意力两种模块并行去推测情感极性，最后将经过两种注意力模块后的特征向量连接到最终的 MLP 层进行情感分类。

方面注意力模块使用点积注意力机制。此注意力机制的查询是方面嵌入  $h_a \in R^{m \times 2d_h}$ ，这种注意力机制的键值和值项是词图卷积网络  $h_s^k \in R^{n \times \dim_{word\text{gcn}}}$  的输出。在模型设置中， $\dim_{word\text{gcn}}$  和  $2 \times d_h$  的值项相等。因此，在方面注意力模块中，使用点积注意力作为下面的等式。该方面注意力模块的输出是  $h_w \in R^{2d_h}$ 。

$$aspect - att.(h_a, h_s^k, h_s^k) = \text{soft max} \left( \frac{h_a (h_s^k)^T}{\sqrt{2d_h}} \right) h_s^k \quad (3-2)$$

与方面注意力模块不同，依赖注意力模块使用附加注意力机制。这种注意力机制的查询就是依赖标签图卷积网络  $D^k \in R^{n \times \text{dim}_{\text{depen}}}$  的输出。依赖注意力模块的键值和值项是 *Bi-LSTM*  $h_s \in R^{n \times 2d_h}$  输出词的隐藏状态。因此，在依赖注意力模块中，下式中使用了附加注意力。 $W_q$ ， $W_k$  和  $W_v$  是可学习的参数，此依赖注意力模块的输出是  $h_d \in R^{2d_h}$ 。

$$dp - att.(D^k, h_s, h_s) = w_v^T \tanh(W_q D^k + W_k h_s) h_s \quad (3-3)$$

### 3.7 Bi-LSTM 层

将从预训练的 BERT 模型和依赖注意力层输出的向量  $h_w \in R^{2d_h}$  和  $h_d \in R^{2d_h}$  首尾相连，得到句子的词向量表示，然后通过 Bi-LSTM 层进行进一步的加工和处理，得到包含语义和语法信息更为丰富的新的句子词向量  $h^* \in R^{4d_h}$ ，这是最终 MLP 层的输入。

### 3.8 最终 MLP 层

最终的多层感知器是通过几个简单的全连接层来进行分类。在方面注意力模块和依赖注意力模块中，得到了结果  $h_w \in R^{2d_h}$  和  $h_d \in R^{2d_h}$ 。为了进行最终的分类，将它们集中成一个新的特征向量  $h^* \in R^{4d_h}$ ，这是最终 MLP 层的输入。在最终的多层感知器中，可以得到输出  $y' \in R^3$ ，公式如下。

$$y' = \text{relu}(W_2 \text{relu}(W_1 h^* + b1) + b2) \quad (3-4)$$

将最终的结果在 *Soft max* 层进行归一化处理，公式如下所示。

$$\hat{y} = \text{soft max}(y') \quad (3-5)$$

### 3.9 本章小结

本章提出了方面级情感分类的双重注意力模型，模型主要包括七层结构，分为依赖解析层、嵌入层、编码器层、GCN 层、注意力机制层，模型编码器使用了 BERT，并且在注意力机制层后加入了 Bi-LSTM 进行更充分的语义学习。注意力机制模块是双重注意力模型中最重要的部分。该模型中有两种注意力模块，方面注意力模块和依赖注意力模块。这两个模块分别关注不同的部分，从而带来了巨大的性能提升。

## 4 实验与分析

### 4.1 数据集介绍

本文的数据集选择了提供于国际语义测评大会的 SemEval 2014 Task4 的参观评论 (Restaurant) 和笔记本电脑评论 (Laptop) 子数据集以及 Twitter 短文本评论数据集。在这三个数据集上对模型的有效性进行验证。在这个数据集里, 每一条评论都是专门针对某一特定目标, 并带有相应的情感倾向。Restaurant 的数据集详细描述了餐厅的服务质量和食物的口感等关键目标; Laptop 数据集包含了关于电脑的价格、质量等多个目标的信息。通过这两个数据集的主要相似之处在于, 它们的评论大部分都是正面的。然而, 与 Laptop 数据集相比, Restaurant 数据集包含的隐式情感表达样本相对较少, 因此其数据量也相对较大。Twitter 数据集中数据的数目有六千多条且大部分评论的情感极性都是中性, 和另外两个数据集相比, Twitter 数据集的样本质量较差, 在分析起来的难度较高。把三个数据集划分为测试集和训练集, 并且每个数据都可分为积极、中性和消极三个情感极性。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

为了评估模型的性能, 模型采用两个评估指标: 精确度 (Accuracy) 和 F1 值进行综合评估。

在处理情感分类这类二分类任务时,根据实际类别和学习器预测的类别差异,可以将样例分类为四大类:真正例(True Positive, TP)、假正例(False Positive, FP)、真反例(True Negative, TN)、假反例(False Negative, FN)。真正例表示在对标签为正类的样本进行预测时,其预测结果也表现为正类,假正例则是样本的标签是正类,但是,模型的预测结果变成了负类。真反例是指在预测负类结果时,模型的预测结果也是负类,而假反例是指在预测负类结果时,模型的预测结果是正类。其模型预测结果表现的是正类。则样例总数可表示为:  $TP+FP+TN+FN$ 。表为分类结果的混淆矩阵(Confusion Matrix)。

准确率(Precision)是基于预测结果与实际情况均为正的样本,与所有预测结果均为正的样本进行比较,计算公式如所示。

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

召回率(Recall)用来表示在所有预测为正类的样本中真正为正类的占总体实际为正类的比例。计算公式如所示。

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

Accuracy 通过计算分类正确的样本数与样本总数的占比来评估模型分类性能的优劣。

理想情况下 Precision 和 Recall 值同时越高越好,但实际操作时在某种特殊情况下两者值无法同时升高。所以只使用 Precision 值进行评估或者只使用 Recall 值进行评估都无法准确地评估模型性能的好坏。因此采取综合指标 F1-Score 来进行评估, F1 值可以对两者值无法同时升高的情况进行很好地平衡。F1 值的计算公式如下:

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

最终才用 Accuracy 值和 F1 值来进行综合评估。

### 4.3 实验环境与超参数配置

使用交叉熵损失函数为目标函数对模型进行训练。 $\hat{y} \in R^3$  是预测的情感分布,

$y \in \mathbf{R}^3$  是真实的情感分布。 $\lambda$  是  $L_2$  的正则化项,  $\theta$  是参数设置。

$$\mathcal{J}(\theta) = -\sum_i \sum_j y_i^j \log \hat{y}_i^j + \lambda \|\theta\|^2 \quad (4-4)$$

在对模型进行训练的过程中, 使用了 AdamW 优化器来优化训练过程。

所有实验均使用 Pytorch 深度学习框架、Ubuntu 20.04 系统 I9-10900XCPU、NVIDIA GeForce RTX 3080 TI 的显卡、12G 显存。

Glove 嵌入的嵌入维度设置为 300, 依赖标签的嵌入维度设置为 300, Bi-LSTM 的隐藏层维度设置为 100。预训练 BERT 模型的隐藏层维度设置为 768, 模型的学习率设置为 0.003, 训练批量大小设置为 32, Adam 优化器的 Adam epsilon 设置为 10 的负 8 次方。除了这些参数之外, 模型中的其他参数是通过从均匀分布  $U(-E, E)$  中采样来初始化的。

## 4.4 对比模型设置

将双重注意力模型与方面级情感分析领域的其他模型做对比, 所有模型都使用一样的数据集。下面对这些模型进行简要描述, 具体的比较结果如表所示。

ATAE-LSTM<sup>[66]</sup>: 利用方面嵌入和注意力机制进行方面级情感分类。在对 LSTM 层进行输入时将方面和词嵌入串联进行搭配使用, 注意力机制层用来提高模型对方面词的权重。

IAN 错误!未找到引用源。: 利用两个 LSTM 和一个交互注意力机制去生成方面和句子的表示。在学习上下文和方面词的注意力分布时使用了交互式学习方式, 最后求其情感特征时将两部分拼接起来。

ASGCN<sup>[68]</sup>: 首次建议采用 GCN 来学习方面级情感分类的特定方面的表示的工具。

RAM<sup>[69]</sup>: 使用多头注意和记忆网络来学习句子表示。在得到 Bi-LSTM 的隐藏层输出后, 在隐藏层位置加权记忆, 最后使用注意力机制进行非线性组合, 以此来得到情感特征。

AEN<sup>[71]</sup>: 设计了一个注意力编码器网络来绘制目标词和上下文词之间的隐藏状态和语义交互。

kumaGCN 错误!未找到引用源。: 采用潜在图结构来补充句法特征。

R-GAT<sup>[73]</sup>:提出了一种面向方面的依赖树结构, 对一个普通的依赖树进行重塑和修剪, 然后用新的 GAT 对依赖树进行编码。

DualGCN 错误!未找到引用源。:该模型是方面级情感分析任务中最先进的模型。使用 SynGCN 和 SemGCN 网络来执行方面级情感分类任务。

Pure BERT<sup>[75]</sup>:通过输入句子和方面并对使用[CLS]的表示进行预测, 是一种普通的 BERT 模型。

对于性能方面的比较, 双注意力模型彻底击败了例如: ATAE-LSTM<sup>[66]</sup>、IAN<sup>[67]</sup>、RAM<sup>[69]</sup>和 AEN<sup>[71]</sup>等基于注意力机制的模型。与 ASGCN<sup>[68]</sup>和 kumaGCN<sup>[72]</sup>等早期使用图神经网络的模型相比, 我们在某些数据集上的性能并不逊色, 甚至超越这些模型。双重注意力模型在餐厅数据集中的准确性优于 ASGCN<sup>[68]</sup>。并且双重注意力模型在 Twitter 数据集中的准确率和 F1 值优于 ASGCN<sup>[68]</sup>和 kumaGCN<sup>[72]</sup>。虽然双重注意力模型的性能与这两年的顶级模型<sup>[11]</sup>的性能有一定的差距。但在这两年基于 BERT 的模型中, 双重注意力模型在这两年的顶级模型中有着精彩的表现。基于 BERT 的双重注意力模型的性能比 Pure-BERT<sup>[75]</sup>模型的性能要更好, 这从侧面证明了依赖注意力模块的有效性。基于 BERT 模型的性能优于 AEN-BERT、R-GAT+BERT, 并且接近于 Dual-GCN<sup>[74]</sup>。

从上面的比较分析来看, 基于 BERT 的双重注意力模型对于解决方面级的情感分类任务是有用的。另外, 该模型还有很大的提高和进步空间。

表 4.2 模型实验结果对比

Models	Restaurant		Laptop		Twitter	
	Acc	F1	Acc	F1	Acc	F1
ATAE-LSTM	77.2		68.7			
IAN	78.6		72.1			
RAM	80.23	70.8	74.4	71.35	69.36	67.3
AEN	80.98	72.14	73.51	69.04	72.83	69.81
ASGCN	80.77	72.02	75.55	71.05	72.15	70.4
KumaGCN	81.43	73.64	76.12	72.42	72.45	70.77
R-GAT	83.3	76.08	77.42	73.76	75.57	73.82
DualGCN	84.27	78.08	78.48	74.74	75.92	74.29
Pure BERT	85.09	77.88	77.74	73.71	74.71	73.54
AEN-BERT	83.12	73.76	79.93	76.31	74.71	73.13
R-GAT+BERT	86.6	81.35	78.21	74.07	76.15	74.88

续表 4.2

Models	Restaurant		Laptop		Twitter	
	Acc	F1	Acc	F1	Acc	F1
DualGCN+BERT	87.13	81.16	81.8	78.1	77.4	76.02
<b>Dual-Attention Model</b>	<b>86.63</b>	<b>80.71</b>	<b>79.53</b>	<b>76.35</b>	<b>75.13</b>	<b>73.97</b>

## 4.5 实验结果及分析

### 4.5.1 消融实验

双重注意模型和 Non-Dep 模型：为了观察依赖注意力的依赖性和有效性，本文删除了依赖嵌入层、依赖 GCN 和依赖注意力模块来创建一个名为 Non-Dep 的模型的新模型。在对这个模型进行训练，最终得到了如下表所示的结果。正如表所示那样，没有依赖模块的模型在所有三个公共数据集中都与原始模型存在差距。这表明依赖注意力模块有助于提高模型的性能。

其他结果比较：在模型构建过程中，将方面注意力和依赖注意力串行而不是并行，然后将它们连接起来。在这个模型中，除了依赖注意力模块和方面注意力模块之外，其他层都照常工作。在单词 GCN 层之后，依赖特征由依赖嵌入层处理，依赖 GCN 层用于进行依赖关注。并将依赖注意力的结果输入到方面注意力模块中，得到最终的表示向量，输入到最终的 MLP 进行分类。但正如下面的所示，这个模型的性能并不是特别理想。双重注意力模型性能比这个更好。所以最后选择了将方面注意力和依赖注意力并行并连接结果向量的模型作为本文设计。

表 4.3 双注意力模型和 non-dep 模型实验结果比较

Models	Restaurant		Laptop		Twitter	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
Non-dep model	79.37	68.68	73.67	68.29	71.1	69.05
<b>My dual-att. model</b>	<b>81.25</b>	<b>72.53</b>	<b>75.39</b>	<b>71.16</b>	<b>73.12</b>	<b>72.05</b>

表 4.4 双注意力模型和其他模型实验结果比较

Models	Restaurant		Laptop		Twitter	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
Another designed model	80.09	68.79	72.73	67.8	72.25	69.89
<b>My dual-att. model</b>	<b>81.25</b>	<b>72.53</b>	<b>75.39</b>	<b>71.16</b>	<b>73.12</b>	<b>72.05</b>

## 4.5.2 GCN 层数的影响

GCN 层数对于引入到模型的相关信息有着重要的作用。GCN 层数设置不同，则模型的预测性能也会产生不同。通过在其他部分不变的条件下，设置不同的 GCN 层数来观察模型在三个数据集上的准确率以及 F1 值，以此来判断 GCN 层数对模型预测性能的影响。在实施过程中，将 GCN 层数从 1 设置到 7。实验结果如图 4.5 和 4.6 所示。

从实验情况可以发现，当 GCN 的层数为 3 的时候，模型在三个数据集上的准确率和 F1 值为最高。当层数从 3 层增加到 7 层时，模型在三个数据集上的准确率和 F1 值都出现不同程度的下降。其原因是当 GCN 层数设置太少时，GCN 网络对关键特征信息的获取不充分，从而导致模型性能较差。而当 GCN 层数设置太多时，模型中引入的其他关系增多，文本中会更新大量参数，导致模型不能够更好地训练，所以模型预测的准确率降低。

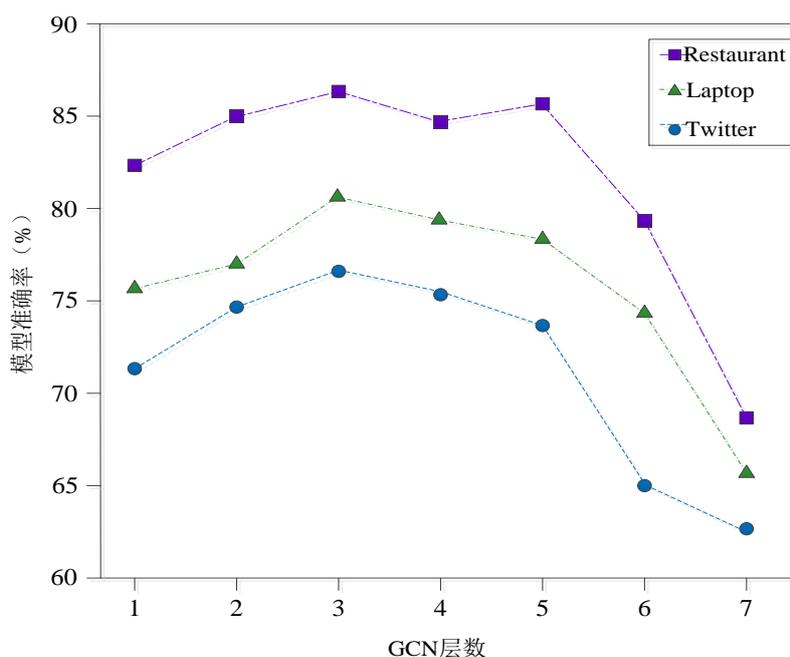


图 4.1 GCN 层数对模型预测的影响

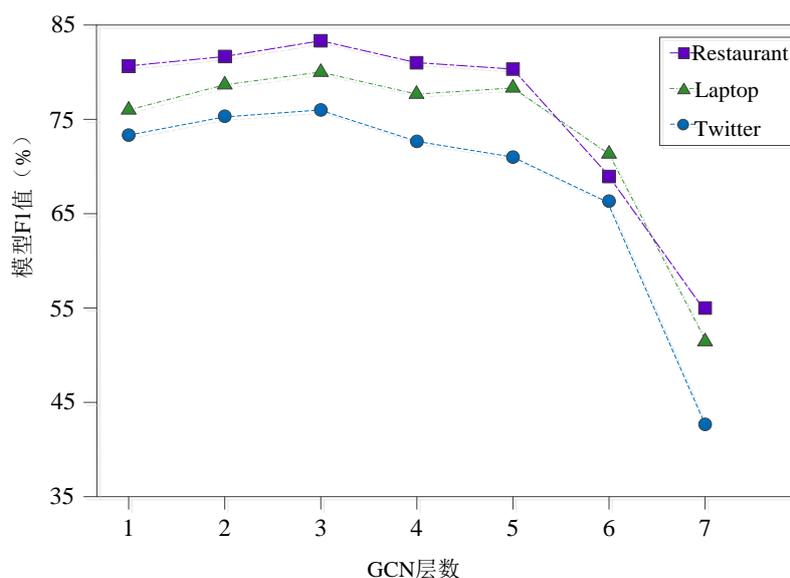


图 4.2 GCN 层数对模型预测的影响

### 4.5.3 案例研究

为了研究双重注意力模型的推理过程和两个注意力模块的作用，对“Despite the quality is excellent, the services are disappointing.”这句话进行了案例研究。在这句话中，有两个方面需要分析，“质量”的情感极性是正的，“福利”的情感极性是负的。方面注意力模块和依赖注意力模块是双重注意力模型中最重要的两个模块。因此，将方面注意力和依赖注意力的注意力权重可视化，如图 4.1 和图 4.2 所示。图 4.1 表示热图，左图是注意方面的重量，右图是依赖注意的重量。图 4.2 表示“services”方面的注意热图，左图是方面注意的权重，右图是依赖注意的权重。在“quality”方面的注意力模块中，“quality”的注意力权重较大，但对于“services”方面，句子后半部分的注意力权重较大。在依赖注意力模块中，您可以发现方面“quality”和方面“services”的注意力权重是相同的。而依赖注意力模块则更关注“优秀”“很少”“不好”等情感信息丰富的词。

通过分析单词位置上的注意力权重，我们可以发现方面注意力模块总是使得该模型关注与方面接近的特征。而依赖注意力模块始终使模型关注具有丰富情感信息的特征。

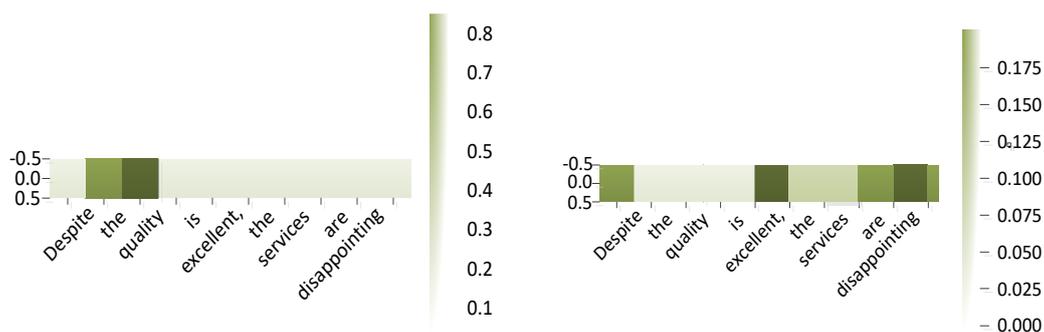


图 4.3 “quality” 方面的注意热图

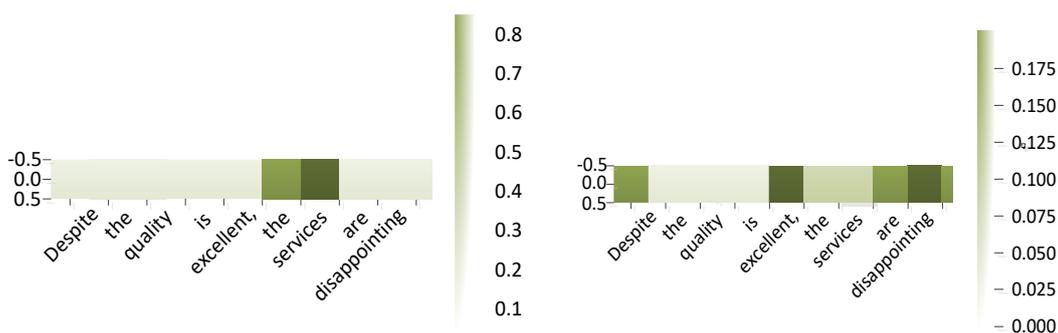


图 4.4 “services” 方面的注意热图

## 4.6 本章小结

这一章节详尽地阐述了模型实验测试的各个方面。首先介绍了实验中使用的三个公开数据集，并对这三个数据集进行了预处理。随后，解释了用于评估和比较模型性能的评价指标，并对实验的环境和配置进行了说明。然后在相同的数据集上，将双重注意力模型与方面级情感分析领域的其他九个模型做对比，最终证明了模型的有效性。在进行对比实验中，设置了一个删除了依赖嵌入层、依赖GCN和依赖注意力模块的Non-Dep模型，以此来验证依赖注意力模块的有效性。在另外一个模型中改变了两个注意力模块的运行方式，尝试串行进行方面注意力和依赖注意力，与原模型进行对比，证明了将方面注意力和依赖注意力串行是效果最理想的。最后，为了研究模型的推理过程和两个注意力模块的作用，进行了实例研究。

## 5 总结与展望

### 5.1 工作总结

随着互联网的迅猛发展，人们的生活方式发生了深刻变革。互联网已深入参与诸多人类活动，包括人们对热点事件的看法和观点、电商平台上的购物行为以及舆论事件的管控等。因此，在当今的自然语言处理领域，识别评论语句中隐含的情感态度，其精确性与速度已成为公众关注的焦点。情感分析在电子商务领域展现出了明显优势，它不仅为消费者提供了购买的建议和风险规避的信息，同时也为商家在产品质量反馈和销售策略的制定上提供了宝贵的建议。此外，情感分析为决策过程中的数据提供了坚固的后盾，确保了网络环境与社会日常生活的顺畅运行。

本文提出了具有方面注意模块和依赖注意模块的模型来解决方面级别的情感分类任务。依赖注意力模块嵌入依赖标签信息，对单词特征进行注意力机制，捕捉情感倾向较强的单词。实验表明，新提出的依赖性注意力模块是有效的并且正确的设计。并且为了增强原始模型的性能，使用预训练的 BERT 模型来替换 Glove 嵌入层。基于 BERT 的模型在三个公共数据集上表现出了出色的性能。通过在三个公开的数据集上进行实验，并与 10 个具有代表性的对比模型进行比较，我们发现本文提出的模型在准确性和 F1 值方面都实现了不同程度的提升，从而证实了该模型的有效性。为了更深入地探究注意力层在模型性能中的作用，设置了一个删除了依赖嵌入层、依赖 GCN 和依赖注意力模块的模型以及改变了两个注意力模块的运行方式来验证注意力层对模型性能的提升作用。进行实例研究双重注意力模型的推理过程和两个注意力模块的作用。

### 5.2 未来展望

本文提出的双重注意力模型虽然较以往只使用注意力的模型在性能上有了很大的提升，但和目前效果最好的模型相比还有很大的提升空间。

(1) 造成这种差距的一部分原因是模型中的某些网络设计不完美，导致神经网络收敛到全局最小值的效果较差。我认为如何使用依赖解析弧和标签值得深

入讨论。

(2) 用户在表达自己的观点时，经常会在他们的评论中加入如表情、符号和颜文字等非官方的元素，这些元素很容易反映出发表者的情感偏好。因此，探索如何使用高效的算法来精确地描述这些特性，并确保它们与模型完美结合，已经变成了提高模型预测精度的核心研究领域。

(3) 本文在提取句子依赖关系时存在一些依赖解析的错误，这些错误的信息可能会对模型的预测产生影响。研究如何减少错误信息的呈现并减轻其带来的负面效应，对于提高模型的整体表现是非常有益的。

## 参考文献

- [1] Zhang L, Wang S, Liu B. Deep learning for sentiment analysis: A survey[J]. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2018, 8(4): e1253.
- [2] 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.
- [3] Liu B. Sentiment analysis: Mining opinions, sentiments, and emotions [M]. Cambridge university press, 2020.
- [4] Nakagawa H, Torisawa K, Kitsuregawa M. WWW 2008 workshop: NLP2008 summary[C]//Proceedings of the 17th international conference on World Wide Web. 2008: 1277-1278.
- [5] Yu J, Zha Z J, Wang M, et al. Aspect ranking: identifying important product aspects from online consumer reviews[C]//Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies. 2011: 1496-1505.
- [6] Choi Y, Cardie C. Learning with compositional semantics as structural inference for subsentential sentiment analysis[C]//Proceedings of the 2008 conference on empirical methods in natural language processing. 2008: 793-801.
- [7] 曾锋, 曾碧卿, 韩旭丽, 等. 基于双层注意力循环神经网络的方面级情感分析[J]. 中文信息学报, 2019, 33(6): 108-115.
- [8] Li Z, Zou Y, Zhang C, et al. Learning implicit sentiment in aspect-based sentiment analysis with supervised contrastive pre-training[J]. arXiv preprint arXiv:2111.02194, 2021.
- [9] Popescu A M, Etzioni O. Extracting product features and opinions from reviews[J]. Natural language processing and text mining, 2007: 9-28.
- [10] Bagheri A, Saraee M, De Jong F. Care more about customers: Unsupervised domain-independent aspect detection for sentiment analysis of customer reviews[J]. Knowledge-Based Systems, 2013, 52: 201-213.

- [11]Vashishtha S, Susan S. Highlighting keyphrases using senti-scoring and fuzzy entropy for unsupervised sentiment analysis[J]. *Expert Systems with Applications*, 2021, 169: 114323.
- [12]Dai Y, Liu J, Zhang J, et al. Unsupervised sentiment analysis by transferring multi-source knowledge[J]. *Cognitive Computation*, 2021, 13: 1185-1197.
- [13]Nasukawa T, Yi J. Sentiment analysis: Capturing favorability using natural language processing[C]//*Proceedings of the 2nd international conference on Knowledge capture*. 2003: 70-77.
- [14]Zhuang L, Jing F, Zhu X Y. Movie review mining and summarization[C]//*Proceedings of the 15th ACM international conference on Information and knowledge management*. 2006: 43-50.
- [15]Filippova K, Strube M. Dependency tree based sentence compression[C]//*Proceedings of the Fifth International Natural Language Generation Conference*. 2008: 25-32.
- [16]Xu H, Hu C, Shen G. Discovery of dependency tree patterns for relation extraction[C]//*Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation*. Waseda University, 2009: 851-858.
- [17]O'Shea K, Nash R. An introduction to convolutional neural networks[J]. *arXiv preprint arXiv:1511.08458*, 2015.
- [18]Sherstinsky A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network[J]. *Physica D: Nonlinear Phenomena*, 2020, 404: 132306.
- [19]Tang D, Qin B, Feng X, et al. Effective LSTMs for target-dependent sentiment classification[J]. *arXiv preprint arXiv:1512.01100*, 2015.
- [20]Gu S, Zhang L, Hou Y, et al. A position-aware bidirectional attention network for aspect-level sentiment analysis[C]//*Proceedings of the 27th international conference on computational linguistics*. 2018: 774-784.
- [21]Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. *arXiv preprint arXiv:1609.02907*, 2016

- [22]Huang L, Sun X, Li S, et al. Syntax-aware graph attention network for aspect-level sentiment classification[C]//Proceedings of the 28th international conference on computational linguistics. 2020: 799-810.
- [23]Bai X, Liu P, Zhang Y. Investigating typed syntactic dependencies for targeted sentiment classification using graph attention neural network[J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2020, 29: 503-514.
- [24]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.
- [25]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.
- [26]Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]// Annual Conference on Neural Information Processing Systems. 2017: 5998-6008.
- [27]Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[J]. 2018.
- [28]Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.
- [29]Chen P, 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.
- [30]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.
- [31]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.

- [32] Marcus M, Santorini B, Marcinkiewicz M A. Building a large annotated corpus of English: The Penn Treebank[J]. Computational linguistics, 1993, 19(2): 313-330.
- [33] Koehn P, Federico M, Shen W, et al. Open source toolkit for statistical machine translation: Factored translation models and confusion network decoding[C]//Final Report of the 2006 JHU Summer Workshop. 2006.
- [34] Song X, Salcianu A, Song Y, et al. Fast wordpiece tokenization[J]. arXiv preprint arXiv:2012.15524, 2020.
- [35] Islam S M, Bhattacharya S. Ar-BERT: Aspect-relation enhanced aspect-level sentiment classification with multi-modal explanations[C]//Proceedings of the ACM Web Conference 2022. 2022: 987-998.
- [36] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.
- [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] 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
- [39] Kong J, Wang J, Zhang X. Hierarchical BERT with an adaptive fine-tuning strategy for document classification[J]. Knowledge-Based Systems, 2022, 238: 107872.
- [40] 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.
- [41] Kaneko M, Komachi M. Multi-head multi-layer attention to deep language representations for grammatical error detection[J]. Computación y Sistemas, 2019, 23(3): 883-891.

- [42]Hochreiter S, Schmidhuber J. Long short-term memory[J]. *Neural Computation*, 1997, 9(8): 1735-1780.
- [43]Chung J, Gulcehre C, Cho K H, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling[J]. *arXiv preprint arXiv:1412.3555*, 2014.
- [44]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.
- [45]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.
- [46]第 48 次《中国互联网络发展状况统计报告》发布 超十亿用户接入互联网 我国成全球最庞大数字社会[J]. *网络传播*, 2021(9): 76-81.
- [47]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.
- [48]Zhang C, Li Q, Song D. Aspect-based sentiment classification with aspect-specific graph convolutional networks[J]. *arXiv preprint arXiv:1909.03477*, 2019.
- [49]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.
- [50]Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. *arXiv preprint arXiv:1609.02907*, 2016.
- [51]Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate[J]. *Computer Science*, 2014,48(5):334-340.
- [52]Robinson J J. Dependency structures and transformational rules[J]. *Language*, 1970:259-285.

- [53]Huang B, Ou Y, Carley K M. Aspect level sentiment classification with attention-over-attention neural networks[C]//Social, Cultural, and Behavioral Modeling: 11th International Conference, SBP-BRiMS 2018, Washington, DC, USA, July 10-13, 2018, Proceedings 11. Springer International Publishing, 2018: 197-206.
- [54]巫浩盛, 缪裕青, 张万桢, 等. 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).
- [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]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.
- [57]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.
- [58]Kim Y. Convolutional neural networks for sentence classification[J]. arXiv preprint arXiv:1408.5882, 2014.
- [59]Pang B, Lee L, Vaithyanathan S. Thumbs up? Sentiment classification using machine learning techniques[J]. arXiv preprint cs/0205070, 2002.
- [60]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.
- [61]Ma X, Zhu Q, Zhou Y, et al. Improving question generation with sentence-level semantic matching and answer position inferring[C]//Proceedings of the AAI conference on artificial intelligence. 2020, 34(05): 8464-8471.
- [62]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.

- [63] 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.
- [64] Vinodhini G, Chandrasekaran R M. Sentiment analysis and opinion mining: a survey[J]. International Journal, 2012, 2(6): 282-292.
- [65] 王婷, 杨文忠. 文本情感分析方法研究综述[J]. 计算机工程与应用, 2021, 57(12): 14.
- [66] Wang, Y., Huang, M., Zhu, X., & Zhao, L. (2016, November). Attention-based LSTM for aspect-level sentiment classification. In Proceedings of the 2016 conference on empirical methods in natural language processing (pp. 606-615).
- [67] Ma D, Li S, Zhang X, et al. Interactive attention networks for aspect-level sentiment classification[J]. arXiv preprint arXiv:1709.00893, 2017.
- [68] Zhang, C., Li, Q., & Song, D. (2019). Aspect-based sentiment classification with aspect-specific graph convolutional networks. arXiv preprint arXiv:1909.03477.
- [69] Chen, P., Sun, Z., Bing, L., & Yang, W. (2017, September). Recurrent attention network on memory for aspect sentiment analysis. In Proceedings of the 2017 conference on empirical methods in natural language processing (pp. 452-461).
- [70] 陈珂, 梁斌, 柯文德, 许波, 曾国超. 基于多通道卷积神经网络的中文微博情感分析[J]. 计算机研究与发展, 2018, 55(05): 945-957.
- [71] Song, Youwei, et al. "Attentional encoder network for targeted sentiment classification." arXiv preprint arXiv:1902.09314(2019).
- [72] 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.
- [73] Tang D, Qin B, Liu T. Aspect level sentiment classification with deep memory network[J]. arXiv preprint arXiv:1605.08900, 2016.
- [74] Li R, Chen H, Feng F, et al. Dual graph convolutional networks for aspect-based sentiment analysis[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint

Conference on Natural Language Processing (Volume 1: Long Papers). 2021: 6319-6329.

- [75]Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

## 致谢

鲜衣怒马少年时，不负韶华且行知。行文至此，思绪万千，研究生三年时光转瞬即逝，可以写下去的梦很长。我与兰财的故事始于 21 金秋，终于 24 盛夏。回首三年光阴，满眼繁华，目光所及，皆是回忆，纵有万般不舍，皆是感恩，今借此文聊表谢忱。

青春有几年，疫情占三年。感谢国家、兰州财经大学和信工学院给我提供了一个和平、安全、舒适的学习环境和平等接受教育的机会，让我能够安心完成学业。

研究生三年我最想感谢的人就是我的导师李强教授。研究生期间给我提供了最好的学习环境和实验设备。从我的论文的选题，开题，写作，修改每一步都给了我悉心的指导和宝贵的意见，愚钝有时，老师从不责备总是回以关心。在平常生活中老师也总是给我很多人生建议，使我受益匪浅。感谢您这三年以来的指导包容和培养。在此，我想对我的导师李强老师致以最崇高的敬意，您永远是我人生路上的榜样。同时也非常感谢信息工程学院每位敬业负责的老师，正是他们所传授的知识让我不断地进步和成长。

感谢我的父母，感谢这么多年来你们无怨无悔的付出，永不停歇的照顾。感谢你们在我迷茫时做我最坚实的靠山，感谢你们在我每一次面临选择时帮我分析利弊却从来不替我做决定。养育之恩，无以回报！希望我的成长速度能赶得上你们变老的速度，我最大的心愿就是你们可以平安健康。

感谢与我一起度过研究生三年的同学和朋友们。感谢我美丽的师姐王文瑞、王雪绒，我帅气的师兄黄建民和学长刘伟，我优秀的同门赵金雨，我可爱的师妹张昭云、张艺华，我的师弟何博文、张永富。还有我 504 的室友们。愿我们此去前程似锦，再见依旧如故，未来可期。

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，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]李聪聪,李强,王雪绒等.基于双向门控机制和层次注意力的方面级情感分析[J].  
井冈山大学学报(自然科学版),2023,44(02):71-78.