摘要

近年来，随着大数据技术的迅猛发展，通过挖掘和分析用户在互联网平台上产生的大量文本数据，利用这些数据进行情感分析可以为政府以及企业提供合理且科学的决策方案，因此文本数据的情感分析引起了学术界的广泛关注。作为一项细粒度的情感分析任务，方面级情感分析可以得到文本中不同方面所对应的情感极性，包括积极、消极和中性。然而准确挖掘不同方面的情感极性在实际应用中情况较复杂，现有研究在分析方面级情感极性时忽略了方面词的左右局部语义信息，影响数据集中情感极性识别的准确性。为此，本文提出了一种多语义学习模型 BEDSA-MSL，该模型以 BERT 预训练语言模型、依存句法分析、注意力机制为基础，探究针对多语义的情感分析方法，以提升模型的情感极性识别率。本文的研究内容主要为以下两点：

（1）提出了一种适应方面级情感分析的轻量级中文预训练语言模型 smallBERT。该模型主要从缩减参数和优化 MASK 方案两个方面进行改进。在缩减参数方面，主要是在 BERT 基准模型基础上降低词嵌入维度和减少 Transformer 层数，缩短了模型的预训练时间。为了优化 MASK 方案，在处理中文维基百科语料时，针对语料中可能出现的部分英文单词，本文使用全词掩码方案遮盖中文中出现的全部英文单词，使模型更好地适应中文情感分析任务。同时在制作词典时增加了数字和特殊字符的替代符号，以降低无重要意义的文本内容对情感分析任务的干扰。

（2）提出了基于 BERT 和依存句法分析的多语义学习模型 BEDSA-MSL。该模型设计基于轻量级中文预训练语言模型 smallBERT，将输入的文本经词嵌入矩阵映射为实值向量，经局部语义学习层以方面为中心提取左右局部语义向量，通过方面感知层提取与方面关联度较高的注意力权重以生成方面情感向量，经语义信息融合层将左右两侧文本的方面情感向量进行拼接，拼接的方面情感向量经情感分类层得到方面情感极性。

对于 smallBERT，文中与 BERT 基准模型进行对比实验，最后使用预训练时间、AI challenger 语料情感分析 Accuracy、500 个样例预测时间对模型进行评

价，三个评价指标值分别为 8 天、85.27%、6.19s，相比于 BERT 基准模型，实验结果均得到优化，验证了本文轻量级中文预训练语言模型的有效性。

对于 BEDSA-MSL，本文在公开数据集上进行实验，同时与现有的方面级情感分析模型进行对比实验，最终的 Accuracy 值在 Restaurant、Laptop 和 AI challenger 2018 数据集上达到了 85.34%、81.24%和 85.63%，MF1 值在三个数据集上分别达到了 77.43%、76.81%和 77.85%，皆优于对比模型，实验结果表明本文设计的多语义学习模型具有实用性和可行性。

关键词：方面级情感分析；预训练模型；依存句法分析；多语义学习

Abstract

In recent years, with the rapid development of big data technology, by mining and analyzing the large amount of text data generated by users on the Internet platform, the use of these data for sentiment analysis can provide the government as well as enterprises with reasonable and scientific decision-making solutions, so the sentiment analysis of text data has attracted extensive attention from the academic community. As a fine-grained sentiment analysis task, aspect-level sentiment analysis can obtain the sentiment polarity corresponding to different aspects in the text, including positive, negative and neutral. However, accurately mining the sentiment polarity of different aspects is more complicated in practical applications. Existing studies ignore the left and right local semantic information of aspect words when analyzing aspect-level sentiment polarity, which affects the accuracy of sentiment polarity identification in the dataset. For this reason, this paper proposes a multi-semantic learning model BEDSA-MSL, which is based on BERT pre-trained language model, dependent syntactic analysis, and attention mechanism, and explores the sentiment analysis method for multi-semantics in order to improve the model's sentiment polarity recognition rate. The research content of this paper is mainly the following two points:

(1)A lightweight Chinese pre-training language model smallBERT for aspectual sentiment analysis is proposed, which is mainly improved in terms of parameter reduction and optimisation of the MASK scheme. In terms of parameter reduction, it mainly reduces the word embedding dimension and the number of Transformer layers based on the BERT benchmark model, which shortens the pre-training time of the model. In order to optimise the MASK scheme, when dealing with some English words that may appear in the Chinese Wikipedia corpus, this paper uses a full-word masking scheme to cover up all the English words appearing in Chinese, so that the model can be better adapted to the Chinese sentiment analysis task. Meanwhile, alternative symbols for numbers and special characters are added when making dictionaries to reduce the interference of unimportant text content on the sentiment analysis task.

(2)A multi-semantic learning model BEDSA-MSL based on BERT and dependent syntactic analysis is proposed, which is designed based on a lightweight Chinese pre-training language model, the input text is mapped into real-valued

vectors by word embedding matrix, the left and right local semantic vectors are extracted from the aspect-centered local semantic learning layer, and the attention weights that are highly correlated with the aspects are extracted from the aspect-aware layer to generate aspect sentiment vectors, and the aspect polarity is obtained by stitching the aspect sentiment vectors from the left and right sides of the text by the semantic information fusion layer. The aspect sentiment vectors of the left and right texts are spliced by the semantic information fusion layer, and the spliced aspect sentiment vectors are used to obtain the aspect sentiment polarity by the sentiment classification layer.

For smallBERT, the paper conducts comparison experiments with the BERT benchmark model, and finally evaluates the model using the pre-training time, AI challenger corpus sentiment analysis Accuracy, and 500 samples prediction time, and the values of the three evaluation metrics are respectively 8 days, 85.27%, and 6.19s, the experimental results have been optimized compared to the BERT benchmark model, verifying the effectiveness of the lightweight Chinese pre-training language model in this paper.

For BEDSA-MSL, this paper conducts experiments on public datasets, and also conducts comparison experiments with existing aspect-level sentiment analysis models, and the final Accuracy values reach 85.34%, 81.24%, and 85.63% on the Restaurant, Laptop, and AI challenger 2018 datasets, and the MF1 values reach 77.43%, 76.81% and 77.85% respectively, all of which are better than the comparison model, and the experimental results show that the multi-semantic learning model designed in this paper is practical and feasible.

Keywords: Aspect-level sentiment analysis; Pretraining model; Dependent syntactic analysis; Multi-semantic learning

目录

第一章 绪论

第一节　研究背景及意义

语言文字作为传递信息的载体，成为日常沟通交流中不可或缺的一部分，文字交流在增强团体中个体间互动性的同时提高了个体间相互合作能力。近年来，移动互联网和智能设备的发展日新月异，电子商务平台诸如亚马逊和淘宝随之展现出蓬勃发展的新气象。如此以往，这些平台上的用户评论数据呈现逐渐增长的趋势，其中大部分以语言文字和图像视频的形式存在。在互联网中，相比于图像视频，文本数据由于其便于编辑和所占存储空间小的优点，被广泛应用。提取这些数据中的关键信息并为实际应用提供支持是自然语言处理的主要任务。情感分析作为一种能够对文本数据提取情感信息的技术，通过分析文本数据中的情感信息，获取用户对某一对象的情感极性，对制定营销策略、建立舆论监督体系等具有重要的研究价值，成为当今自然语言处理领域最重要的任务之一[1]。

情感分析按照不同的研究对象可以分为：文档级、句子级和方面级[2]。文档级情感分析和句子级情感分析皆属于粗粒度情感分析任务，一般情况下，对于一个文档或句子来说，只包含一种情感倾向。传统的情感分析主要关注个人对某人或某事是否持有积极、中立或消极的态度。当前，新型社交媒体成为用户表达观点、交流思想的大数据平台，情感分析处理的数据规模逐渐增大。在实际应用中，细粒度情感分析可以更精准地捕捉用户情感并为产品改进提供有力支持。而方面级情感分析（ABSA）[3]的独特优势在于：可以挖掘文本中的特定信息，分析不同方面的情感极性，因此被广泛应用于市场营销、舆情监测、社交媒体监控等领域。

在细粒度情感分析任务中，方面一般指实体属性[4]，例如服装的属性有质量和价格，民宿的属性有地理位置和服务态度等。当前很多应用场景需要对评论数据进行深度挖掘，即分析文本中不同方面的情感极性。如表 1.1 中可以看出，对于评论文本：“这家照相馆的工作人员的拍照技术很好，可惜照相馆的位置有点儿偏僻”。传统的粗粒度情感分析只能分析出评论文本的整体情感极性，方面级情感分析可以根据对象的不同，得出不同方面的情感极性，具体情况如表 1.1 所示。

表 1.1 方面级情感分析任务中情感元素的介绍

名称	内容	
文本	这家照相馆的工作人员的拍照技术很好，可惜照相馆的位置有点儿偏僻。	
目标词	工作人员	照相馆
方面词	拍照技术	位置
观点词	很好	有点儿偏僻
基于方面词的情感信息	正面	负面

第二节　国内外研究现状

一、基于情感词典的方面级情感分析

情感词典是情感分析的一个重要工具，其中蕴含着丰富的情感词汇和情感极性，其中情感极性包括：积极、消极和中性等[5,6]。基于情感词典的方面级情感分析是指查找文本中出现的词或短语在情感词典中对应的情感极性，以此来推断文本中各个方面所表达的情感，同时将其归纳为正面情感、负面情感或中性情感。这种方法在社交媒体、电子商务、新闻报道等领域有着广泛的应用。基于情感词典的方面级情感分析的核心资源为情感词典，所以该方法非常依赖情感词典的构建，情感词典的质量直接影响方面词的情感极性分析结果。同时，该方法在使用过程中仍存在一些问题，比如情感词典中难以覆盖特定领域的情感词汇、方面识别困难即对于如何准确描述方面词情感信息的成分缺少完备的规则。此外，依赖于规则抽取信息的方式往往会出现高准确率、低召回率的现象[7]，现在已很少单独使用该方法。为此，张伟通过将情感词典和词性标注二者结合的方式，对信息抽取结果进行判断[8]。

二、基于机器学习的方面级情感分析

机器学习是基于数据驱动的一种学习方法[9]，它通过从大量标注好的数据中学习，以此来模仿人类智慧。评论数据的增长驱动文本分析技术的进步，使用情感词典分析文本内容已不适合大多数场景。此时，机器学习方法在挖掘文本时更胜一筹。传统的机器学习方法包括支持向量机[10]、决策树[11]和朴素贝叶斯[12]等。在情感分析领域，机器学习可以从大量标注数据中训练情感分类器，使用这种分类器可以自动地分析出文本中的情感极性，而不需要手动构建情感词汇；同时，机器学习方法可以适应于新的领域，与基于情感词典的方面级情感分析相比，基于机器学习的方面级情感分析模型泛化能力较强。然而，机器

学习方法同样存在不足之处，该方法需要特征工程（Feature Engineering）来处理文本，这种人工特征会耗费大量人力[13]。

三、基于深度学习的方面级情感分析

深度学习这一概念最早由 Hinton[14]等人于 2006 年提出，相比于情感词典和机器学习的方法，深度神经网络模型拥有深层次的网络结构，这类模型更加注重特征学习的重要性，通过将样本在原始空间的特征表示转换到一个新的特征空间，以便于情感分类和预测。与使用人工规则的方法构建特征相比，深度学习需要大量数据学习特征，由此可见训练深度神经网络的计算量十分巨大。

随着深度学习在学术界的逐步兴起，自然语言处理领域的学者开始使用深度神经网络模型进行分析研究。其中，以卷积神经网络[15]、循环神经网络[16]、门控循环单元[17]和长短时记忆网络[18]为代表的深度神经网络得到了广泛应用。而后，注意力机制[19]的出现使得自然语言处理领域的研究进入了新阶段。

在方面级情感分析中，一些学者提出将深度神经网络和注意力机制相结合的模型，新模型在分析文本内容时取得了不错的效果。Dong[20]等提出自适应递归神经网络（AdaRNN），使用该网络时，首先需要根据文本数据集生成依存树，目标词被放置在根节点。在计算目标节点的表示时，AdaRNN会根据上下文和句法关系，通过使用不同的语义组合，采用自底向上的方式进行。在获得目标词的表示后，使用分类器预测目标方面情感极性。Tang[21]等人设计了长短期（TD-LSTM）模型，该模型从两个方向将文本输入到LSTM网络中，在生成句子表示的同时捕获到目标单词与上下文的联系。Ruder[22]等人在此基础上展开进一步研究，提出了基于方面情感分析的层次模型，该模型通过将前向LSTM和后向LSTM的最终状态与方面词嵌入连接在一起，对评论中句子的相互依赖性进行建模。Wang[23]等人认为句子的情感极性不仅与句子内容有关，而且与涉及的方面相关联，上述提出的LSTM模型不能够充分提取到特定方面的上下文信息，为此设计了一个基于注意力机制的长短期记忆网络（Attention-based LSTM with Aspect Embedding, ATAE-LSTM）用于方面级情感分析，该网络架构利用方面嵌入决定句子的注意力权重。结果表明，加入注意力机制能够很好地解决问题。

Ma[24]等人认为前述研究虽然意识到方面在情感分析中的重要性，但是没有

对方面进行单独建模，于是构建了交互式注意网络（IAN）模型。IAN模型通过获取隐藏状态的平均值监督注意力向量的生成，再使用注意力机制关注方面和上下文信息，以此完成交互操作。Chen[25]等人提出了面向方面情感分析的记忆循环注意网络模型，模型首先将输入层的数据生成记忆信息，同时在此过程中生成单词序列特征，然后，通过非线性组合不同特征，对输入层生成的记忆进行多重关注，从而获取到重要信息来预测情感极性，增强了模型处理更多复杂问题的能力。Fan[26]等人设计了一种包含注意力机制的卷积记忆网络来预测方面的情感极性。模型中的注意力机制能够对多词组成的复杂表达式进行建模，解决了传统记忆网络只能捕获词级信息而不能对多个词构成的复杂表达式进行建模的问题。同时受卷积运算的启发，通过控制卷积窗口的大小将上下文以适当的顺序存储到不同的内存槽中。Hazarika[27]等人使用LSTM对句子中方面间的依赖关系进行建模，然后对所有方面进行情感极性分析。

Huang[28]等人提出了AOA（Attention-over-Attention）神经网络，该网络使用两个Bi-LSTM来学习方面词和句子的隐含语义，其中Bi-LSTM由两个LSTM网络堆叠而成，借助双向LSTM计算出成对交互矩阵，即计算方面到上下文的注意力和上下文到方面的注意力，从而使模型能够有效学习到上下文和方面词中的重要部分，结果表明AOA神经网络性能比其他以LSTM为基础的模型表现更优越。Majumder[29]等人发现一个句子中的某一方面会受到邻近方面的影响，由此构建一个新的模型IARM用于方面级情感分析，IARM使用循环记忆网络与多跳注意机制，将相邻方面相关信息融入到目标方面中，实验验证了该模型具有良好的性能。Xue[30]等人在前人的研究基础上提出了一种新模型，模型将卷积神经网络和门控机制相结合，用于方面类别情感分析和方面术语情感分析，该模型结构简单，门控单元独立工作，可以实现并行计算。根据特定方面，GTRU可以有选择性地提取情感特征，同时，模型中的两个卷积层可以分别对方面内容和情感信息进行处理。He[31]等人提出了一种交互式多任务学习网络（IMN），实现方面和意见术语的共同提取。与以往方法不同，该网络引入了新的消息传递结构，以此充分地利用相关性。Yang[32]等人设计了交替共注意网络模型，模型对方面层面和上下文层面的注意进行交替建模，从而关注到方面的关键词，学习更有效的情感特征。Jiang[33]等人提出了一个新的大规模多方面多情感（MAMS）数据集用于方面情感分类，同时构建了一个简单有效的胶囊网

络，在新数据集上验证了该网络模型的有效性。Xu[34]等人构建了ReviewRC数据集用于评论阅读理解（RRC）任务，该任务的主要目的是将评论转换为回答用户问题的资源，与此同时在语言模型BERT上探索了一种联合训练后方法来增强领域知识和任务知识的学习能力，提出的后训练方法应用在方面提取和方面情感分类等基于评论的任务中，结果表明这种后训练方法是有效的。

Liang[35]等人探索出一种新的解决方案，构建了方面间的依赖关系，在此基础上，设计了一种交互式图卷积网络（InterGCN）模型，该模型在提取情感特征时充分应用了方面间的信息。为此，可以交互的学习特定方面的情感特征，捕获到重要的上下文和方面词，在处理多词方面时，可以关注到关键方面词。Phan[36]等人认识到句法信息的重要性，将词性嵌入、基于依赖的嵌入和情境化嵌入相结合来增强方面提取器的性能，并提出了句法相对距离，以减轻不相关词的不利影响，提高了方面情感分类器的准确性。杜成玉[37]等人认为当方面和上下文较长时，以往基于方面和上下文的平均值学习注意力权重的方法可能会导致信息丢失，于是构建了一种螺旋注意力网络（BHAN），分别利用螺旋注意力层表示方面词和上下文，在有效获取信息量较大单词的同时，降低了信息交互的损失。Yan[38]等人提出了一种统一生成框架，统一的生成任务由所有ABSA子任务集合而成，这些子任务由预训练序列到序列模型求解而成。Zhang[39]等人提出了两种生成式范式，即注释式和抽取式，通过使用这两种范式来构造目标句子，用统一生成模型解决了多个情感对和三元组提取任务，模型在多个数据集上都取得了不错的效果，表明所提出的模型框架具有通用性和有效性。Liang[40]等人设计了一个基于SenticNet的图卷积网络（Sentic GCN）模型，模型在利用语境词与方面词之间依赖关系的同时考虑了方面词与意见词的情感信息，这种方法可以增强上下文的依赖关系，提升模型的性能。

第三节　论文研究内容

本文以 BERT 预训练语言模型、依存句法分析以及注意力机制为基础，将轻量级中文预训练语言模型应用到多语义学习模型中，对高效的多语义学习模型进行研究，文章研究的主要内容如下：

（1）当前预训练模型主要处理英文文本情感分析任务、很少处理中文文

本情感分析任务；此外，预训练模型参数量大、对算力要求高、预训练时间长，有必要设计一种轻量级中文预训练语言模型。预训练时使用的中文维基百科语料中包含的部分英文单词、数字和特殊字符可能会引入噪声，文中对 MASK 方案进行优化，对中文维基百科语料中出现的单词进行全词掩盖，同时在制作字典时增加数字和特殊字符的替代符号，使模型更好地适应中文文本情感分析任务。此外，文中通过降低模型词嵌入维度和减少 Transformer 层数以减少模型预训练时间。

（2）针对现有方面级情感分析研究忽略了方面词左右局部语义问题，提出了基于轻量级中文预训练任务的多语义学习模型。首先经嵌入层对输入文本进行编码，然后经局部语义学习层提取方面词左右局部语义。方面感知层获取与方面关联度较高的语义信息，为此，文中提出两种掩码机制$\mathrm{mask}_{\mathrm{A}}(\cdot)$和$\mathrm{mask}_{\mathrm{C}}(\cdot)$以获取 query、key 和 value 向量。query 和 key 向量经点积运算得到注意力权重，同时面向方面的依存树对 query 和 key 之间的点积注意力进行过滤，得到基于树的注意力权重，两种注意力权重经对比损失函数进行监督，从而获取与方面关联度高的注意力权重，注意力权重与 value 向量进行点积运算得到方面情感向量。语义信息融合层将方面情感向量进行拼接并降维，经情感分类层得到方面情感极性。最后通过实验验证了本文模型是有效的。

文中先提出了轻量级中文预训练语言模型 smallBERT，该模型以 BERT 模型为基准，从两个方面进行改进：优化 MASK 方案和缩减模型参数。随后，本文利用 smallBERT 模型、注意力机制、依存句法分析提出了多语义学习模型，即以方面目标为中心，分别学习方面的左侧局部语义和右侧局部语义。在多语义学习模型的词嵌入层使用 smallBERT 进行词嵌入，将词转换为词向量。使用注意力机制获取注意力权重，结合依存句法分析得到与方面关联度较高的权重。最后将局部语义向量拼接，经情感分类层进行情感极性的分析。

第四节　论文结构安排

本文内容共分为五个章节，各章节内容安排如下：

第一章：绪论。本章节首先概括了情感分析的研究背景及相关意义，进而引入方面级情感分析的内涵。随后，详细介绍了国内外方面级情感分析领域的

研究现状，全面阐述了相关的研究成果和发展趋势。最后，本章节主要阐述了本文的研究内容和结构安排，为全文的展开提供了指导性框架。

第二章：相关理论与技术。这一章主要对文本情感分析的理论基础和相关技术进行介绍，主要包括文本预处理技术、神经网络模型、注意力机制、Transformer 模型以及预训练语言模型 BERT 的相关理论。

第三章：轻量级中文预训练语言模型。这一章在介绍 Google 公布的 BERT 基准模型的基础上，分析大语言模型在预训练阶段存在的弊端，进而设计出一种适合中文情感分析任务的轻量级预训练语言模型，通过对比实验证明本章设计的模型是有效的。

第四章：基于 BERT 和依存句法分析的多语义学习模型。当前针对方面级情感分析的研究大多忽略了方面词两侧局部语义对情感分析结果的影响，导致模型在情感分析方面的准确率降低。本章提出了基于 BERT 和依存句法分析的多语义学习模型，模型的局部语义学习层获取方面词两侧的局部语义，方面感知层利用面向方面的依存树对左右两侧局部语义的注意力权重进行监督，从而选取与方面词关联度高的语义向量，提高模型的情感分析准确率。最后，本文在公开数据集上进行实验，与各类基准模型相比，本文提出的模型在 Accuracy 和 MF1 上均有所提升。

第五章：总结与展望。这一章对本文研究工作进行了深入的分析与总结，不仅指出当前工作的不足之处，还对未来研究方向进行展望，以期为未来研究提供有益的参考与启示。

第二章 相关理论与技术

本章主要介绍与方面级情感分析任务有关的理论知识和基础模型。首先介绍文本预处理涉及到的关键技术。其次，文中探讨了常用的神经网络结构，以便更好地理解其内在机制。随后，介绍了自然语言处理领域中的注意力机制。在此基础上，文中进一步介绍 Transformer 模型及相关理论，该模型同样在自然语言处理领域具有举足轻重的地位。最后，详细介绍了 BERT 预训练语言模型，该模型不仅在本领域中具有广泛的应用，更是本文研究的基础和出发点。

第一节　文本预处理技术

文本预处理[41]是自然语言处理任务的基石，当文本数据输入模型之前，需要经过一系列预处理操作，以符合模型的输入要求，比如：将文本数据转换为模型需要的向量形式以及向量大小。常见的预处理技术通常包括以下几个过程：数据清洗[42]、分词以及向量化[43]。

一、数据清洗与分词

一般情况下，原始的文本数据中往往掺杂着一些冗余信息或错误语法，如停用词、空格或者毫无意义的标点符号。为了提升数据质量，需要通过数据清洗和分词技术对这些文本数据进行预处理，以去除无用信息，确保文本数据的准确性和规范性。常用的数据清洗与分词方法如下：

（1）删除停用词。停用词在文本处理中经常容易被忽略，这些词一般不会蕴含太多实际意义，不仅会占用存储空间，浪费资源，而且会引入噪声，影响模型处理结果。

（2）删除标签和特殊符号。原始文本中如果出现 html 标签，则需直接删除。url 或者表情等特殊符号也可以根据实际情况删除。

（3）中文编码。编码是计算机处理文本数据的根基，计算机中的中英文可能使用不同的编码方式，所以需要将它们转换为统一的格式。

（4）分词。由于中英文本身的构词和语法规则不尽相同，所以中英文分词方法也有所区分。一般情况下，英文单词之间通过空格分隔，因此可以直接

使用空格分词。然而，中文语法结构与英文截然不同，无法通过空格或标点符号直接进行分词。在中文分词实践中，jieba 分词提供了三种分词模式：全模式、搜索引擎和精确模式，以满足不同场景下的分词需求，确保分词结果的准确性。

二、词向量

计算机处理文本数据的关键难题是如何将文本数据数值化，用词向量表示文本是解决这一难题的重要方案。根据发展历程，词向量表示方法主要有两种：独热编码和分布式表示。

独热编码的主要思想简单易懂，将文本中的每个词依据人工规则映射到高维且稀疏的向量空间，词表大小即向量维度。在进行映射之前，需要对词典中每个词进行编码。映射之后，向量空间中只有某一维度的值为 1（词在词典中的对应位置为 1），其余都为 0。这种方法虽然实现简单，但是它的缺点也是显而易见的。

（1）“词汇鸿沟”：两个单词的对应维度间彼此正交，相互独立，即使是语义非常相似的两个词，比如：“壮观”和“宏伟”，因为它们的词向量表示不尽相同，所以其词向量形式不能体现两个词之间的关系。

（2）“维度灾难”：当词表很大时，用独热编码表示的词向量空间维度会很高，这不仅会降低存储空间效率，在进行特征组合时，还有可能出现维度爆炸问题。

Hinton[44]提出用分布式词向量表示文本数据，这种词向量表示方法可以克服独热编码的缺点。分布式词向量的核心点在于，通过对语言模型进行学习，将各个词汇表示为固定而连续的稠密向量，同时将这些向量集合形成词向量空间。词向量空间的可视化效果为：空间中的每一个点代表不同的词汇。由此看来，分布式表示有以下优点。

（1）此方法可以方便地引入“距离”概念来计算词之间的相似性。

（2）此方法表示的词向量能够包含丰富的语义信息，且词向量的每一维度都有特定的含义。

2003年Bengio等[45]提出神经网络语言模型对词汇进行表示，该语言模型沿用n-gram思想，n-gram根据某一词的前n个词的分布来计算该词出现的概率，计算方法如下。

$$p(w_t|W_1^{t-1}) \approx p(w_t|W_{t-n+1}^{t-1}) \quad (2\text{-}1)$$

Bengio 等用w_t的前 n 个词来预测w_t出现的概率。具体做法为：输入层输入w_t的前 n 个词w_{t-n+1}，...，w_{t-2}，w_{t-1}的索引，通过共享权重矩阵 C 将索引映射成向量，然后将这些向量拼接成（n-1）*d 维的向量，其中 d 为词向量维度；拼接后的向量随后被送入隐藏层，该层利用 tanh 函数进行非线性映射，从而提取并转化向量的深层特征。随后，这些特征被传递到 softmax 层，该层负责计算出所有单词对应的归一化概率，为此可以提供每个单词在特定上下文中出现的概率。神经网络语言模型的三层框架如图 2.1 所示。

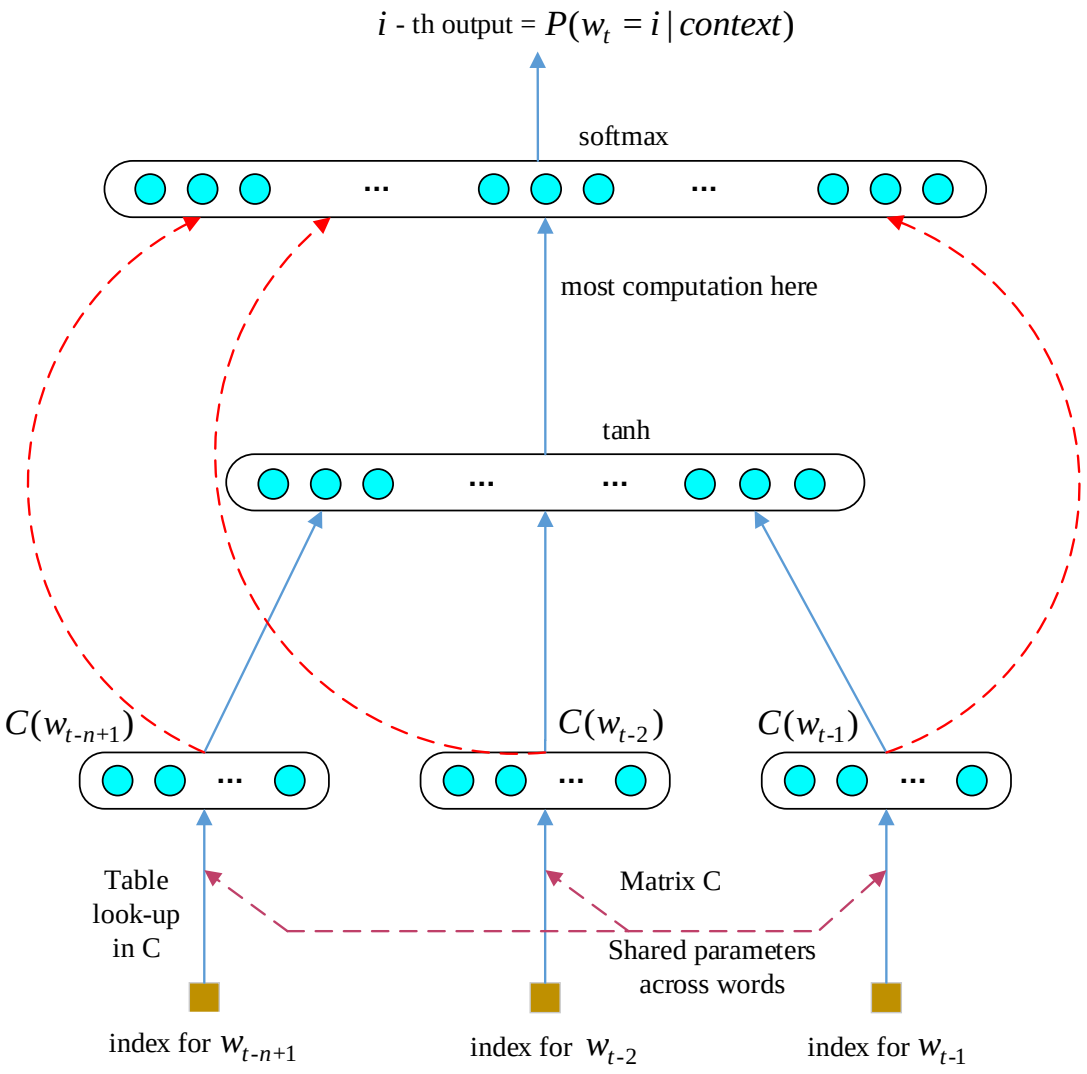


图 2.1 神经网络语言模型的三层框架

2013 年 Mikolov 等[46]在神经网络语言模型基础上提出 word2vec 框架，word2vec 包含两种训练模型：CBOW 和 Skip-gram 两种模型，即连续词袋模型和跳字模型，其具体架构如图 2.2 所示。

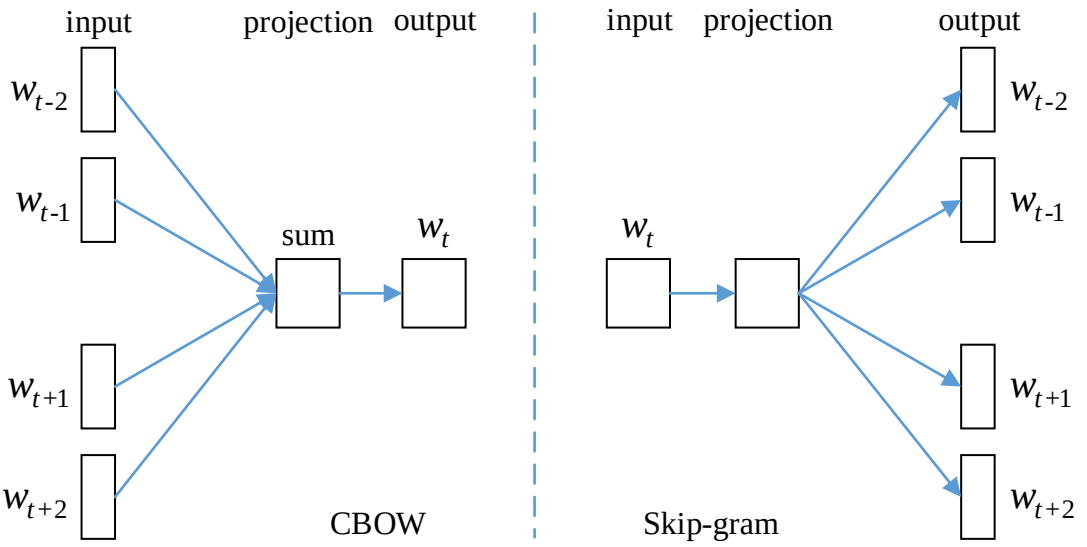


图 2.2 CBOW 和 Skip-gram 模型架构

CBOW 模型可以根据上下文预测中心词，模型共有三层结构，分别为输入层，投影层和输出层。

（1）输入层：将单词根据索引进行独热编码。

（2）投影层：将独热编码后的向量进行投影，使其变为规定维度的词向量，投影后的向量做求和累加。

（3）输出层：通过全连接层将求和后的向量映射到词表大小的向量空间中，同时使用 softmax 进行归一化，得到中心词的概率分布。CBOW 模型的优化目标函数为最大似然函数。

$$\pi_{\in C} p(w|Context(w)) \quad (2\text{-}2)$$

在实际应用时取对数，得到最终的目标函数。

$$\mathcal{L} = \sum_{w \in C} \log p(w|Context(w)) \quad (2\text{-}3)$$

其中，C代表语料库，w表示单词，Context(w)代表单词w依赖的上下文。

Skip-gram 模型的预测思路与 CBOW 模型正好相反，其主要思想为根据中心词来预测上下文，所以其优化目标函数如下所示。

$$\mathcal{L} = \sum_{w \in C} \log p(Context(w)|w) \quad (2\text{-}4)$$

第二节　神经网络模型

一、卷积神经网络

卷积神经网络（Convolution Neural Network, CNN）在前馈神经网络

（Feedforward Neural Network, FNN）的基础上进行了改进，两者既有相同点又有不同之处。与 FNN 相似，CNN 通过前向传播计算每层网络的输出值，同时通过反向传播调整权重和偏置；两者最大的区别在于 CNN 与相邻层的神经元是部分连接，而不像 FNN 那样与所有的神经元进行连接。起初，CNN 因其特殊的卷积运算被广泛应用于图像处理领域，Collobert[47]首次将用于图像处理领域的 CNN 应用于文本分类任务中，这一创造性设计取得了不错的成果。近年来，CNN 被广泛应用于自然语言处理领域。CNN 主要由卷积层、池化层与全连接层构成。

卷积层的主要作用是提取局部区域的特征，其中，不同的卷积核类似于不同的特征提取器。如果是对图像卷积，具体的操作为：局部区域矩阵和卷积核矩阵的对应位置元素相乘，再将各乘积值相加得到结果。例如，图 2.3 中的输入矩阵和2×2的卷积核进行卷积运算，运算过程为：左上角区域的元素和卷积核的相应位置元素相乘，求和后得到输出矩阵 S 的元素S_{00}。随后，将卷积核向右平移一个单位，进行相似运算，以此类推，得到输出矩阵元素S_{01}、S_{02}、S_{10}、S_{11}、S_{12}。

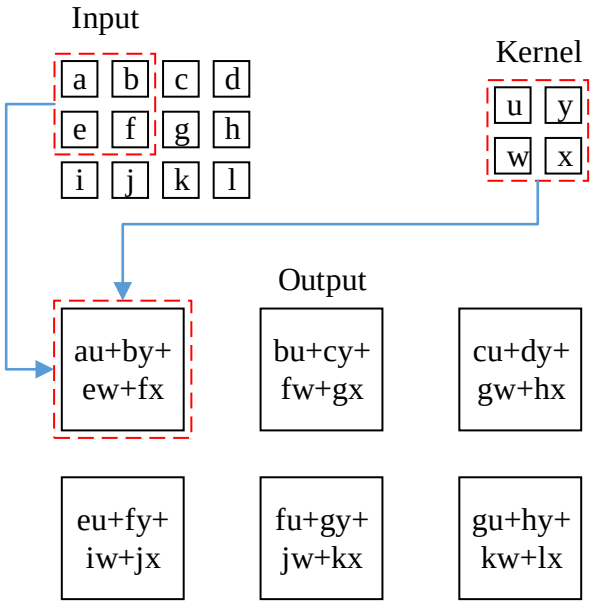


图 2.3 卷积运算

池化层的作用是进行特征选择，为了降低计算复杂度，通常会对特征矩阵进行压缩处理，以减少参数向量的维度。常用的池化方法由两种，平均池化和最大池化。两种池化方式的具体过程如图 2.4 所示。

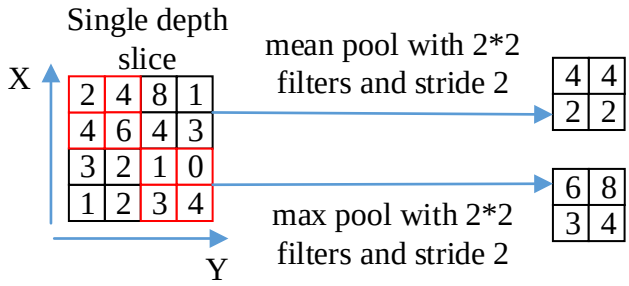


图 2.4 平均池化与最大池化过程

全连接层的主要作用是对卷积层得到的数据进行降维操作，将学到的特征映射到样本标记空间中，以便用于分类。

二、循环神经网络

循环神经网络（Recurrent Neural Network, RNN）是一种具有短期记忆能力的神经网络，上述卷积神经网络虽能够提取图像中的局部特征，但是很难处理像语音、文本这样的时序数据。在自然语言处理领域中，需要处理上下文相关联的时序语句，例如上下文句子“今天天气很好”和“适合组织团建”，使用卷积核为3×3的卷积神经网络处理语句时，每次只能看到 3 个文字，其处理序列数据的能力受到感受野大小的限制。循环神经网络（RNN）使用带自反馈的神经元，可以记录每个时刻的状态，所以当前网络的输出不仅与当前输入有关，还会考虑之前的输入信息。经典的 RNN 结构如图 2.5 所示。

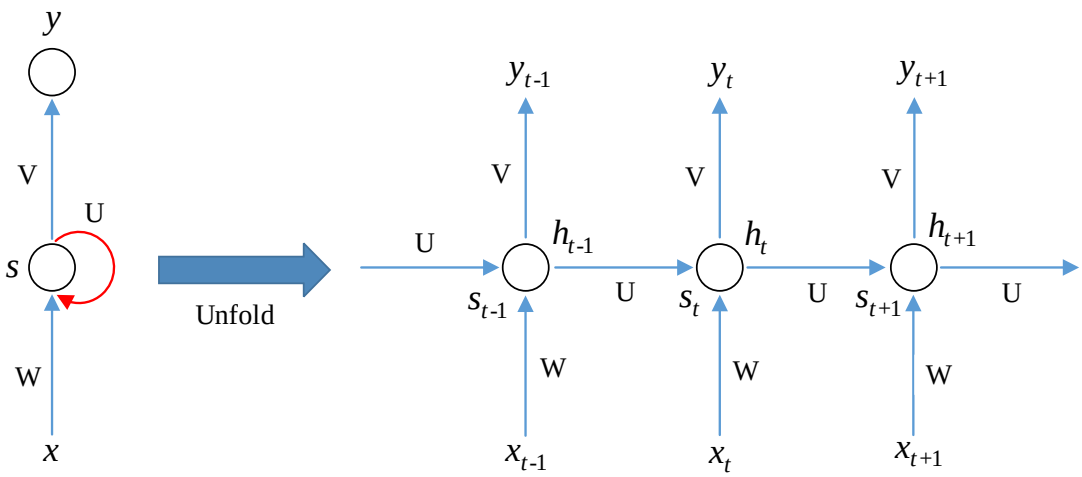


图 2.5 RNN 网络结构

图 2.5 左侧是具备标准结构的循环神经网络单元结构示意图，右侧则是由该单元按照时间顺序展开形成的网络结构示意图。

典型的循环神经网络单元分别由输入单元、隐藏单元以及输出单元构成，模型的输入由序列$\{x_0, x_1, \ldots, x_t, x_{t+1}, \ldots\}$表示，输出单元的输出内容由序列

$\{y_0, y_1, \ldots, y_t, y_{t+1}, \ldots\}$ 表示。此外，RNN 的隐藏单元状态标记为 $\{h_0, h_1, \ldots h_t, h_{t+1}, \ldots\}$。从上图可知，某一时刻隐藏层的输入由两部分组成，即当前时刻的输入和上一时刻隐藏层的输出，表明循环神经网络单元具有“记忆”功能。

由图中结构可以看出，RNN 能够很好地处理输入的每一个文本序列，因其具有短期记忆能力，所以 RNN 难以处理长文本序列。在进行误差反向传播（Back-Propagation, BP）更新参数时，每次的输出不仅与当前的输入状态有关，而且与前面若干时刻的状态相关，用链式法则计算乘积和求导时易出现梯度爆炸和梯度消失问题。

三、长短时记忆网络

长短时记忆网络（Long Short-Term Memory, LSTM）在 RNN 基础上进行改进，主要处理 RNN 在处理长文本序列时出现的梯度消失和梯度爆炸问题。

LSTM 的网络结构与 RNN 结构大体相同，因为二者的网路建构方法相同。从图 2.5 可以清晰地看出，RNN 的网络结构相对简单，由此改进得到的 LSTM 模型的神经单元却相对复杂。LSTM 在 RNN 的基础上引入门控机制来控制信息的选择与传递，研究者通过设计三个门控结构单元实现门控机制，提高网络的记忆能力。它们分别是：输入门、遗忘门、输出门，门控结构单元的设计，从一定程度上缓解了梯度爆炸和梯度消失问题，也充分弥补了循环神经网络的不足之处，使得 LSTM 在实际应用时能够处理长文本序列，三个门控结构分别对应图 2.6 中的三个部分。

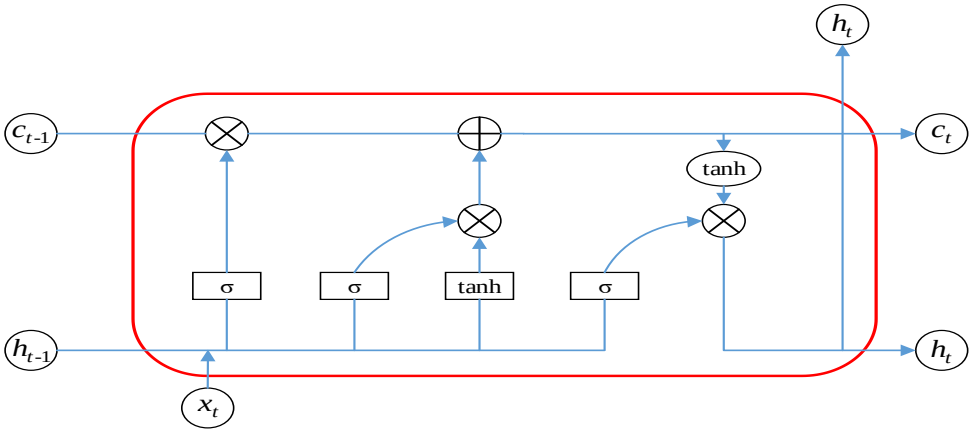


图 2.6 LSTM 网络单元结构示意图

图 2.6 所示为 LSTM 的标准网络结构，其中 σ 代表 sigmoid 函数，⊕代表向

量元素相加，⊗代表向量元素相乘，tanh 为双曲正切函数。

在 t 时刻时，序列输入为x_t，隐藏层状态h_t，上一时刻隐藏层状态h_{t-1}，记忆单元c_t，上一时刻记忆单元c_{t-1}，三个门分别是输入门i_t，遗忘门f_t，输出门o_t。其中，每个门结构都有对应的 sigmoid 函数，取值范围为（0,1），用不同的取值代表信息的更新和遗忘。

遗忘门的主要职责是确定在上一时刻存储的信息中，应保留或丢弃多少信息。通过遗忘门的作用，模型能够有选择地保留重要的历史信息，过滤不再需要的信息。经过激活函数计算出（0,1）之间的值。0 代表遗忘信息，1 代表保留信息，计算过程如式 2-5 所示。

$$f_t = (W_f x_t + U_f h_{t-1} + b_f) \tag{2-5}$$

其中W_f和U_f为参数矩阵，b_f为偏置参数。

随后对记忆单元信息更新，重要信息经输入门的处理之后保存到记忆单元中。输入门的 sigmoid 函数计算输入信息得出（0,1）之间的值i_t，i_t决定当前候选状态向量$\tilde{c}_t$中哪些信息需要保存，计算公式如下所示。

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{2-6}$$

$$\tilde{c}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{2-7}$$

其中，W_i、U_i、W_c和U_c为参数矩阵，b_i和b_c是偏置参数。

在遗忘门的选择作用和输入门的更新作用下，记忆单元的信息得以更新，具体更新过程如式（2-8）所示。

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{2-8}$$

经过记忆单元的更新，输出门将会决定最终的输出信息h_t，该输出以c_t为基础。输出操作分两部分进行，类似于遗忘门和输入门，用输出门的 sigmoid 函数计算当前信息x_t和上一时刻隐藏层状态h_{t-1}，计算结果为o_t。由计算结果决定哪部分记忆单元信息需要输出，更新之后的记忆单元信息c_t经 tanh 函数处理结果值在（-1,1）之间，然后再点乘o_t来决定最终的输出h_t。计算过程如下。

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{2-9}$$

$$h_t = o_t \odot \tanh(c_t) \tag{2-10}$$

其中W_o和U_o为参数矩阵，b_o为偏置参数。

LSTM 在 RNN 基础上引入门控机制，门控机制可以对信息进行筛选。其中判定为重要的信息被存储到记忆单元中，并且记忆单元只通过线性交互就可以

在不同时刻传递信息，这样促使 LSTM 具有处理长文本序列的能力。此外，LSTM 网络通过链式相加连接不同时刻的状态，使得导数计算由相乘转变为相加的方式，在一定度上规避了梯度爆炸和梯度消失问题的产生。

第三节　注意力机制

注意力机制的主要思想来源于认知神经学中的注意力，人类大脑总是可以有意或者无意从大量信息中选择一部分重要信息进行处理，同时忽略其他不重要信息。比如，我们去超市购买蔬菜时，我们的注意力在蔬菜区而不会关注其他区域。起初，注意力机制主要被用于机器视觉领域，Xu[48]等在神经网络中引入注意力机制，实验结果有明显提升。Bahdanau[49]以及 Shen[50]等人使用注意力机制处理机器翻译任务，取得了很好的效果。下面使用机器翻译的例子来介绍深度学习中的注意力机制。

在实际应用时，常采用序列到序列模型来处理机器翻译任务，使用的典型框架为编码-解码（encoder-decoder）结构。图 2.7 中的下面部分代表编码器端，上面部分代表解码器端。其中，encoder 端对“Where are you”利用 RNN 或 LSTM 等其他模型进行编码，解码得到输入语句的向量表示，将该向量表示传递给 decoder 端。在解码阶段，利用 RNN 或 LSTM 等模型对输入的句子向量进行解码，生成对应的中文翻译语句“你在哪”。在这个机器翻译过程中，decoder 端是逐字翻译的，如果输入信息过多，可能会造成模型内部混乱，最终出现错误的翻译结果。

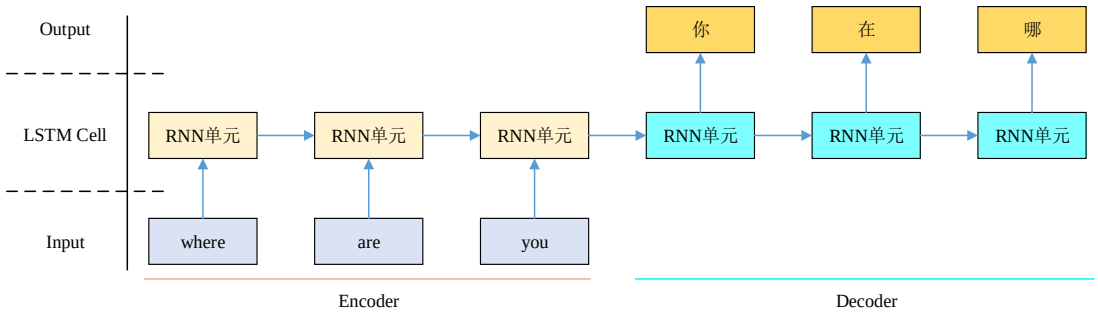


图 2.7 编码-解码（encoder-decoder）结构

注意力机制可以解决上述机器翻译过程中遇到的问题，从图 2.7 可以看出，在生成“你”的时候与单词“you”高度相关，与“Where are”的关系较弱。基于此，选择注意力机制处理机器翻译任务时，在翻译过程中会将注意力集中

到“you”上，而不是“Where you”，从而提高模型的性能。

下面将通过注意力机制在机器翻译任务中的实现方式，来了解注意力机制的具体原理。图 2.8 为经典的机器翻译结构 Seq2Seq，并且其中加入了注意力计算。图中具体展示了生成单词“deep”的计算过程，首先将前一时刻的输出状态q_2和 encoder 端的输出进行注意力计算，得到当前时刻的 context，计算公式如下。

$$[a_1, a_2, a_3, a_4] = \mathrm{softmax}([s(q_2, h_1), s(q_2, h_2), s(q_2, h_3), s(q_2, h_4)]) \quad (2\text{-}11)$$

注意力权重计算如式（2-12）所示。

$$a_i = \mathrm{softmax}(s(q, h_i)) = \frac{\exp(s(q,h_i))}{\sum_{j=1}^{n} \exp(s(q,h_i))} \quad (2\text{-}12)$$

$$\mathrm{context} = \sum_{i=1}^{4} a_i \ h_j \quad (2\text{-}13)$$

其中，$s(q_i, h_j)$为注意力打分函数，该函数结果值为标量，描述了q_i和每个输入h_j之间的相关性。然后使用 softmax 函数对注意力分数归一化，最后根据注意力分数对输入信息加权求和取得当前时刻的上下文向量 context，context 可以理解为截止当前 decoder 部分已有“I love”，有了注意力机制模型可以选择性的提取输入信息内容。

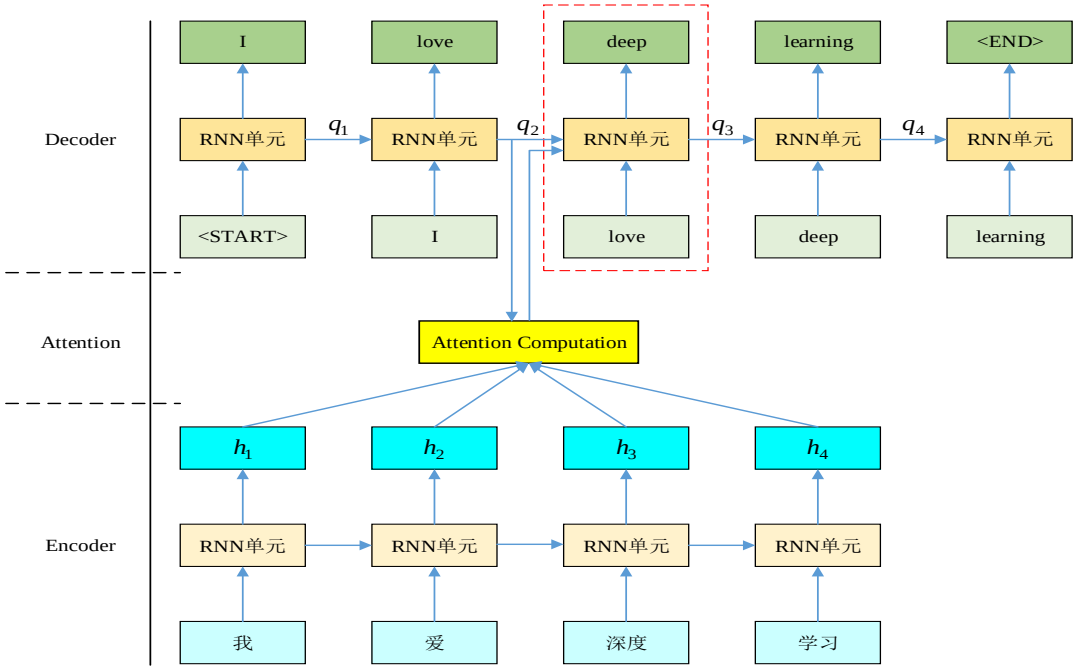


图 2.8 经典的机器翻译结构 Seq2Seq

由上述内容可知，注意力机制的实质是根据信息的重要程度来分配资源。在本文研究的方面级情感分析任务中，文中比较期望模型能够对给定的评论性

语句进行深度解析。具体来说，针对不同方面，文中期望模型能够精准地分析对应方面的情感极性，从而提供更为准确的情感分析结果。这就需要模型能够在同一语句中根据上下文给出不同方面的向量表示，这与注意力机制的思想非常吻合。因此，本文在后续方面级情感分析任务中引入了注意力机制。

第四节　Transformer 结构

即使模型在不断改进，但是以往的模型在实际应用时仍存在一些问题，RNN 等序列化模型只能按从左向右或者从右向左的次序处理数据。采用这种工作机制存在两个问题：其一是所需计算结果在时间 t 的情况下依赖于 t-1 时刻的计算结果，这将严重限制模型的并行能力；其二则是具有较强的计算复杂性。序列化模型在计算时会带来梯度爆炸和梯度消失问题，具有门控机制结构的 LSTM 虽然在一定程度缓解了长期依赖问题，但是对于特别长期的依赖情况，LSTM 依旧无能为力。而 Transformer[51]的提出解决了上述两个问题，突破序列化模型的限制，能够并行处理数据。下面将介绍 Transformer 的框架结构。

一、Transformer 总体结构

Transformer 同样遵循编码-解码结构。其中，编码器和解码器分别由 6 个编码块、6 个解码块组成。编码器的输入是词嵌入矩阵，解码器的输入由两部分组成，一部分来自多头注意力模块之前，通过词嵌入方法得到的向量和位置向量和作为输入，另一部分则在多头注意力前以 encoder 部分处理的结果作为输入。Transformer 总体模型结构如图 2.9 所示。

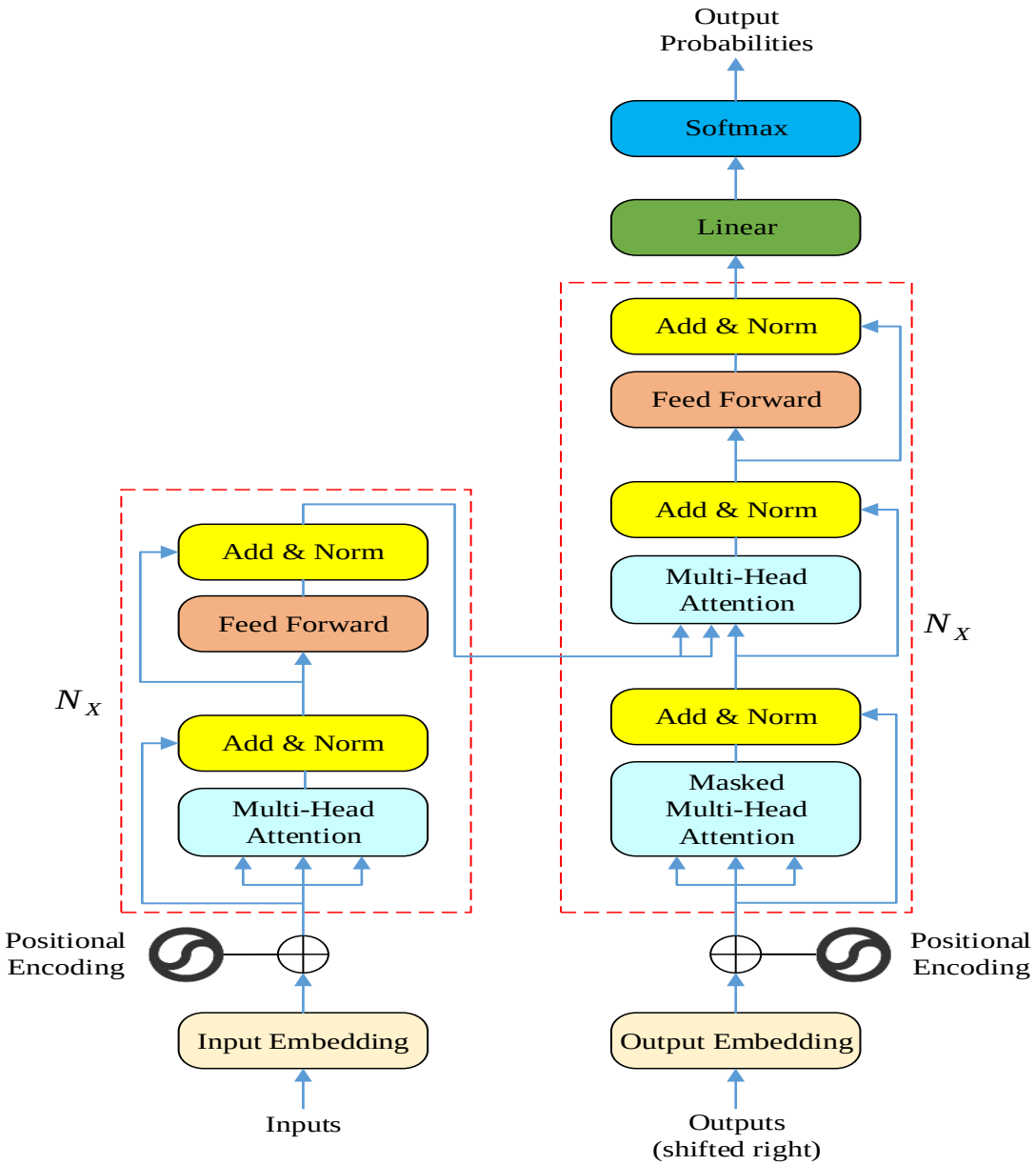


图 2.9 Transformer 模型

二、Transformer 的输入

Transformer 的输入部分由词嵌入方法得到的特征向量和位置编码得到的向量相加而得。位置嵌入（Position Embedding）是 Transformer 和 LSTM、RNN 之间的主要区别，LSTM、RNN 等结构中含有输入序列的顺序信息，而 Transformer 在编码时没有位序信息。为了让 Transformer 有能力理解输入词句中的单词顺序以及单词间的依赖关系，引入位置嵌入可以弥补 Transformer 不能利用序列顺序的不足。Transformer 中的位置嵌入有两种实现方式，一是单词的绝对位置嵌入，二是单词的相对位置嵌入，相对位置嵌入的计算公式如下。

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/d_{model}}}\right) \quad (2\text{-}14)$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/d_{model}}}\right) \quad (2\text{-}15)$$

其中，PE表示位置嵌入，pos表示单词在句子中的位置，i是指每个维度，2i表示偶数维度，$2i+1$表示奇数维度。

由上述内容可知，Transformer 的最终输入表示由单词的词嵌入和位置嵌入相加而得，输入过程详见图 2.10。

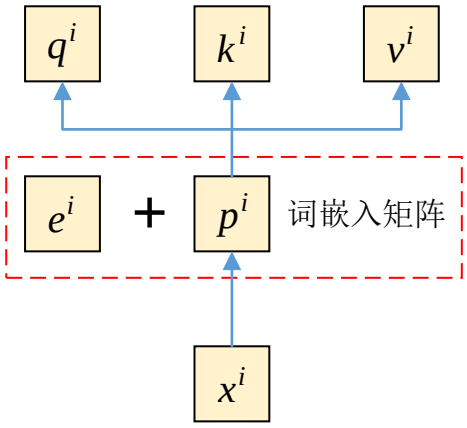
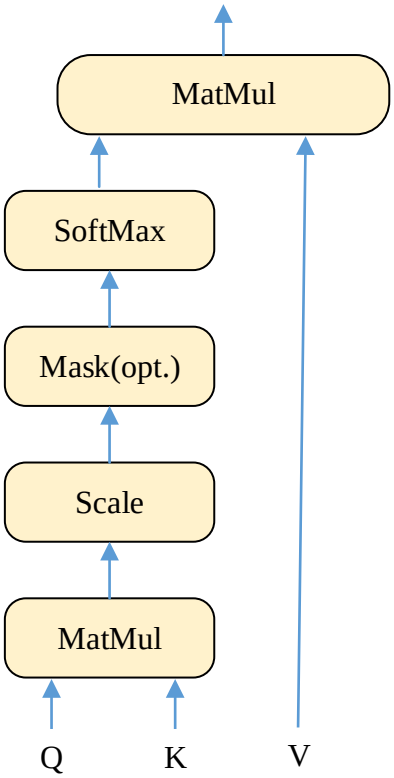


图 2.10 Transformer 输入过程

在输入语句中，每个单词均通过x^i表示，其向量化过程中对应的符号为e^i，单词在序列中的位置用p^i表示。

三、自注意力

Transformer 中的多头注意力模块由多个自注意力构成，自注意力、多头注意力的结构如图 2.11 所示。由图 2.11（a）可知自注意力的输入为三个部分，它们分别是 Q（Query）、K（Key）、V（Value）矩阵，Q、K、V 是每个输入在不同空间上映射的结果，三个矩阵经过缩放点积与 softmax 后得到自注意力权值。由图 2.11（b）可知，多头注意力机制实现并行计算的同时可以捕捉文本中不同的关注点和特征表示，使模型拥有更强大的表达能力和泛化性能。



2.11(a) 自注意力结构图

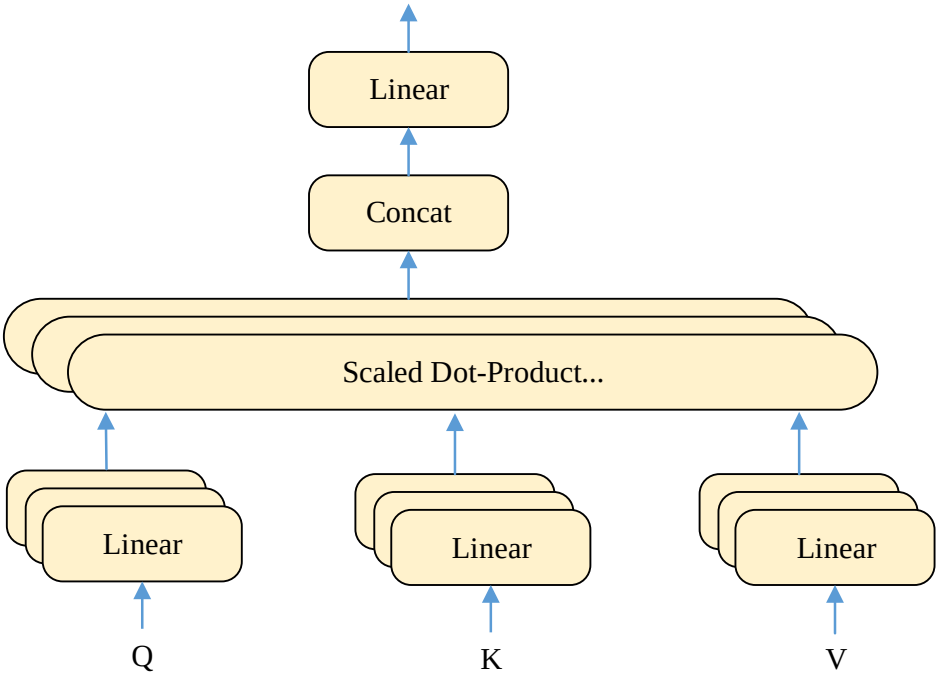


图 2.11（b） 多头注意力结构图

Self-Attention 的输入由三个部分组成，分别是 Q、K、V，其计算过程如式（2-16）所示。

$$\mathrm{Attention}(Q,K,V)=\mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{2-16}$$

其中，d_k代表 K 矩阵的向量维度，$\sqrt{d_k}$在公式（2-16）中为缩放因子。由于多头注意力机制是由多个子空间表示，相比于自注意力具有更强的表达能力。因此，在进行模型训练时，使用多头注意力可以有效避免模型出现过拟合的情况，多头注意力的计算公式如下所示。

$$\mathrm{MultiHead}(Q,K,V)=\mathrm{Concat}(\mathrm{head}_1,\mathrm{head}_2,\ldots,\mathrm{head}_h)W^O \quad (2\text{-}17)$$

$$\mathrm{head}_i=\mathrm{Attention}(QW_i^Q,KW_i^K,VW_i^v) \quad (2\text{-}18)$$

其中，h 代表头的数量，head_i代表第 i 个头的输出，W^O代表输出变换矩阵，多头注意力由 h 个单头拼接而成，形成最终的输出。

四、残差连接和层归一化

（1）残差连接：从 Transformer 总体结构可知，每个子层后面都有残差连接。加入残差连接之后，前一层的输入直接与当前层的输出相加，形成“跳跃”连接。这样可以使信息在网络中更容易传递和保持，避免出现梯度爆炸或梯度衰减的情况。残差连接如图 2.12 所示。

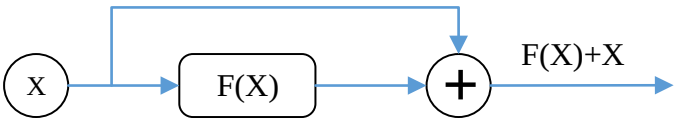


图 2.12 残差结构

当 X 为词嵌入时，F(X)代表自注意力时，此时残差连接计算[52]如下所示。

$$F(X)=X_{\mathrm{embedding}}+\mathrm{Attention}(Q,K,V) \quad (2\text{-}19)$$

在后续运算中，为获取残差连接，需要在经过每个模块时将前一层的输入与当前层的输出结果相加。

$$F(X)=X+\mathrm{SubLayer}(X) \quad (2\text{-}20)$$

（2）层归一化：层归一化是一种特征缩放技术，可以使神经网络的隐藏层化为标准正态分布，用于稳定深度神经网络的训练速度。层归一化用于在每个样本的所有特征上进行归一化，使得输出均值为 0，标准差为 1。层归一化计算公式如下。

$$\mu_i=\frac{1}{n}\sum_{i=1}^{n}x_{ij} \quad (2\text{-}21)$$

$$\sigma_i^2=\frac{1}{n}\sum_{i=1}^{n}(x_{ij}-\mu_i)^2 \quad (2\text{-}22)$$

$$\mathrm{LayerNorm}(x)=\alpha\otimes\frac{x_{ij}-u_i}{\sqrt{\sigma_i^2+\varepsilon}}+\beta \quad (2\text{-}23)$$

公式中的均值和方差是以矩阵的行为单位计算而得。

由式（2-23）可知，由每行的元素减去该行的平均值，然后将结果除以该行的标准差，以此方法进行层归一化处理。在该过程中，引入常数元素ε以防止除数为 0 的情况，α和β为可训练参数，用以弥补由层归一化引起的部分损失，⊗表示两个元素间的乘积。

第五节　预训练语言模型 BERT

BERT（Bidirectional Encoder Representation from Transformers）[53]是 2018 年 10 月由 Google 提出的一种典型预训练语言模型。其采用 Transformer 结构中的编码器部分，因 Transformer 结构可以堆叠，BERT 由多层 Transformer 结构堆叠而成，主要作用为学习词嵌入的表示。BERT 在多个自然语言处理任务中取得了优异的成绩，特别是为文本处理提供了功不可没的帮助。在 NLP 任务中，一般使用词嵌入作为模型的输入，然而传统的词嵌入方法存在两个显著的问题：一是无法捕捉文本的长距离依赖关系；二是无法充分地获取上下文信息以及恰当地处理多义词。为解决这些问题，学术研究者提出一些基于上下文的语言模型，比如 ELMo[54]、GPT[55]等。但这些模型仍存在一定的问题，ELMo 本质上只能处理单向上下文信息，正向和逆向的 LSTM 输出向量通常会忽略另一边的信息，因此处理后的信息是不完整的。GPT 虽是双向语言模型，但是只能通过自回归方式进行预训练。

预训练任务的提出源于现实生活中标注数据短缺而无标注数据丰富这一现象，针对特殊任务只能使用较少的标注数据训练模型，训练好的模型便不能够从中充分地学习到有意义的规律。预训练任务在巧妙利用大量标注数据资源的同时，可以获得满意的结果。预训练任务通常由上游任务和下游任务两个部分组成，上游任务的核心在于采用大量无标注数据来训练语言模型，得到具有共性特征的模型，下游任务将模型训练迁移到某一特定任务中，然后使用特定领域的标注数据进行微调，以学习特定任务的特性特征。这好比要培养学生英语写作能力，需要先认识单词，然后做大量英语阅读积累重点单词和优美语句，最后训练写作能力。BERT 的预训练模式可以同时捕捉长距离依赖关系和上下文信息，从而显著提升模型的性能。

第六节　本章小结

本章主要涵盖了与方面级情感分析相关的理论知识和基础模型，首先系统梳理了文本预处理在 ABSA 任务中的重要性，其中对于数据清洗、分词以及词向量的理论和方法进行了重要论述。常用的神经网络亦得到重用，文中分别对卷积神经网络、循环神经网络、长短时记忆网络的优缺点进行了归纳总结。此外，文中对注意力机制的原理以及计算过程进行较为详细的论述。最后，重点介绍了 Transformer 框架结构及相关理论，并阐述了其在文本处理中的应用优势。同时，也对 BERT 的结构和在文本处理中的优势进行了分析。

第三章 轻量级中文预训练模型

预训练任务因其可以利用已有模型处理新问题、计算效率高等优点在深度学习中备受瞩目。当前基准模型 BERT 在处理特定任务时存在参数量大、对算力要求高等问题。除此之外，掩码方案的选择会根据任务的不同而有所差异。在处理中文情感分析任务时，使用的语料库一般会包含部分英文单词，在分词时使用以字为单位的分词方法，则会将英文单词分成独立的字母从而失去单词本意，影响模型的分类性能。为了缩短模型的训练时间以及提高情感分析性能，本章旨在 BERT 模型基础上设计一种轻量级中文预训练语言模型 smallBERT，同时详细介绍本章模型的结构缩减方案以及 MASK 优化方案，最后通过对比实验验证本章模型的有效性。

第一节 BERT 模型分析

BERT 模型在训练之前，需要用一些特殊标记对输入数据进行修饰，包括令牌嵌入（Token Embeddings）：在每一个文本序列的头部加入[CLS]令牌，同时在每一句话的末尾加入[SEP]令牌。其中，最后一层[CLS]对应的向量代表整句语义信息，以此用作分类任务；段落表示（Segment Embeddings）：在每一句中都会添加表示第一句和第二句的特殊标记，因为在预训练过程中要以两个句子作为下一句预测任务的输入；位置嵌入（Position Embeddings）：每一句中的每个词都会添加位置嵌入，用于表示该词在句中的位置。如图 3.1 所示。

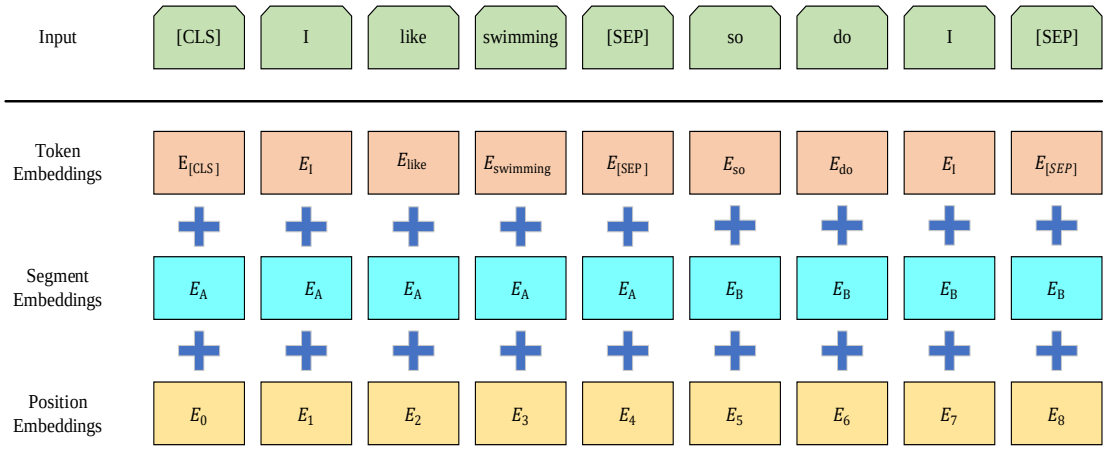


图 3.1 嵌入层输入表示

BERT 在预训练过程中引入了两种训练任务：掩码语言模型和下一句预测。

一、掩码语言模型（Masked Language Modeling, MLM）

若用上文中介绍的 Self-Attention 机制进行语言模型的预测任务，模型在每次预测时会看到后面的文本内容，因此会影响预测结果的真实性，使模型丧失学习能力。掩码语言模型的任务为：屏蔽输入文本中 15%的词，被屏蔽的词用[MASK]标记替换，在训练过程中根据未屏蔽的上下文信息预测被屏蔽的单词。然而此方法只会使模型在出现[MASK]标记的时候进行预测，对于其他情况则无从应对。因此，研究者针对被屏蔽的单词使用三种不同的屏蔽方式：其中 80%的词被[MASK]标记替换；10%的词被随机词替换；10%的词不做改变。

二、下一句预测（Next Sentence Prediction, NSP）

为了使模型有能力理解句子之间的关系，BERT 使用下一句预测任务进行训练，即预测两个句子是否连在一起。具体方法：对于每个训练样例，在语料库中挑选句子 A 和句子 B 来构成。其中，50%的情况下句子 B 为句子 A 的下一个句子，50%的情况下句子 B 是语料库中的随机语句。通过训练，BERT 模型有能力预测出第二个句子在原文中是否为第一个句子的下一句。

从图 3.1 可看出 BERT 模型的输入内容为两句话[CLS]I like swimming[SEP]so do I[SEP]，根据下一句预测任务特点，在预训练前需要准备相同格式但不属于上下文关系的句子对，比如[CLS]I like swimming [SEP]The flower is beautiful[SEP]，这种不属于上下文关系的句子对从掩码任务中无法学习到知识，因为模型无法判断哪些[MASK]标记是噪声。而使用下一句预测任务可以帮助模型分辨噪声和非噪声，所以 BERT 模型中的掩码语言模型任务和下一句预测任务会同时进行训练，最终完成 BERT 模型语义训练任务。

BERT 模型在训练时使用两种策略同时进行训练，所以模型的损失函数为两个任务的损失函数之和，当两个任务的损失函数值达到最小时，模型也得到了最理想的训练效果。预训练模型的整体结构如图 3.2 所示。

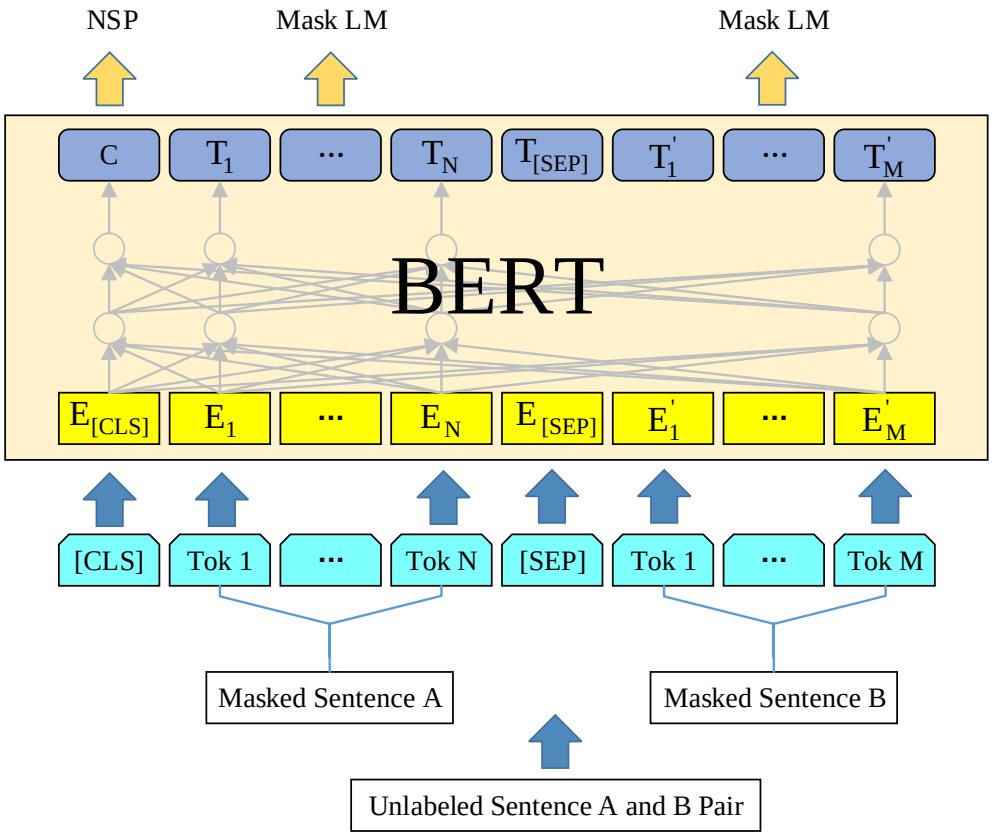


图 3.2 预训练模型整体结构

BERT 的内部双向结构与 Bi-LSTM 的双向网络结构有所区别，在 Bi-LSTM 网络结构中，每个单词都从词的正反两个方向分别得到一个词向量表示，之后将这两种表示进行拼接，这种拼接的词向量表示则被认为是单词的双向表示。BERT 采用双向 Transformer 结构进行训练，通过对输入文本进行整体感知，以生成融合上下文信息的深层双向语义信息。BERT 的双向结构如图 3.3 所示。

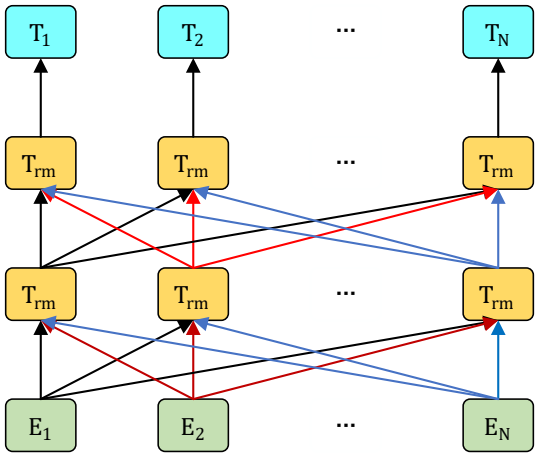


图 3.3 BERT 的双向结构

由图 3.3 可知，图中的 Transformer 为两层。其中，E_N指词向量表示，T_{rm}表示 Transformer，T_N指最终的隐藏层表示。从图中可看出上层的 Transformer

输入来自于下层的 Transformer 输出，语义向量经双向 Transformer 处理之后，句子中每个字中都含有该句中每个字的语义信息。

第二节　smallBERT 模型分析

一、smallBERT 模型结构

2018 年 Google 发布的论文中提出两种 BERT，其中一个是 BERT-BASE，另一个是 BERT-LARGE，两种 BERT 模型结构相似，主要参数不同，各主要参数具体数值如表 3.1 所示。

表 3.1 BERT 模型主要参数表

BERT 模型	Transformer 层数 L	隐藏层词嵌入维度 H	多头注意力的头数 A	参数量
BERT-BASE	12	768	12	110M
BERT-LARGE	24	1024	16	340M

BERT 模型自成功发布以来，相关学者在此基础上进行改进，研究出一系列预训练模型，这些模型大都具有很大的参数量。然而，巨大的参数量也存在弊端，模型对算力要求更高、预训练耗时更长，BERT-BASE 使用 16 块 TPU 需要运算 4 天，BERT-LARGE 使用 64 块 TPU 则需要运算 4 天。不仅如此，大模型在处理数据量较小的任务时模型表现效果反而会降低。BERT-BASE 预训练模型起初是针对机器翻译、问答等多种任务设计的，采用该模型进行下游任务时，可能会引入噪声，从而影响实验结果的准确性。

综合以上论述，本章设计了一种轻量级预训练模型 smallBERT，模型由 6 层 Transformer 构成，隐藏层词嵌入维度由 768 维降到 384 维，多头注意力的头数与 BERT-BASE 相同为 12 个，参数量为 27M。与 BERT-BASE 相比，smallBERT 能够在保证性能的同时有效缩短训练时间。smallBERT 预训练模型的结构如图 3.4 所示。

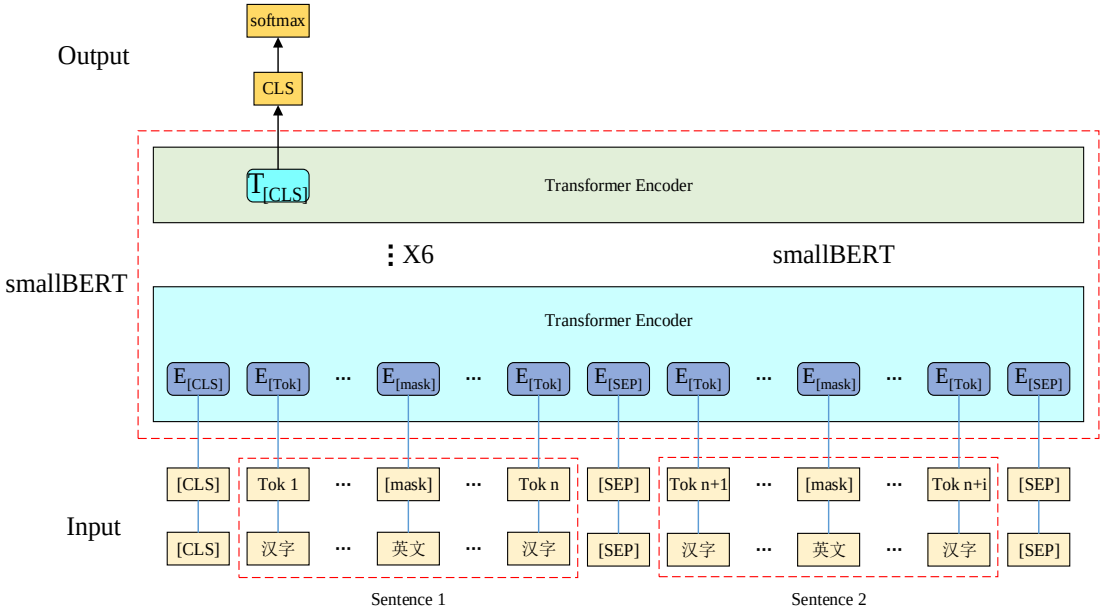


图 3.4 smallBERT 预训练模型结构

由上图可知，smallBERT 预训练模型结构共有三层，分别为输入层、smallBERT 层以及输出层。其中，输入层负责将数据集中的句子对转化为词向量形式，在转化之前，每句句首添加[CLS]分类符，句尾添加[SEP]结束符；词向量经 smallBERT 层计算后得到词与词之间的注意力分数，模型根据注意力分数的高低获取语料特征；输出层根据[CLS]分类符对应的表征向量进行情感极性分类，同时可借助[CLS]来判断语料句子对是否具有上下句关系，从而达到学习的效果。

二、MASK 方案

在自然语言处理任务中，一个常见的问题是输入的文本序列长度不一，所以通常需进行 padding 操作，使文本序列长度相等；在预训练语言模型中，通常需要根据上文预测下文，以增强模型的学习能力。因此，需要借助 MASK 方案处理输入语料，MASK 的作用为：处理不定长文本序列和防止未来信息泄露。

（1）处理不定长文本序列。在文本处理中通常使用 padding 处理不定长文本序列，对于未达到标准长度的语句需补 0 以使句子达到指定长度，方便模型运算，具体的补 0 方案如图 3.5 所示。