摘 要

文本蕴含识别旨在推断两段文本之间的语义关系：蕴含、矛盾、中立。在该任务中，如何能够使深度学习模型更好地理解文本的语义，对于文本之间的语义关系分类至关重要。目前，大多数文本蕴含识别方法都是通过互注意力的方法，判定句子之间的语义关系，这种方法只能捕捉句子之间的交互信息，弱化了句子本身的全局信息，且没有考虑到句子的句法结构信息；同时，这些模型在面对低频词时表现欠佳。基于上述问题，本文提出了以下解决方法。

(1)针对大多数深度学习模型只能捕捉句子交互信息，且未考虑句法结构信息这一问题，本文提出了融入句法结构和摘要信息的文本蕴含识别模型。通过结合自注意力和互注意力机制的方式，从句子的全局和局部交互信息考虑，并融入句子的句法结构信息，从而更准确地推测句子之间的语义关系；同时，收集和整理了公务员试题的部分选择题，利用摘要信息抽取的方法，解决公务员试题中题目冗长和答案简短导致的长度不对称问题，最后，将该模型和文本蕴含识别的思想应用于试题答题中。实验结果表明，该模型在公共数据集和公务员试题上的表现，超越了多个基准模型。

(2)针对深度学习模型面对低频词表现欠佳的问题，本文展开了基于文本增强的文本蕴含识别研究。该方法根据词频阈值划分不同的低频词序列，通过义原信息增强和同义词替换的方式，增强低频词语义信息，如果不存在义原或同义词，则进行字级别信息增强。通过实验对比发现，两种文本增强策略均可带来不同程度的性能提升，特别是在单独抽取出包含低频词的语句对时，性能提升更加显著。

关键词：文本蕴含识别；自注意力；句法结构信息；摘要信息抽取；文本增强

Abstract

Recognizing Textual Entailment aims to infer the semantic relationship between two pieces of text: Entailment, Contradiction, Neutral. In this task, it is crucial to enable deep learning models to better understand the semantic meaning of text for the classification of semantic relationships between texts. Currently, most Recognition Textual Entailment methods use the method of mutual attention to determine the semantic relationship between sentences, which can only capture the interaction information between sentences, weaken the global information of sentences, and do not consider the syntactic structure information of sentences. Moreover, these models perform poorly when dealing with low-frequency words. Based on the above problems, this article proposes the following solutions.

(1)To address the issue that most deep learning models can only capture the interaction information between sentences and do not consider syntactic structure information, this article proposes a Recognition Textual Entailment model that incorporates syntactic structure and summary information. By combining self-attention and mutual attention mechanisms, this model considers the global and local interaction information of sentences and integrates syntactic structure information to more accurately infer the semantic relationship between sentences. Additionally, a portion of the civil service exam multiple-choice questions was collected and organized, and a summary extraction method was used to solve the problem of length asymmetry caused by lengthy questions and brief answers. Finally, this model and the Recognition Textual Entailment idea were applied to exam question answering. Experimental results demonstrate that the performance of this model on both public datasets and civil service exam questions outperforms multiple benchmark models.

(2)To address the issue of poor performance of deep learning models when faced with low-frequency words, this article presents research on Recognition Textual Entailment based on text enhancement. This method divides different low-frequency word sequences based on a frequency threshold and enhances the semantic information of low-frequency words through Sememe information enhancement and synonym replacement. If Sememe or synonyms do not exist, character-level information enhancement is performed. Experimental comparisons reveal that both

text enhancement strategies can bring varying degrees of performance improvement, particularly when extracting sentence pairs containing low-frequency words.

Keywords: Recognition Textual Entailment; self-attention; syntactic structure information; summary information extraction; textual enhancement

目 录

1 引 言

1.1 研究背景与意义

文本蕴含识别（Recognizing Textual Entailment，RTE）是 2018 年全国科学技术名词审定委员会公布的计算机科学技术名词，它是判断一个文本所含信息是否可以推演出另一个文本所含信息的过程、技术和方法[1]。文本蕴含识别的任务是识别两个文本之间的语义关系，在这项任务中，其中一个文本作为前提（Premise），另一个文本作为假设（Hypothesis），任务的目的是判断这两个文本之间的语义关系：蕴含（entailment）、矛盾（contradiction）、中立（neutral），具体示例如表 1-1 所示：

表 1-1

	ID	Sentence	Label
Premise		A dog jumping for a Frisbee in the snow.	
Hypothesis	Example1	An animal is outside in the cold weather,playing with a plastic toy	entailment
	Example2	A cat washed his face and whiskers with his front paw.	contradiction
	Example3	A pet is enjoying a game of fetch with his owner	neutral

近些年来，随着深度学习的发展，越来越多的研究者开始将深度神经网络应用于文本蕴含关系识别任务中，并在一些数据集上取得了巨大的提升[2-4]。但是，目前大部分的文本蕴含识别研究都是在英文数据集上开展，并未在中文语料上开展太多研究，这主要是由于：中英文之间存在天然的语义鸿沟，在英文数据集上表现良好的深度学习模型，在中文语料库上的表现却差强人意；并且，文本蕴含识别领域的主流深度学习模型都是建立在互注意力的基础之上，虽然能够捕捉句子的交互信息，但是忽略了句子本身的全局信息和句法结构信息。此外，在自然语言处理中，对于词语语义的理解至关重要，因此，如何使模型能够更好地理解低频词的语义是一个亟待解决的问题。所以，针对上述问题提出一些解决方案，具有一定的研究意义。

文本蕴含识别被应用在问答系统、信息检索等多个领域。在问答系统中，它可以在一个问题的多个答案选项中，推断出合理的答案[5,6]，Chen 等人[7]将文本蕴含识别模型应用于问答任务中，发现文本蕴含识别的方法可以改善跨领域问答模型的置信度估计；而在信息检索领域，仅仅依靠关键词和信息库中的内容进行相似度匹配，具有一定的局限性，如果融入文本蕴含识别，从文本之间

的语义角度考虑并进行检索，则能够检索出更合理、有效的信息。因此，文本蕴含识别具有重要的实际应用价值。

1.2 国内外研究现状

1.2.1 国外研究现状

国外对文本蕴含识别的研究在本世纪初就已经开始，早在 2005 年，英国就曾举办第一界 RTE 挑战赛[8]，后续的每一届挑战赛中主办方都会给出相关的数据集，众多学者也纷纷给出了不同的方法和改进的算法。但是，上述比赛中给出的数据集普遍规模较小，深度网络难以在训练过程中充分学习文本的特征。为解决文本蕴含识别数据集规模较小，致使深度网络难以得到充分训练的问题，在 2015 年，Bowman 等人[9]发布了大规模的文本蕴含识别数据集 SNLI，SNLI 数据集一共包含 570k 个文本对，其中训练集 550k，验证集 10k，测试集 10k，一共包含 entailment、contradiction 和 neutra 三种语义关系标签。2018 年，Williams 等人[10]又推出多种类自然语言推理（MultiNLI）数据集，与 SNLI 数据集类似，这个数据集一共包含 433k 个文本对，并涵盖了多种口语和书面语体，支持独特的跨语体泛化评估。这些大规模英文语料库的发布，为基于神经网络的研究方法垫底了基础，但是，在大规模英文语料库未发布之前，也有一些学者使用不同的方法在小规模语料库上展开研究，接下来，本文将介绍不同阶段的文本蕴含识别方法。

1）基于相似度的方法

如果两个句子之间具有蕴含关系，那么这两个句子往往比较相似，因此有学者提出：通过前提句和假设句之间的相似度来判断文本之间的蕴含关系。

Jijkoun 等人[11]提出了基于词袋模型的文本蕴含识别方法，这种方法根据词频的大小对单词赋予不同的权重，然后计算 Lin 相似度和 WordNet 相似度，最终在 PASCAL-2005 的 RTE 数据集上达到 0.55 的准确率；Adams 等人[12]在 Jijkoun 的基础上利用 WordNet 抽出词链来链接前提句和假设句，并计算两个句向量的距离，然后使用决策树结合其他特征对文本蕴含关系进行分类，在 PASCAL-2005 的 RTE 数据集上达到了 0.657 的准确率。这种方法虽然简单，但是强行假设“相似即蕴含”是有明显缺陷的。

2）基于文本对齐的方法

在基于相似度的基础之上，产生了基于文本对齐的方法，这种方法首先把前提句 P 和假设句 H 中相似的部分找出来进行对齐，然后把对齐的方式和程度作为判断是否构成蕴含的依据。

Marneffe 等人[13]首先把前提句和假设句中的单词进行对齐,然后再加工特征，

使用逻辑回归进行蕴含分析,实现了通过自动对齐辅助识别蕴含关系的方法，不过这篇论文主要做的是冲突检测,和蕴含关系识别稍有区别;Iftene 等人[14]提出了非监督的文本蕴含识别方法,作者先使用外部知识库把前提句和假设句中的相似部分做映射，再计算局部对齐程度和全局对齐程度,最终通过全局对齐程度是否超过阈值判断两个句子之间的关系。

基于文本对齐的方法是对于基于相似度方法的一种改进，更关注于前提句和假设句中相同的部分，不过这类方法也有着自身的缺陷，即需要加入先验知识进行文本对齐，也不能很好地处理两个文本之间复杂对齐的情况。

3）基于数理逻辑的方法

文本蕴含识别实质上是一种语义推理关系，在数学上可视为一种数理逻辑问题，数学界在这一方面已经有了比较成熟的方法和工具，所以将数学中的数理逻辑在文本蕴含识别中，理论上是一种行之有效的方法。

Hobbs 等人[15]首次将溯因推理应用到文本蕴含识别当中，溯因推理是通过试图找到某个命题成立的原因的方式进行推理，这是将数学中的推理知识应用到知识蕴含识别的早期探索；Toledo 等人[16]把演绎法应用到推理当中，演绎法也被称为演绎推理，它一种由事物已知部分推理得到事物未知部分的方法，如“明天要下雨”作为前提句，可以推理得到“明天要带伞”这一结果。文中设定，如果由前提句能够推理得到假设句，那么两个文本之间构成蕴含关系，反之则不构成蕴含关系。

基于数理逻辑的方法是把数学界的知识迁移到知识蕴含识别领域的一次尝试，但是这种方法需要足够的背景知识来补齐推理链，而且只能适用于特定数据集，鲁棒性和容错性欠佳。

4）基于文本转换的方法

基于数理逻辑的方法虽然有着足够的数学理论作为支撑，但是有着其明显的缺陷，所以又有学者利用语言分析技术，如句法树、依存图等，通过这些方法将前提句和假设句进行改写，然后通过图间的相似度判别是否存在蕴含关系。

Bar-Haim 等人[17]把前提句和假设句都改写为句法树的形式，然后融入推理规则进行再次改写，如果前提句能够改写成为假设句的句法树形式，则认为两句话之间存在蕴含关系。MacCartney 等人[18]使用了 Natural Logic 的方法进行推理，这种方法需要把推理规则的序列作为特征，然后把特征放入决策树中进行判定。

文本转换的方法的核心还是数理逻辑的方法，但是前者会把文本转化为句法树、依存图等形式，然后通过图间的相似来判断蕴含关系，是一种基于图间相似度进行判断的方法，不过由于其核心还是逻辑演算，所以逻辑演算存在的问题，也同样存在于文本转换中。

5）基于混合模型的方法

文本转换的方法有着本身的不足之处，所以有学者提出了基于混合模型的方法，这种方法是把词级别的相似度、句法树相似度、对齐程度等混合在一起作为特征，然后使用分类器进行分类的方法。Zhang 等人[19]把前提句和假设句中的公共子序列合成最小信息树，再加以句子余弦相似度、反义词等特征进行分类。Tan 等人[20]利用两个句子之间重叠词的 IDF，计算两个文本之间的相似度，过滤推断的结果，然后推断蕴含的可能性。

6） 基于神经网络的方法

随着深度学习的发展，神经网络在自然语言处理领域中得到了广泛的应用，这类方法主要以卷积神经网络和循环神经网络为主。Yin 等人[21]将卷积神经网络和注意力机制结合起来处理文本蕴含识别问题，不过普通的卷积神经网络无法有效捕获句法信息，这就丧失了文本蕴含识别中的重要特征。Mou 等人[22]将句法依存树作为卷积的操作对象，提取子节点和父节点之间的依存关系，最终在 SNLI 数据集上达到了 0.824 的准确率。Wang 等人[23]重点关注两个句子之间的匹配和对齐情况，当两个句子无法对齐时，利用 NULL 进行两个句子之间的软对齐，然后使用 LSTM 对句子进行建模，同时把 Attention 向量进行拼接，输入到 LSTM 中，在 SNLI 数据集上的准确率达到了 0.861。Kim 等人[24]引入了 DenseNet 的思想，利用 LSTM 搭建 5 层 RNN 网络的同时，将上一层的参数拼接到下一层，然后使用 AutoEnconder 进行降维，在 SNLI 数据集上达到了 90.1 的准确率。Zhang 等人[25]在 bert[26]中融入语义角色标注信息，在 SNLI 数据集上达到了 91.9 的准确度。Laban 等人[27]为检测自动生成的文本摘要和人工编写的文本摘要之间的匹配度，将文本蕴含识别模型应用于匹配度检测任务中，相对于基准模型，准确度提升了 5%。Bauer 等人[28]将外部知识融入文本蕴含识别任务中，实验结果表明，在跨领域数据集中引入外部知识，能够显著提高模型性能。

1.2.2 国内研究现状

目前的文本蕴含识别数据集还是以英文为主，在早期，仅有 RTE-2 比赛和 RIET-3 比赛提供了少量的简体中文和繁体中文数据集，后来，CCL2018 发布了包含 11 万条数据的中文数据集 CNLI[29],但是发布的中文数据集只是将英文数据集翻译之后进行人工审查得到。刘焕勇老师利用在线翻译工具将 SNLI 数据集翻译为中文，构造了中文数据集[30]，共有 88w 对句子。2020 年，Hu 等人[31]构建了第一个非翻译的、使用原生汉语的大型中文文本蕴含数据集（OCNLI），OCNLI 数据集包含 5 万余训练数据，3 千验证数据及 3 千测试数据，数据来源于政府公报、新闻、文学、电视谈话节目等多个领域。

张鹏等人[32]利用深度优先搜索技术，查询 FrameNet 中前提句和假设句之间的上下位关系，然后利用 WordNet 中的语义关系比较两句话之间的框架元素是否相似，以此判断两个句子之间的蕴含关系。谭咏梅等人[33]将句子的字符特征、句法特征、语义特征等特征提取出来，使用贝叶斯逻辑回归模型进行蕴含识别得到初步结果，然后使用规则集合进行过滤，得到最终的蕴含结果，但是传统的机器学习方法需要人工筛选大量特征，所以又提出了基于神经网络的方法[34]，该方法使用 CNN 与 LSTM 分别对句子进行建模，自动提取相关特征，然后使用全连接层进行分类。于东等人[35]将文本蕴含识别的三分类扩展为七分类的蕴含类型识别和蕴含语块边界类型识别任务，在 ESIM[36]和 BERT[26]模型上分别达到了 69.19%和 62.09%的准确率。王伟等人[37]认为现有推理模型的训练时间较久，于是提出了轻量级的文本蕴含模型，在保持识别准确率的同时，推理速度比其他主流文本蕴含模型提升了一倍。

1.3 本文的主要研究内容和章节安排

1.3.1 本文的主要研究内容

文本蕴含识别任务在英文领域已经取得了一定的成就，但是在中文领域仍缺乏一定的研究，并且，目前的中文文本蕴含识别研究大多是建立在互注意力的基础之上，没有考虑句子的句法结构信息，同时，在面对低频词时，模型难以充分学习到低频词的语义。基于此，本文提出了基于摘要抽取和文本增强的中文文本蕴含识别研究，其主要研究如下：

①将自注意力机制和句法结构应用于中文文本蕴含识别。传统的文本蕴含识别方法都是利用互注意力机制捕获句子之间的交互信息，以交互信息作为语句对分类的标准，这种方法弱化了句子的全局信息，并且没有考虑句子的句法结构。所以，本文在互注意力的基础之上，利用自注意力机制获取句子的全局信息，并在对句子建模时，融入句子的句法结构信息。

②使用摘要抽取技术处理公务员试题，并将文本蕴含识别研究应用于试题答题中。本文通过爬虫技术爬取公务员试题，并使用摘要信息抽取的方法，解决公务员试题题目冗长、答案简短导致的句子长度不对称问题，最后将文本蕴含识别的思想应用于该部分数据中。

③提出基于文本增强的文本蕴含识别模型。当语句中出现低频词时，大部分模型难以充分学习到其语义信息，这有可能会导致模型学习到的句意不准确。因此，对于低频词，本文以融入义原信息和同义词替换为主，字级别信息增强为辅的方式，使模型学习到更多的语义信息，该方法能够在一定程度上解决模型难以充分学习低频词语义的困难。

1.3.2 本文的主要创新

本文提出的基于摘要抽取和文本增强的中文文本蕴含识别研究，在多个公共数据集上均取得了良好的实验效果，其具体创新点主要体现在以下几个方面：

（1）**模型创新**：捕获句子的全局信息，并融入句子的句法结构信息。通过自注意力机制获取句子的全局信息，并在句子建模阶段融入句法结构信息，综合考虑句子的多方面特征信息，从而使模型具有更优秀的语义分类能力。

（2）**应用点创新**：将该研究应用于公务员试题中。本文从公务员试题的选择题入手，将题目和答案分别作为前提句和假设句，并利用摘要信息抽取的方法，解决公务员试题中题目冗长和答案简短导致的长度不对称问题，最后，将文本蕴含识别方法应用于该部分数据进行实验。

（3）**文本增强策略创新**：利用文本增强方法增强低频词语义信息。以义原信息和同义词替换为主，字信息增强为辅的方法处理低频词，从数据层面使模型获取更准确的语义信息。

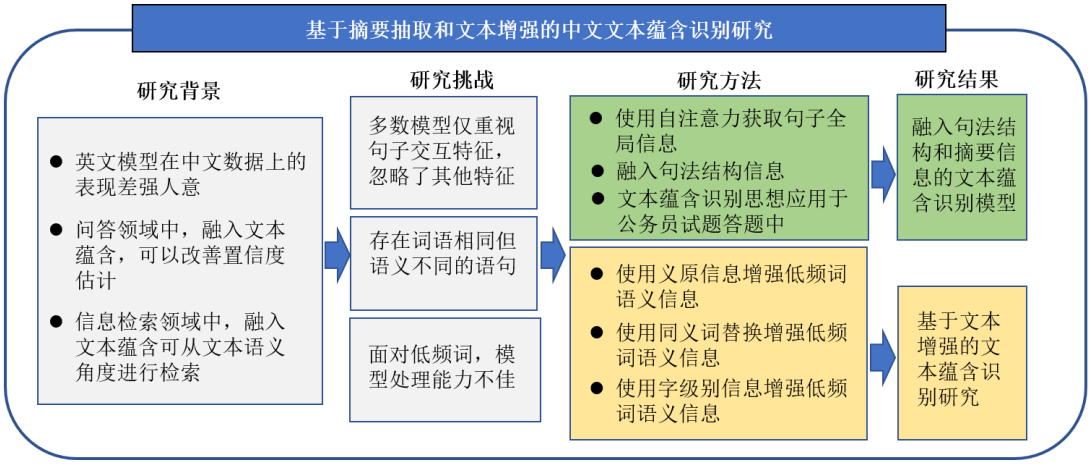


图 1-1 本文主要研究工作

1.3.3 本文的章节安排

本文章节安排如下：

第一章主要介绍文本蕴含识别任务的研究背景、意义和研究现状。第一节描述文本蕴含识别任务的形式以及该任务的研究意义和应用价值；第二节则是将本文任务的研究现状从国外和国内两个维度展开叙述，并对国外研究现状进行了系统分类；第三节介绍了本文的研究内容以及主要创新点，并对本文的章节安排进行了概括。

第二章详细描述了本文所使用到的相关理论及方法。第一节介绍了深度学

习模型，其中，着重描写了本文所使用的循环神经网络和注意力机制；第二节对自然语言处理领域的词嵌入模型进行了展开介绍，分别介绍了静态词嵌入和动态词嵌入模型；第三节对句法分析任务进行了概括；第四小节从抽取式和生成式两个维度介绍文本摘要任务；第五小节总结了文本增强的策略。

第三章具体描述了本文所提出的融入句法结构和摘要信息的文本蕴含识别模型。第一节介绍了公务员试题的收集与整理过程；第二节对该章节的思路进行了具体描述；第三节则是将模型展开，自下而上描述模型中每个模块的细节和作用；在第四节中，本文将该模型应用于不同数据中进行实验，并展开分析，以此论证方法的有效性，同时，开展了消融实验，以探究各模块在模型中的重要程度。

第四章详细介绍了基于文本增强的文本蕴含识别研究。第一节介绍了该章节所需的义原信息；第二节对该章节的思路进行了具体描述；第三节主要对低频词的处理、文本增强策略以及所使用的深度学习模型进行了介绍；第四节则是研究不同文本增强策略对实验结果的影响。

第五章对本文所进行的研究进行了总结，并提出了对未来的展望。

2 相关理论及方法

2.1 深度学习模型

深度学习是一个用人类的数学知识与计算机算法构建起整体架构，再结合尽可能多的训练数据，以及计算机的大规模运算能力去调整内部参数，尽可能逼近问题目标的半理论、半经验的建模方式。接下来将分别介绍本文使用的循环神经网络、注意力机制等深度学习模型。

2.1.1 循环神经网络

循环神经网络（recurrent neural network，简称 RNN）一类专门用于处理不定长序列数据的神经网络，源自于 1982 年由 Saratha Sathasivam 提出的霍普菲尔德网络，具体而言，循环神经网络，是指在全连接神经网络的基础上增加了前后时序上的关系。在自然语言处理领域中，通常需要对序列文本进行建模，文本中每个词语的语义信息又和上下文息息相关，而 RNN 由于具备捕捉序列前后时序关系的能力，因此，RNN 被广泛应用于自然语言处理领域的各个任务中。其具体结构如图 2-1 所示。

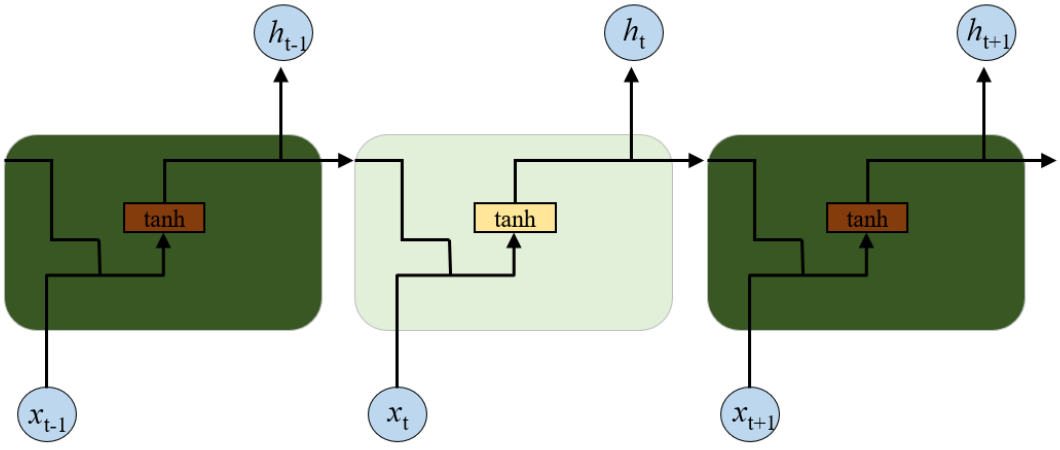


图 2-1 循环神经网络

在上图中，h_t为每个时间步 t 时刻下的隐藏层状态值，x_t表示 t 时刻的输入，tanh是激活函数。每个单元计算得到的隐藏层状态h_t，传递给下一个单元，以此获取序列的时序信息，具体计算公式如下所示：

$$h_t = \tanh\left(W_h \cdot [h_{t-1}, x_t] + b_h\right) \tag{2.1}$$

对于基本 RNN 来说，它能够处理一定的短期依赖，但是如果句子长度过长，则会“梯度爆炸”和“梯度消失”的问题。为了解决这种问题，LSTM 应运而生，LSTM 引入了细胞状态，并使用输入门、遗忘门和输出门三种门阀机制，来保持和控制信息的流通。

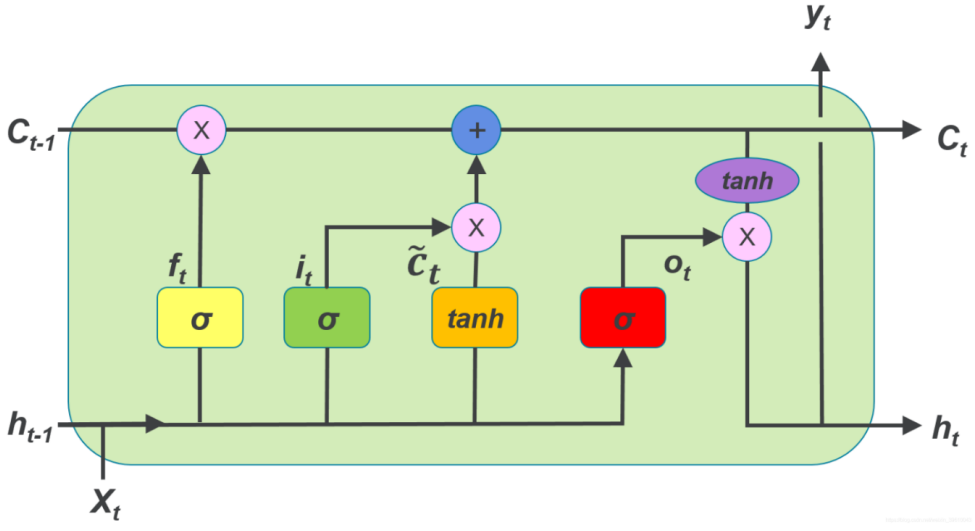


图 2-2 LSTM 单元结构

图 2-2 是 LSTM 的结构图，c_t表示细胞状态，即当前单元细胞状态的输出，和下一个时间步 t+1 的输入，σ为 sigmoid 激活函数，$\tilde{c}_t$为细胞状态候选值。LSTM 在时间步 t 下的计算公式如下所示：

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2.2}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2.3}$$

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{2.4}$$

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \tag{2.5}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{2.6}$$

$$h_t = o_t * \tanh(c_t) \tag{2.7}$$

LSTM 通过上述机制决定如何丢弃、保留和更新信息。由于最后的结果h_t是由多个函数组合作用得出，也不存在求和操作，因此在反向传播时不容易出现“梯度爆炸”和“梯度消失”的问题。

随着越来越多研究者的关注，LSTM 也衍生出许多的变种模型，比如门控循环单元网络（Gate Recurrent Unit，也称为 GRU），其主要变化是将遗忘门和输入门合成为一个更新门，然后将细胞状态信息和隐层信息合并为一个信息流，其参数更少，训练速度更快，也能降低过拟合的风险。双向长短时记忆网络（Bidirectional Long Short-term Memory，也称 Bi-LSTM）分为 2 个独立的 LSTM 网络，对于两个独立的 LSTM 网络，可以分别输入原序列和反序列作为输入序列，然后，将 LSTM 网络的两个输出向量（即进行特征提取后的特征向

量）进行拼接，将得到的拼接向量作为某个词语的最终特征表达。由于 Bi-LSTM 将反序列作为输入，故而，该网络能够捕获序列中从后到前的信息。

2.1.2 注意力机制

注意力机制源于人类视觉系统，在人类观察外界事物时，一般会选择性的观察外界事物的某些重要部分，对这一区域投入更多的注意力，降低对其他部分的关注度，进而快速筛选出高价值的信息。在自然语言处理领域，由于关注对象和计算方式的差异，注意力机制衍生出不同的变种机制，这些变种注意力机制可以赋予模型更高的辨识能力和解释能力。接下来将分别介绍互注意力机制、自注意力机制以及多头注意力机制。

- 互注意力机制

在自然语言处理领域中，互注意力机制的关注对象是两个不同的序列，根据序列中不同子部分的权重，重点关注权重高的部分，降低对权重较低部分的关注度。在文本处理相关任务中，互注意力机制最早应用于机器翻译任务中，Bahdanau 等人[38]利用互注意力机制将源端句子和目标端句子进行软对齐，得到源端句和目标端句子中词与词之间的依赖关系，以此提升翻译效果。

目前大多数的互注意力机制源于 sequence-to-sequence 模型（seq2seq），seq2seq 模型由编码器-解码器结构组成，如图 2-3 所示。在 seq2seq 模型中，编码器是一个 RNN 结构，其接受的输入序列为$X(x_1, x_2, \dots, x_n)$，其中，n 是输入序列的长度，输入序列被编码为固定长度的向量$H(h_1, h_2, \dots, h_n)$。解码端同样是 RNN 结构，输入为固定长度的向量h_n，通过 Token-to-Token 的方式生成一个输出序列$Y(y_1, y_2, \dots, y_n)$，h_t和s_t分别表示编码器和解码器的隐藏状态。

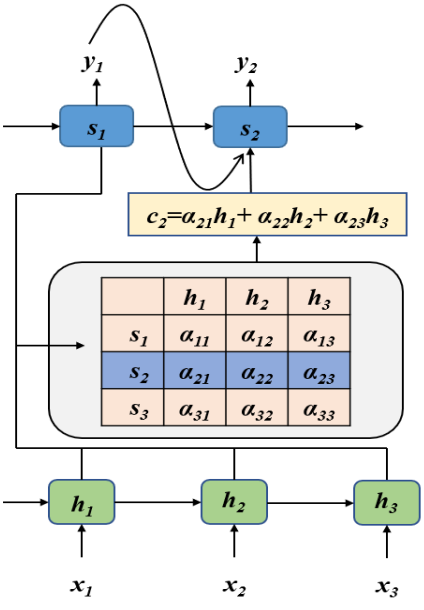


图 2-3 互注意力计算示例

图 2-3 的过程可由公式（2.8）-（2.10）表示：

$$c_j = \sum_{i=1}^{N} \alpha_{ij} h_i \tag{2.8}$$

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{N} exp(e_{ik})} \tag{2.9}$$

$$e_{ij} = f(h_i, h_j) \tag{2.10}$$

其中α_{ij}可以看做权重系数，表示h_j对c_i的重要程度，其数值范围在[0,1]之间，f为打分函数，具体实现可以是相似度计算函数或者前馈神经网络。

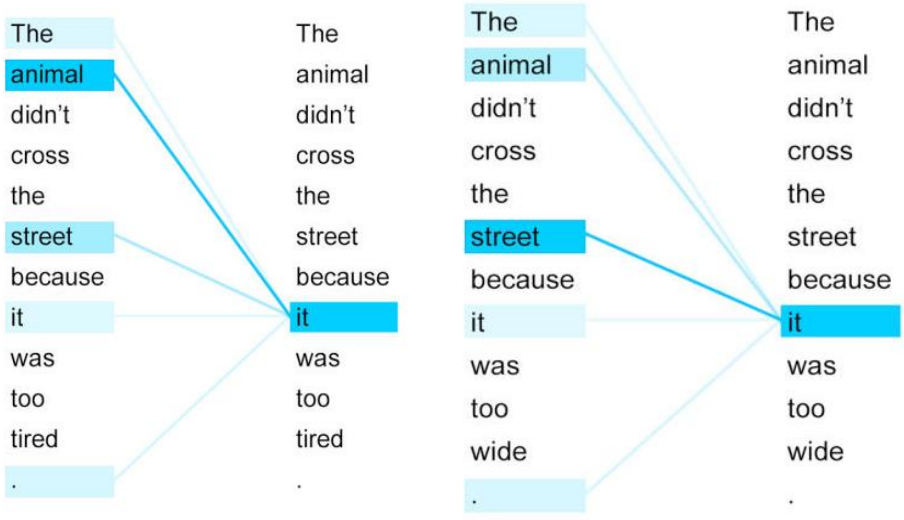
- 自注意力机制

自注意力机制（Self-Attention）是注意力机制的变体，其减少了对外部信息的依赖，更擅长捕捉数据或特征的内部相关性。在深度学习领域，我们期望网络在获取全局信息的同时，又能够将注意力聚焦在重点信息上，而自注意力机制则能够站在全局视野下，重点关注关键部分。在自然语言处理领域，句子中的一个词语往往不是独立的，它的语义和上下文信息息息相关，所以，在处理单个词语的同时，也要重点关注它的上下文信息，以及和它本身关联性较高的词语，自注意力机制由于本身的特性，能够很好地解决这一问题。2017 年，Lin 等人[39]首次提出自注意力机制，并在多个数据集上进行实验，均取得了不错的效果。Vaswani 等人[40]以自注意力机制为基础，提出了 transformer 模型，被广泛应用于图像和视频等方面的研究，Devlin 等人[26]又在 transformer 基础之上，提出了预训练模型——BERT。

自注意力机制的计算过程分为几个步骤：①根据嵌入向量得到 Query、Key、Value 三个向量；② 为每个向量计算得分：score=$Q \cdot K$；③ 对 score 进行归一化处理，即除以$\sqrt{d_k}$，d_k是 Q 和 K 的维度；④对 score 施以 softmax 函数，并点乘 Value 向量，得到最终输出结果z (这种通过 query 和 key 的相似性程度来确定 value 的权重分布的方法也被称为 scaled dot-product attention)。上述过程可由公式 2.11 表示：

$$z = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{2.11}$$

以上就是自注意力机制的计算过程，其中,Q=K=V。下边是两个句子中"it"与上下文单词的关系热点图，可以看出图 2-4(a)中的"it"与"animal"关系很强，图 2-4(b)中"it"与"street"关系很强，说明自注意力机制是可以很好地学习到上下文相关的语言信息。



（a）　　（b）

图 2-4 关系热点图

- 多头注意力机制

在自注意力机制中，神经网络模型在对输入序列进行编码时，会过度关注当前位置的信息，为了解决该问题，Vaswani 等人[40]在他们的研究中提出了一种称为多头注意力机制的方法。和使用单独的一个注意力池化不同，多头注意力通过 h 组不同的线性投影（linear projections）来变换 Query、Key 和 Value 向量，接着，将这些变换后的向量并行地进行注意力池化，然后，将池化后的 h 组向量拼接在一起，拼接向量经过线性变换后，产生最终的输出，这种设计被称为多头注意力，与使用单一注意力池化方法不同，多头注意力可以学习不同的变换，以更好地捕捉不同方面的信息。其中 h 个注意力池化输出中的每一个输出都被称作一个头。具体结构图如下所示：

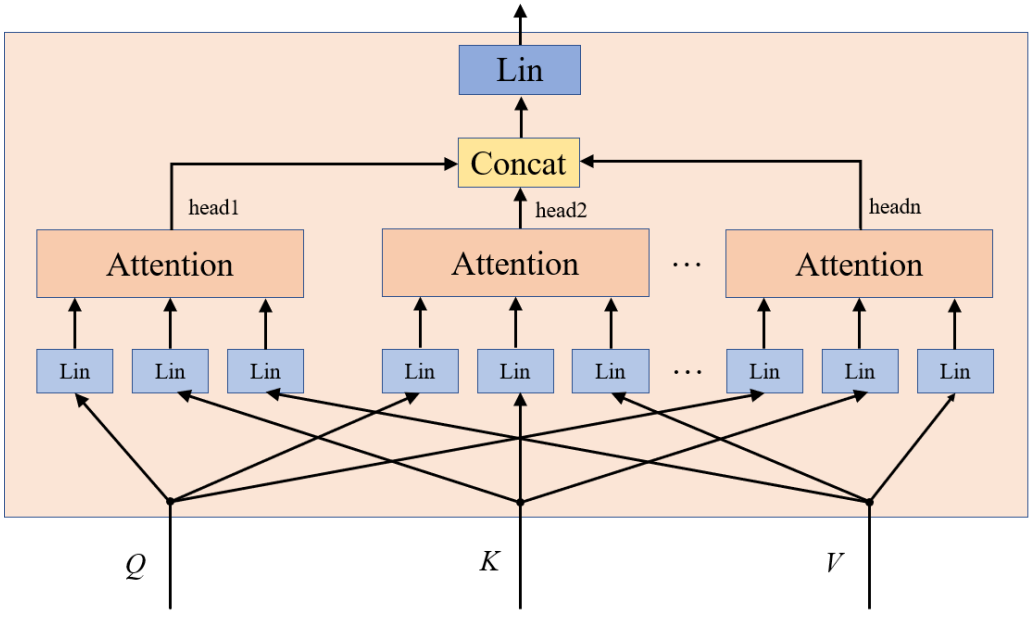


图 2-5 多头注意力机制

2.2 词嵌入模型

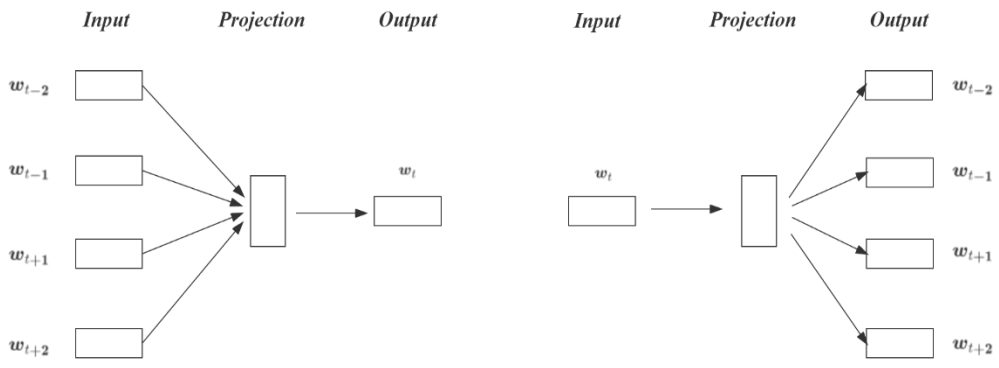
自然语言是一套用来表达语义的复杂系统，在这套系统中，词语是表义的基本单元，而将词语映射为实数域向量的技术叫做词嵌入技术，词嵌入也被称为词语的特征向量或表征。在深度学习领域，当前的主要研究趋势是追求一种通用的嵌入技术：在大型语料库中预训练词嵌入，它能够被添加到各种各样下游的任务模型中（情感分析、分类、翻译等）[41-43]，从而达到提升模型性能的目的。随着深度学习的发展，词嵌入的方式又被分为静态词嵌入和动态词嵌入。

2.2.1 静态词嵌入

静态词嵌入是通过大规模语料库训练得到的词向量，将其应用于下游任务时，不会因为上下文而改变词嵌入本身。具体的静态词嵌入技术可分为Word2Vec[44]和 Glove[45]等方法。

- **Word2Vec**

2013 年，Google 团队提出了开源的训练词嵌入向量的工具 Word2vec，其核心思想是利用关键词去预测其上下文或根据关键词的上下文去预测关键词，需要注意的是：词语的向量化表示是该模型训练的附带产物。Word2vec 模型包含两种结构，分别是连续词袋模型（Continuous Bag of Words）和跳字模型（Skip-gram）。



（a）词袋模型（Continuous Bag of Words） （b）跳字模型（Skip-gram）

图 2-6 Word2vec 模型

CBOW(continuous bag of words)模型的核心思想是：在一个句子中遮住目标单词，通过其前面以及后面的单词来推测出这个单词w。

首先规定词向量的维度V，对数据中所有的词随机赋值为一个V维的向量，每个词向量乘以参数矩阵W ($V \cdot N$维矩阵)，转换成N维数据，然后要对窗口范

围内上下文的词向量相加取均值作为 Projection 层的输入，Projection 层将向量的维度拉伸至一定程度后，输入至输出层，随后利用 softmax 分类预测目标词。最终用预测出的w与真实的w作比较计算误差函数，然后利用梯度下降算法调整参数矩阵。其具体过程如公式（2.12）-（2.14）所示。

$$x = \frac{1}{n-1}\sum_{w_j \in c} v(w_j) \tag{2.12}$$

CBOW 结构使用上下文词向量的平均值作为 Projection 层的输入，具体形式如上所示，其中 n 表示滑动窗口的大小，一般取做奇数，c表示目标词w的上下文，$v(w_j)$表示词语w的向量表示。

$$P(w|c) = \frac{exp(v'(w)^T x)}{\sum_{w' \in V} exp(v'(w')^T x)} \tag{2.13}$$

上述公式中，w为遮盖的词语，w'表示词典中的词语。最终，模型的目标是最大化下列似然函数：

$$\sum_{(w,c)\in D} \log P(w,c) \tag{2.14}$$

其中(w,c)表示预训练数据集中的文本序列$w_{i-(n-1)/2}, \cdots, w_{i+(n-1)/2}$。

Skip-gram 模型的核心思想是：模型根据目标单词来推测出其前面以及后面的单词，它的模型结构与 CBOW 正好相反，其输入是目标词，输出是目标词的上下文。首先，将目标词的词向量映射到 Projection 层，然后将 Projection 层的输出作为输入传递到输出层，通过这种方式，可以预测在目标词周围窗口内出现的上下文。换言之，相较于 CBOW 模型来说，Skip-gram 模型是将输入层和输出层交换位置，并在它们之间添加一个 Projection 层作为中间层，其具体过程如公式（2.15）-（2.16）所示。

$$P(w|w_j) = \frac{\exp\left(v'(w)^T v(w_j)\right)}{\sum_{w' \in V} \exp\left(v'(w')^T v(w_j)\right)} \tag{2.15}$$

其中w为中心词，w_j是上下文c中的词语。和 CBOW 模型相似，Skip-gram 模型的最终目标是最下化下列似然函数：

$$\max\left(\sum_{(w,c)\in D} \sum_{w_j \in c} \log P(w|w_j)\right) \tag{2.16}$$

除上述内容之外，由于连续词袋模型和跳字模型的最后一步都使用了 softmax，因此输出的结果是多分类，并且分类的数量是词库中词语的数量，所以存在时间复杂度过高和浪费计算资源的问题，针对这样的问题，有研究人员提出了负采样（Negative Sampling）和层级 softmax（Hierarchical Softmax）的优化方案。负采样的思想是随机选择一小部分的 negative words 来更新对应的权重，以此减少反向转播中的计算量；层级 softmax 的思想是利用哈夫曼树，将词频高的词语置于树中靠近根的叶子结点，从而将 v 分类问题转化为二分类问

题。利用上述两种优化方案，Word2vec 模型能够在大规模语料库中快速学习到词语的向量表示，并能够在不同的语料库中，学习到不同的语义。

- Glove 模型

上文中描述的 Word2vec 模型只虑到窗口内词与词之间的局部关系，忽略了窗口内外词语之间的关系，具有一定的局限性，因此， Pennington 等人[45]提出了新的词向量训练模型—Glove。

Glove 模型将语料库中的词频进行统计，作为学习词向量的重要特征，具体可分为以下三个步骤：

①构建词频共现矩阵X。在矩阵X中，X_{ij}表示词语i和词语j出现在同一滑动窗口内的频数，并根据滑动窗口内词语i和j之间的距离 d，设置权重衰减函数：decay=1/d，其物理意义为：距离越近的词语相关性越高，反之，则越低。

②得到词向量和频次共现矩阵的近似关系，其具体定义如下所示：

$$w_i^T w_j + b_i + b_j = \log(X_{ij}) \tag{2.17}$$

其中, w_i和w_j表示第i个词语和第j个词语的向量表征，b_i和b_j为偏置项，以便于更好地拟合数据，X_{ij}表示词语 i 和词语 j 出现在同一滑动窗口内的频数。

③构造损失函数。作者在论文中提出了一个新的加权最小二乘回归模型来构造损失函数。将等式（2.17）看作一个最小二乘问题，并在损失函数中引入权重函数$f(X_{ij})$，损失函数如下所示：

$$J = \sum_{i,j=1}^{V} f(X_{ij}) \left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log(X_{ij}) \right)^2 \tag{2.18}$$

其中，$f(X_{ij})$为权重函数，V表示词典大小。因为少见的共现携带有噪声，并且比频繁的共现携带更少的信息，因此添加权重函数可以避免对所有共现事件赋予相同的权重。

GloVe 是一种全局对数双线性回归模型，用于词向量的无监督学习，它结合了全局矩阵分解和局部上下文窗口方法两者的优点。GloVe 模型通过只训练词-词共现矩阵中的非零元素，而不是整个稀疏矩阵或大型语料库中的单个上下文窗口，有效地利用了统计信息，从而生成更富有表征意义的词向量。

2.2.2 动态词嵌入

上文中所描述的静态词嵌入，将其应用于不同的下游任务时，由于其本身的向量表示不会随着上下文的变化而改变，所以，难以解决一词多义的问题。动态词嵌入则能够根据不同的语境信息，赋予每个词语不同的语义特征，以此解决一词多义的问题。本小节将分别介绍 ELMo[46]、BERT[26]和 RoBERTa[47]等动态词嵌入技术。

- ELMo 模型

ELMo 模型能够通过不同的语句对同一个单词训练，从而得到不同的词向量，有效区分出同一个单词在不同语境下表示的不同含义（例如：apple 可以表示苹果，也可以表示 iphone）。ELMo 模型对语句的正向和逆向分别使用了双向的 LSTM，可以更好地捕获语句中上下文之间的关系，并使用了多层的 LSTM 结构（下图中是两层），底层的 LSTM 可以用于捕获句法信息，顶层的 LSTM 可以用于捕获语义信息。ELMo 模型结构图如下所示：

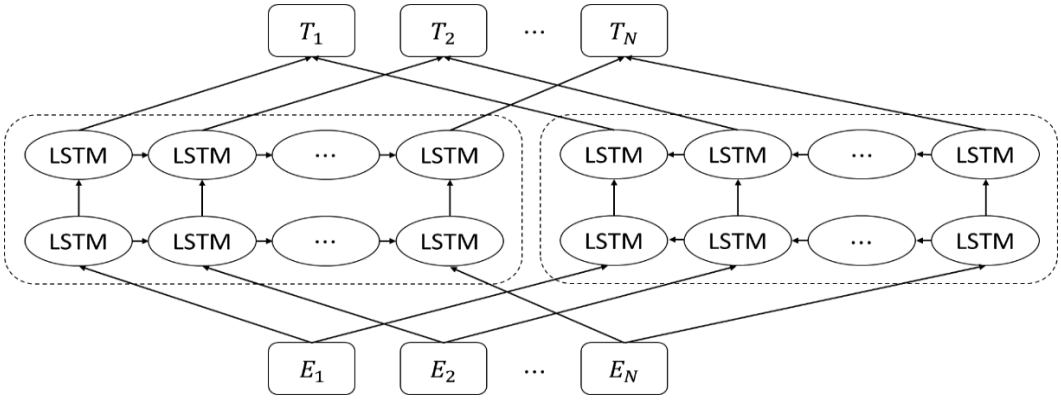


图 2-7 ELMo 模型

对于正向的部分（上图左边的虚线框部分），模型的输入E_k通过对单词使用 embedding 或是字符级的 CNN 得到初始向量表示$\boldsymbol{x}_k^{LM}$，将其输入到 LSTM 后，每一层 LSTM 都可以得到一个基于上下文语境的向量表达。顶层得到的向量表达经过 softmax 后就可以用于预测 k+1 位置的单词。从左至右的模型表达式为：

$$p(t_1, t_2, \dots, t_N) = \prod_{k=1}^{N} p(t_k | t_1, t_2, \dots, t_{k-1}) \tag{2.19}$$

其中，$(t_1, t_2, \dots, t_N)$表示句子中的的N个词语，根据前t_{k-1}个词语预测下一个词语t_k。反向的部分同理：

$$p(t_1, t_2, \dots, t_N) = \prod_{k=1}^{N} p(t_k | t_{k+1}, t_{k+2}, \dots, t_N) \tag{2.20}$$

综合正向和反向的部分，求极大似然函数的对数即可得到损失函数（取负数极小化）：

$$\sum_{k=1}^{N} \Big(\log p\big(t_k | t_1, t_2, \dots, t_{k-1}; \theta_x, \vec{\theta}_{LSTM}, \theta_s\big) + \log p\big(t_k | t_{k+1}, t_{k+2}, \dots, t_N; \theta_x, \overleftarrow{\theta}_{LSTM}, \theta_s\big) \Big) \tag{2.21}$$

其中，θ_x表示词向量的初始化表示，θ_s表示 softmax 归一化参数，$\vec{\theta}_{LSTM}$和$\overleftarrow{\theta}_{LSTM}$分别表示正反两个方向的 LSTM。

ELMo 是第一个实现动态词向量的模型，其核心思想是：利用迁移学习的方法，使得相同的词语在不同的语境中拥有不同的词嵌入表示，以此解决一词多义的问题。但是该模型也有一定的缺陷：LSTM 是串行机制，训练时间长，且由于 LSTM 本身的缺陷，容易带来梯度爆炸的问题。

- BERT 和 RoBERTa 模型

BERT（BidirectionalEncoder Representations from Transformer）是基于上下文的动态词嵌入模型，该模型的主要目的是使用大规模无标注语料进行训练，以获得包含丰富语义信息的文本向量表示形式。这些向量表示形式可以用于特定的自然语言处理任务，并在这些任务中进行微调。因此，模型可以被视为是一种通用的预训练模型，通过学习通用语言表示形式来为各种 NLP 任务提供支持。简言之，该模型旨在通过无监督训练，获得对文本的深度理解，然后将这种理解应用于各种 NLP 任务中。具体模型图如图 2-8 所示：

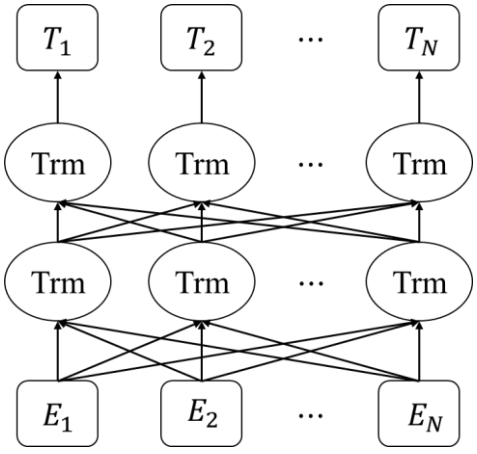


图 2-8 BERT 模型图

BERT 由多个双向 Transformer 编码器构成，共包含 12 个编码器层。相对于 ELMo 模型，BERT 同样采用“预训练-微调”的迁移学习策略。在预训练阶段，BERT 利用两种任务得到词语的向量表征，分别为：Masked Language Modeling（MLM）和 Next Sentence Prediction（NSP）。

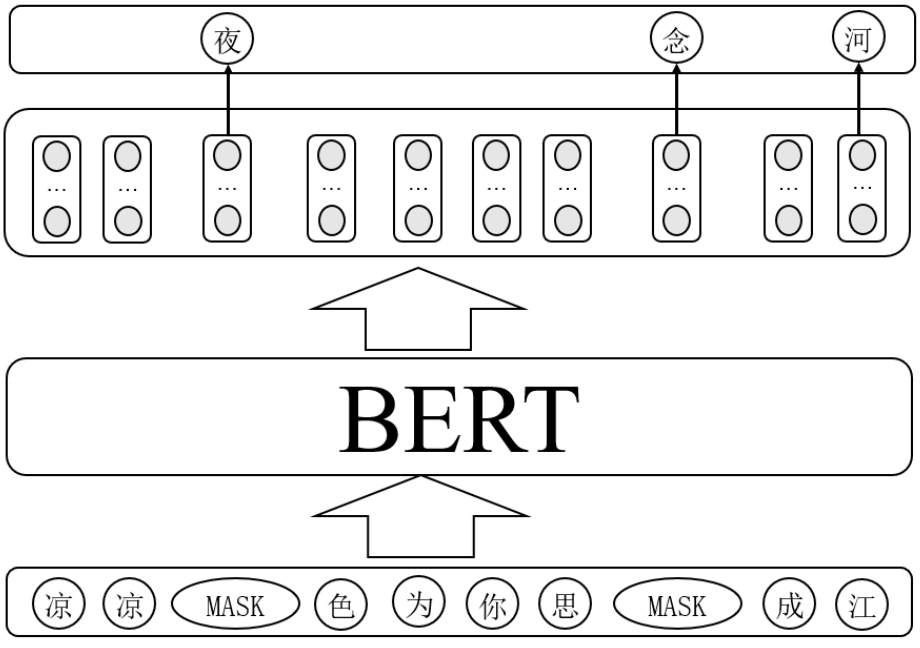


图 2-9 MLM 任务

MLM 任务与完形填空类似，该任务的核心思想为：给定一个句子，随机 mask 句子中的某些词，然后，根据其他词汇预测被随机 mask 的那几个词。在 BERT 原文中，作者在一句话中随机选择 15%的词汇用于预测，对于在原句中被抹去的词汇，80%情况下采用一个特殊符号 MASK 替换，10%情况下采用一个任意词替换，剩余 10%情况下保持原词汇不变。这么做的主要原因是：在后续微调任务中语句中并不会出现 MASK 标记，而且这么做的另一个好处是：预测一个词汇时，模型并不知道输入对应位置的词汇是否为正确的词汇（10%概率），这就迫使模型更多地依赖于上下文信息去预测词汇，并且赋予了模型一定的纠错能力。

NSP 任务和核心是利用上文测下文，具体可以描述为：给定一篇文章中的两句话，判断第二句话是否和第一句话存在前后关系。在该任务中，作者从原始语料库中抽取 50%的语句对作为正样例，另外 50%语句对的第二句，从语料库中进行随机抽取，作为负样例。

BERT 模型通过对 MLM 任务和 NSP 任务进行联合训练，使模型输出的每个字/词的向量表示都能尽可能全面、准确地刻画输入文本（单句或语句对）的整体信息，为后续的微调任务提供更好的模型参数初始值。相对的，BERT 也存在一定的缺陷，如模型的预训练需要大量的语料库和强大的算力等。

RoBERTa 建立在 BERT 模型的基础之上，但是，RoBERTa 并没有在模型层面对 BERT 进行任何改进，而是在预训练阶段去除了 Next Sentence Prediction(NSP)任务，并将静态 MASK 机制改进为动态 MASK，相对于 BERT，RoBERTa 训练数据更多，batch size 更大，性能更加显著。

2.3 句法分析

句法是指短语和句子的结构方式，表现为词语在句子中的排列方式及其相互关系。在处理句子信息时，句法信息是一条极其重要的规则化信息，一些完全相同的词语根据不同的排列形式进行组合，可能得到语义完全不同的句子，如：“我站在他身后”和“他站在我身后”，就是词语完全相同，但是语义相反的一对句子。句法分析是自然语言处理领域中的关键任务之一，是对句子级别的输入进行分析以得到其句法结构的处理过程，该任务的主要目的是：识别出句子所包含的句法成分，以及这些成分之间的关系，具体的句法分析结果可以使用句法依存树的方式来表示，如图 2-10。

图 2-10 句法依存树

上图中的“HED”、“SBV”等特殊标记为句法分析标注关系，目前，现存的句法分析标注关系及含义如下所示：

表 2.1 句法分析标注关系及含义

关系类型	Tag	Example
主谓关系	SBV	我送她一束花 (我 <-- 送)
动宾关系	VOB	我送她一束花 (送 --> 花)
间宾关系	IOB	我送她一束花 (送 --> 她)
前置宾语	FOB	他什么书都读 (书 <-- 读)
兼语	DBL	他请我吃饭 (请 --> 我)
定中关系	ATT	红苹果 (红 <-- 苹果)
状中关系	ADV	非常美丽 (非常 <-- 美丽)
动补结构	CMP	做完了作业 (做 --> 完)
并列关系	COO	大山和大海 (大山 --> 大海)

表 2.2 句法分析标注关系及含义

关系类型	Tag	Example
介宾关系	POB	在贸易区内 (在 --> 内)
左附加关系	LAD	大山和大海 (和 <-- 大海)
右附加关系	RAD	孩子们 (孩子 --> 们)
独立结构	IS	两个单句在结构上彼此独立
核心关系	HED	指整个句子的核心

对句法结构进行分析，一方面是自然语言理解的本身诉求，另一方面，它也能够为自然语言处理领域的其他任务提供支持，例如句法区别的统计机器翻译需要对源语言或目标语言（或者同时两种语言）进行句法分析[48-50]，但是这种语义分析通常以句法分析的输出结果为输入以便获得更多的指示信息，需要对模型的输入做出额外的处理。

2.4 文本摘要

文本摘要是指通过各种技术，对文本或者是文本的集合，抽取、总结或是精炼其中的要点信息，用以概括和展示原始文本的主要内容或大意。一般而言，生成的简短摘要必须满足信息量充分、能够覆盖原文的主要内容、冗余度低和可读性高等要求，从本质而言，文本摘要是一种信息压缩技术。根据实现技术方案的不同，可以分为抽取式文本摘要、生成式文本摘要。

抽取式摘要，即直接从原文中抽取一些句子组成摘要。抽取式的方法基于一个假设，一篇文档的核心思想可以用文档中的某一句或几句话来概括，那么摘要的任务就变成了找到文档中最重要的几句话。这种摘要方式本质上是个排序问题，给每个句子打分排序，再考虑冗余性、新颖性、多样性等做一些后处理，抽取高分的句子作为摘要。这个过程涉及到句子重要性评估算法和基于约束的摘要生成算法。这种方法天然的在语法、句法上错误率低，保证了一定的效果，但是也会存在冗余性、缺乏连贯性以及无法覆盖所有重要信息的等缺点。

生成式摘要，它试图通过理解原文的意思来生成摘要，其实就是模仿人类写摘要的方式，可能会使用原文中的词语，也可能会用新词 (未出现在原文中的词)来做表述。一般来说，大多数研究者都使用 Seq2Seq 的训练模式进行生成式摘要研究[51-54]。生成式摘要相比于抽取式摘要而言用词更加灵活 ，因为所产生的词可能从未在原文中出现过，但是容易产生词语易连续重复、违背文章原意以及产生未登录词的问题。

总的来说，抽取式的文本摘要直接从原文中摘取完整的句子作为文章的摘要，所以通常在语法层面的正确性以及对原文概括的准确性上比较占优势；生成式的文本摘要可以产生原文中没有的单词和短语，容易产生事实性错误，返回不符合事实的结果。

2.5 文本增强

数据增强(Data Augmentation) 是指通过对原始数据进行各种随机变换，生成更多的训练数据以增强模型泛化性能的方法，该方法最早应用计算机视觉领域[53]，通过图像旋转、裁剪、缩放和翻转等方式，能够有效地提高模型的识别

精度和鲁棒性，同时减少模型的过拟合现象。

在自然语言处理领域，数据增强技术可以通过对文本进行各种变换来增加训练数据的多样性，从而提高模型的泛化性能。近些年来，自然语言处理领域也提出了一些数据增强策略，如：①回译，将文本翻译成另一种语言，再翻译回原始语言，以此合成多样化和信息丰富的增强数据，以增加模型对语句的理解能力；②同义词替换，随机选择 n 个非停用词，使用同义词替换原词语，以增加语料库的多样性；③随机交换，随机选择句子中的两个词语交换位置，重复 n 次，以增加模型对句子结构的理解能力；④随机删除，根据概率 p 随机删除句子中的词语，以增加模型对上下文的理解能力。

数据增强技术已经被广泛应用于图像识别、自然语言处理等领域，通过对原始数据进行随机变换和扩充，可以获得更多的训练数据，并提高数据的多样性，从而使得模型更加准确地预测新的未知数据，以此提升模型的性能。

2.6 本章小结

本章主要对文本蕴含识别任务所需深度学习模型以及相关技术进行介绍。第一节介绍了文本蕴含识别任务对句子进行编码的循环神经网络，以及捕获句子重要信息的注意力机制；第二节主要介绍了词嵌入模型，分别对静态词嵌入和动态词嵌入进行了详细介绍；第三节对本文所需要的句法分析任务进行了简要分析和概括；第四节同样对本文所需的文本摘要技术展开了剖析，分别分析了抽取式摘要和生成式摘要的优缺点；第五节则是介绍了文本增强的诞生与具体策略。

3 融入句法结构和摘要信息的文本蕴含识别模型

该模型将自注意力和互注意力机制相结合，能够捕获句子的全局和局部交互信息，并在建模阶段融入句子的句法结构信息，从而更准确地推测句子之间的语义关系；收集公务员的部分试题，并利用摘要信息抽取技术，解决公务员试题中题目冗长和答案简短导致的长度不对称问题，最后，将该模型和文本蕴含识别的思想应用于这部分试题中。经实验验证，该模型在公共数据集和公务员试题上的表现优于多个基准模型

3.1 数据集构建

本文首先在英文数据集(SNLI)、中文数据集(CNLI)验证模型的有效性，然后将模型迁移至公务员试题中进行实验。在这一小节中，本文将具体介绍公务员试题收集和处理过程，在后续的 3.4.1 小节中将会具体介绍公共数据集：SNLI 和 CNLI 数据集。

在整个公务员试题中，包含主旨概括、意图判断、细节理解和细节查找等多种题型，我们将这些类型的试题进行多次筛选，最终选取了主旨概括和意图判断类型的试题，这是因为这两种类型试题的答案大多都是对题目本身的概括或总结，这是一种自然的语义蕴含语句对。满足语义蕴含条件的同时，还需要满足语义矛盾的语句对，因此，在将这些试题爬取后，我们又对这些试题进行了再次筛选，从 5127 条试题中，筛选出 4199 条试题，在人工判定中，这些试题都存在答案和题目语义相反的现象，在这一轮的筛选中，我们将这些答案和题目组成语义矛盾的语句对。基于以上两轮的筛选，从公务员试题中构建了 8398 组数据的语义蕴含和语义矛盾的语句对。本文将上述过程中处理的公务员试题数据命名为：CSEQ(Civil Service Examination Questions)，具体细节如表 3.1 所示。：

表 3.1 CSEQ 数据集

Label	**Train**	**Validation**	**Test**
Entailment	3487	356	356
Contradicition	3487	356	356
Total	6974	712	712