第 5 章　自然语言处理与理解

5.1　为什么要研究自然语言处理与理解？

自然语言是指人们在日常生活中使用的语言，与其对应的是如计算机编程语言这样的人工语言，或者在一些专业领域所说的音乐语言、绘画语言。

在人工智能中为什么要研究自然语言？最主要的一个原因就是**自然语言是人们进行交流的一种自然、便捷的重要方式**。如果一个智能产品能够理解自然语言，和人交流，人就可以通过自然语言要求这个智能产品完成一些任务。

当然不是所有的智能产品都必须和人交流。例如，在门禁系统这一类产品中，人们使用指纹、人脸、虹膜进行个人身份认证。在这个过程中，人不一定要和系统进行语言交流。但是，有些智能产品和人的交流是很必要的，这时用自然语言进行交流是非常方便的。例如，可以要求一个扫地机器人打扫某一个房间的地板，要求家务机器人清洁卫生间，也可以和一个问答系统聊天。

另外，和计算机系统交流，自然语言也不是唯一的方式，甚至都不一定是最好的方式。例如，有时人们更喜欢使用鼠标、写字笔和计算机绘图软件（如画笔、Photoshop）进行交流，让软件生成人们需要的图画。

研究自然语言的另一个重要原因是：人们发现在研究智能的时候，语言是不得不研究的一个对象。因为语言在智能中起着举足轻重的作用。换句话说，研究语言本身特别有助于研究人脑和智能，去理解人脑是怎么工作的，去理解生物智能，从而启发人们去研究人工智能技术。

5.2　自然语言处理与理解的一些任务

在自然语言处理与理解这个领域有很多实际的应用需求，下面列举几个比较主要的

任务。

1. 机器翻译

这个任务就是把一种语言的文本翻译为另一种语言对应的文本。例如，中译英、英译俄等。这是一个非常广泛的重要的应用需求。有时，人们需要由理解不懂的语言写成的文字，或者语音。例如，在互联网上想了解一个陌生语言的网页，在异国他乡想读懂餐馆的菜单等。

2. 自动摘要

这个任务是要求系统对一个长文本进行摘要，转变为符合要求的短文本。对于一个长文本，如几万字、几十万字的文章或者书籍，人们通常希望先读摘要了解其大致内容，然后决定是否开始购买或阅读。另外，在使用谷歌或百度搜索的时候，会得到很多相关的网页。为了让用户一目了然地“了解”搜索结果，系统需要自动对网页摘要，把摘要放在栏目名称下面。这样，用户就能快速了解这些网页大概在说什么。

3. 人机对话

这个任务是要求人和计算机系统使用自然语言对话交流。目前的一些智能台灯、智能音箱就具有这样的一些简单的功能。用户可以问：“今天天气怎么样?”“播放一首 *** 的歌”，之后这些产品会给出响应。

4. 机器写作

这个任务是要求计算机写作。这里要写作的可能是诗歌、小说等。

除了上面这些任务外，还有一些是研究人员定义的中间任务。例如，指代消解。

5. 指代消解

这个任务就是要确定名词、名词短语、代词之间的对应关系。例如，“出租车司机老李开得飞快。老李说……”，在这里需要确定第二句中的“老李”就是指第一句中的“出租车司机老李”；“校长在开学典礼上讲了话。他语重心长地说……”，这里需要确定代词“他”就是第一句中的“校长”。通常来说，这是研究自然语言处理与理解中的一个中间任务，并不直接对应一个实际应用需求。

5.3 自然语言处理与理解包含的几个层次

自然语言处理与理解包含对自然语言由低到高几个层次的处理过程，可以简单总结为下面三个层次。

- 字、词、短语
- 句子
- 篇章

在各自的层次，研究人员关注的问题是不同的。下面分别讨论各个层次的几项研究内容。

1. 字、词、短语

在这个层次上，需要解决很多问题。下面列举几个。

1）词态分析

在英文里，动词是有时态的。例如，“He has gone”。这时需要知道 “gone”是“go”的过去分词。

2）自动分词

中文的一句话中各个字之间是没有空格的。这时需要对句子做自动分词。例如，“物理学起来很困难”，分词后就是“物理|学起来|很|困难”。有的句子的自动分词容易出错。例如，“物理学是一门学问”分词后就是“物理学|是|一门|学问”。同样是“物理学”三个字，在上面两个句子中的划分结果就不一样。再比如，“南京市长江大桥”该怎么分？类似的句子还有很多。通常情况下，人在阅读时，分词（句读）是一个自然的过程，人不需要花费太多精力就能够做好。但是在阅读一些专业文章，或者阅读对于不熟悉的事物描述时，有可能会产生分词错误。自动分词也是这样，如果计算机程序遇到一个新词，因为它从未“见过”这个词，可能会出现错误。

3）词义消岐

一词多义是自然语言的一个通常的现象。因此，需要在一个句子中确定词的具体的含义。句子“I have no interest”，这里的“interest”指什么？是指利息？还是兴趣？句子“昨天我买了个苹果”，这里的“苹果”指水果？还是手机？通常来说，结合上下文和已有的知识可以解决这个问题。**目前的预训练大模型对这种问题解决得比较好**。

2. 句子

在这个层次上，需要解决的主要问题就是对句子做语法分析，这也被称作**解析**

(parsing)或者**解译**。解析的作用就是希望知道句子中词和词之间的关系。如下就是对一个句子“我明天要进行课堂汇报”解析后的结果。这是一个树结构，被称作**依赖树**(dependency tree)，树上的弧线代表了两个词之间的依赖关系，如图 5-1 所示。

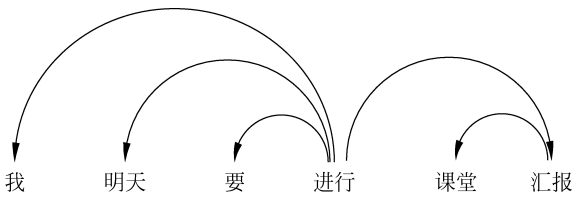

图 5-1　依赖树图例

根据依赖树可以知道，“进行”涉及两个主要方面，一个是它涉及的对象“汇报”，另一个是“谁”做“进行”这个动作；“明天”“要”是相对次要的方面；“汇报”前的“课堂”表示汇报的地点…… 这样的解译有助于理解句子。

3. 篇章

篇章是指由多个句子或者段落构成的有组织、有意义的文本。在这个层面，研究者关心句子之间是如何衔接的，从而让语义连贯。在这个层面涉及的任务有：问答系统，自动摘要，指代消解等。

5.4　词的表示

1. 传统表示方法

传统的词表示方法是使用一个字符串。例如，“旅店”“school”。和这种表示方法性质相同的是**独热向量**(one-hot vector)表示方法。独热向量表示方法把一个词表示为一个向量。这个向量中只有这个词对应的位置为 1，其他位置都为 0。例如，

$$\text{旅店}=[0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0]$$

$$\text{旅馆}=[0\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0]$$

$$\text{雨雪}=[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1]$$

这种表示方法有两个缺点。第一，向量的维数通常很大。因为每一维对应一个词，所以一般来说这个向量维数就是一个字典中词的数目。为了能够表示尽可能多的词，这个维数会在几万以上，如 10 万或 50 万。第二，从这样的表示上看不出两个词之间的关系。用通常的向量运算(如减法，内积运算)看不出“旅店”和“旅馆”更相似，“旅店”和“雨雪”更远。

约翰·鲁珀特·弗斯(John Rupert Firth,1890—1960),英国语言学家。他长期在伦敦大学任教,是现代语言学伦敦学派的创始人。他认为语言是人类生活的一种方式,并非仅仅是一套约定俗成的符号。

约书亚·本吉奥(Yoshua Bengio,1964年3月5日—)在21世纪初就开始使用神经网络方法研究自然语言处理,在深度神经网络方面做出了一系列出色的工作。他和杰弗里·辛顿、杨立昆因为深度学习共同获得了2018年度图灵奖。

2. 词向量表示方法

词向量表示方法的一个重要思想是:理解一个词需要理解其上下文。("You should know a word by the company it keeps",J. R. Firth 1957)。"每一个单词出现在不同的上下文中就是一个新的单词"。可以举一个例子来解释这一思想。例如,中文句子"这本书没意思""不好意思,让你久等了",这两句话中的"意思"的含义依赖其前后的词。

在这样的指导思想下,一个词也用一个取值为实数的向量表示。例如,旅馆=[0.281 0.792 −0.155 0.101 0.220 0.343 0.221]。这样两个向量之间的距离就可以反映它们之间的关系。如图5-2所示就是一组词向量映射到二维空间后的分布情况。在图中,各类水果之间距离比较近,动物、代词也是这样。而这三大类互相之间的距离比较远。

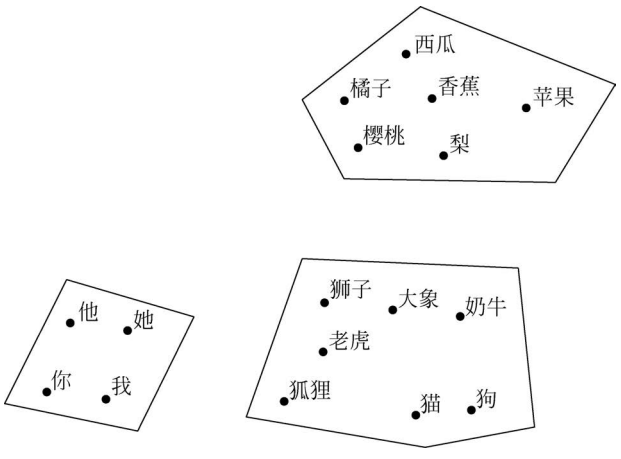

图5-2 一组词向量映射到二维空间后的分布示意图

目前性能最好的自然语言处理系统采用的就是词向量表示方法。这种方法也称为**词嵌入**(word embeddings)或者**词表示**(word representations)。

5.5　三大类方法

下面以机器翻译为例，介绍自然语言处理与理解中采用的三大类方法。在其他任务中，虽然具体的技术会不一样，但是它们的思路都比较类似。

1. 基于规则的方法

基于规则的方法是早期自然语言处理研究主要采用的方法。这种方法的主要思路是总结归纳出一套规则，在对一个具体句子进行翻译时，找到这套规则中适合这个句子的一条或几条进行翻译。

在这类方法中，总结出全面、完整的一套规则至关重要，而这也成为了研究中的主要困难。下面看几个具体的翻译示例。对于英译中任务，在句子“We shall have a symposium on Monday.（周一我们有一个研讨会。）”中，介词短语 on Monday 翻译成中文“周一”放在句首。假如据此总结出这样一条规则：把介词短语翻译的中文放在句首，这看起来是一个简单有用的规则。但是对于句子“We shall have a symposium on mathematics.”（我们有个数学研讨会。）就会翻译成“数学我们有个研讨会。”；对于句子“I have enjoyed hearing about your experience in Africa.”（我很乐意听你在非洲的经历。）就会翻译成“在非洲我很乐意听你的经历。”。这只是几个翻译示例。如果总结出其他的规则，也会遇到类似的问题。上面这几个句子的翻译可以让我们感受到一点其中的困难。

在实践中针对一个复杂问题，要总结出一套全面、完整的规则非常难。换句话说，要求这一套规则不多也不少、不出错、没有例外非常困难。

因此，规则的方法适用于某些简单的问题。这时，需要请对要解决的任务非常熟悉的专家总结出一套规则，来指导系统的实现。

规则系统的另一个问题是对于规则的维护比较困难。一个系统面对的实际问题一旦发生变化，就需要对系统中的规则进行修改、添加、删除等操作，而这也非常困难。

2. 基于统计的方法

这类方法的发展阶段大致在 1990 年～2010 年这二十年中。其主要思路就是把统计学的思想、方法和技术用于机器翻译。简单地说，就是对大量的翻译语料（如中英文对照例句）进行统计分析，寻找其中的统计规律（如一个词的前后通常会出现哪些词），并利用这些规律进行翻译。例如，要把“I am a student.”翻译成中文。就可以对大量英译汉的句子进行统计，得到的可能是下面的结果：

$$p(\text{我} / \text{“I am a student.”}) = 0.95$$

即：要翻译的英文句子是“I am a student.”时，第一个中文字是“我”的概率是 0.95。因此，第一个字就翻译为“我”。

$$p(\text{是} / \text{“I am a student.”“我”}) = 0.96$$

即：要翻译的英文句子是“I am a student.”，并且第一个中文字是“我”时，第二个字是“是”的概率是 0.96。因此，第二个字就翻译为“是”。类似地可以得到下面的结果：

$$p(\text{一个} / \text{“I am a student.”“我是”}) = 0.70$$

$$p(\text{学生} / \text{“I am a student.”“我是一个”}) = 0.98$$

$$p(\text{。} / \text{“I am a student.”“我是一个学生”}) = 0.98$$

这样，就得到了翻译的最后结果：“我是一个学生。”这里的句号“。”被当作一个词来翻译。

在这类方法中，需要用语料训练一个模型，得到各个词之间关系的概率，然后再把训练好的模型用于实际的翻译任务。这与计算机视觉、计算机听觉采用的思路相同。这个阶段得到了很多成果，是一个非常重要的阶段。

基于统计方法的机器翻译(statistical machine translation，SMT)也存在很多问题。下面举几个例子。

语言问题很复杂，因此需要针对每一个特殊的语言现象做特殊考虑，如中文中“把”字句就比较特殊。“小明擦了桌子”说成“小明把桌子擦了”，这就需要针对这种现象进行分析，使得翻译的词的顺序是正确的。另外，“在食堂吃”说成“吃食堂”，也需要针对这种情况做特殊考虑。

需要维护外部词库一类的数据库。在分析自然语言时，需要用到很多的字典或数据库，如地名字典、机构名称字典、成语字典等。除了建立字典，还需要对字典维护好，如修改、添加、删除。

机器翻译系统涉及很多模块，每一个模块都很复杂，都需要一个专门的团队来完成。团队之间的协作非常困难。

此外，当需要翻译一种新的语言时，所有的上述过程需要重新进行一遍。

3. 基于深度学习的方法

在深度学习时代，机器翻译系统的输入是要翻译的句子，输出是翻译好的句子，中间的模型是一个深度神经网络。神经网络结构可以是 LSTM、Transformer 等序列神经网络模型。和解决计算机视觉问题一样，机器翻译的神经网络方法也是利用大量成对的两种语言的句子，端到端地训练这个神经网络。

5.6　Transformer

Transformer 是一个针对自然语言处理与理解的模型。这个模型比较复杂。下面介绍它采用的一些关键技术。

1. 编解码框架

Transformer 采用了如图 5-3 所示的**编解码框架**(encoder-decoder framework)。输入 x_1、x_2、x_3、x_4 构成的句子(字词序列),经过“编码器”将其映射到一个被称作**语义空间**(sematic space)的向量空间,然后将其“语义”向量经过“解码器”翻译为 y_1、y_2、y_3 构成的新的句子(字词序列)。这个过程类似于人在翻译时的工作过程,先对于要翻译的句子进行理解(编码),然后再翻译为另一种语言(解码)。

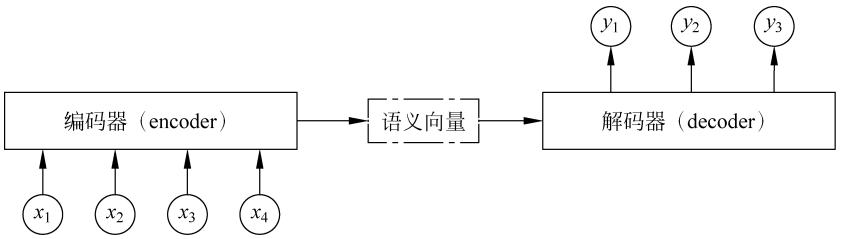

图 5-3　机器翻译的编解码框架

有意思的是,经过编码以后得到的向量有语义的成分。如图 5-4 所示是一个机器翻译系统中的几个句子的语义空间向量在二维空间展开的情况。句子“我在操场上被她搀扶着”,和“在操场上,她搀扶着我”,两句话词的顺序很不一样,并且一个是主动句,另一个是被动句。按照以前的方法,很难得到这两个句子意思相近的结论。但是人们知道,这两句话说的是同一件事,而在语义空间中它俩距离比较近。

对于句子“在操场上,她搀扶着我”“在操场上,我搀扶着她”。这两个句子大致的结构、用词差不多,只不过主语和宾语不同,当然含义就不同。在语义空间中,这两个句子的向量距离则比较远。

这样的向量空间包含了句子的语义信息:语义相似的句子距离比较近,语义不相似的句子距离比较远。得到句子的语义,是研究自然语言处理与理解的关键。这有利于完成自然语言处理的一系列任务。

2. 多层编解码结构

计算机视觉任务中的模型采用了多层卷积神经网络,从而很好地获得了各级特征,以

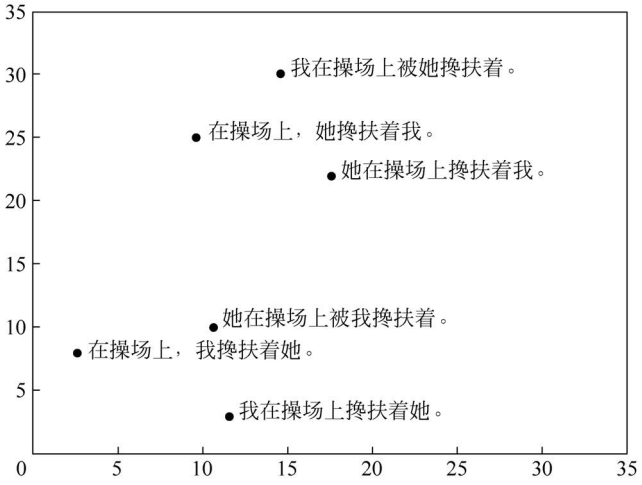

图 5-4　一个机器翻译系统中的几个句子的语义空间向量在二维空间的展开

及特征之间的关系。在 Transformer 中也是这样，为了能够提取词、词组、短语、句子，以及句子之间的关系和语义信息，编解码部分都采用了多层结构，如图 5-5(a)所示。其中的编码器和解码器的结构如图 5-5(b)所示。

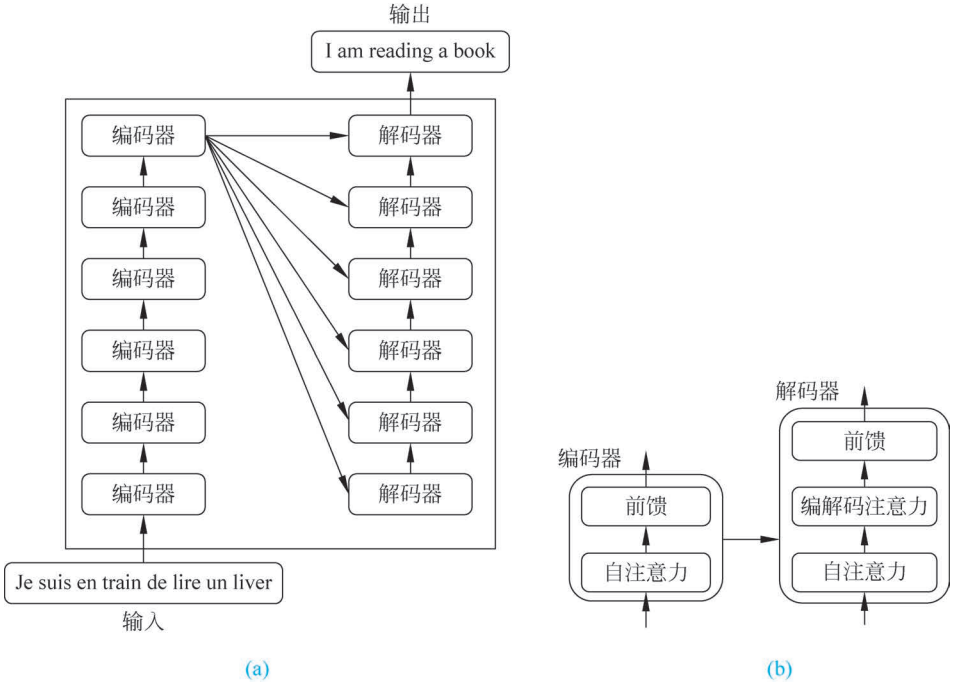

图 5-5　Transformer 的多层编解码结构

数据通过低层的注意力和自注意力模块，模型获得了词之间的关系。在更高层，逐渐获得了词组、短语、句子及句子之间的关系。在自注意力模块之后连接一个前馈网络，对获

得的特征进行非线性变换。

和卷积神经网络不同的是，这里使用了“自注意力”模块。

3. 注意力与自注意力

对于图 5-6(a)，如果要求人们只关注图中的小狗，基本上人们的注意力就如图 5-5(b)所示，白色的区域是人们关注的地方。如果白色的区域是一个非黑即白的地方，那就是只关注一个非常明确的、有边界的区域，一点也不关注黑的地方，这被称作**硬注意力**(hard attention)。但是人的注意力通常不是这样，而是如图 5-6(b)所示，白色区域中间是最被关注的地方。距离中心越远的地方，被关注的程度也越小，这被称作**软注意力**(soft attention)。在目前的 Transformer 模型中，用到的是软注意力。

(a)　　(b)

图 5-6　注意力示例

注意力这样的机制可以用如下数学公式来表示：

$$V(x)=\sum_{n=1}^{m} w_n V_n \tag{5-1}$$

式中，V_n 表示第 n 个被关注对象的值；w_n 是第 n 个被关注对象的权重。得到的 V 是对 m 个对象的关注的总和。如果 w_n 越大，说明 V_n 受到的关注就越多，如图 5-6(b)中比较白的区域的像素；w_n 比较小，说明 V_n 受到的关注就比较少，如图 5-6(b)中比较暗的区域的像素。图 5-6(b)就是把对图像中狗的注意力可视化的结果。

注意力机制也可以用于自然语言处理。比如分析下面这句话中各个词之间的关系，“峨眉山的猴子在吃香蕉因为它很饿”。如果“它”的注意力在“猴子”，如图 5-7(a)所示(颜色深表示注意力的权重大)。就说明“它”和“猴子”这两个词存在某种关系。我们知道，这句话中，“它”就是指 “猴子”。看起来，注意力机制得到的词之间的关系对于理解句子是有帮助的。

由于这里关注和被关注的词都来自同一个句子，因此被称作**自注意力**(self-attention)。

当然，研究人员希望“它”对“猴子”的注意是这个模型自己“找到”的，而不是由人指定的。不仅如此，实际上“它”还和其他词有关，研究人员还希望“它”关注到“很饿”和“吃”。如果这样，由于需要关注的词太多，导致一个注意力模块负担太重而无法“找到”需要关注的所有词。因此，在 Transformer 中一个注意力模块只负责找到一个关注点，如“猴子”，这样的一个注意力模块被称作是一个“头”。为了让它能找到更多的关注点，就设置了多个“头”。头的个数与需要输入的序列长度和复杂程度有关。图 5-7(a)所示的是，不同的颜色深浅代表不同的头和注意力的情况。

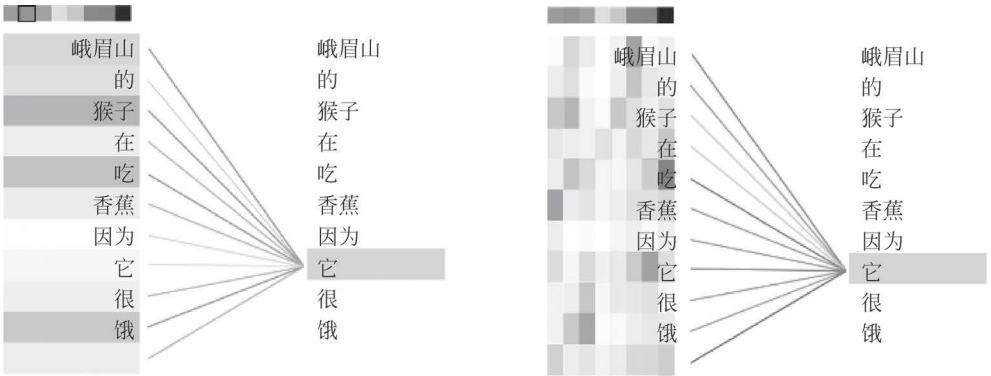

图 5-7　自然语言中的自注意力

这里的“头”(脑袋)用得很形象。一个脑袋的关注力有限，因此让它只关注一个点。但是要解决的语言问题又很复杂，因此就多准备几个脑袋来解决它，让不同的脑袋关注不同的点。

4. 注意力的计算

注意力是按照式(5-1)计算的。下面以句子“峨眉山的猴子在吃香蕉因为它很饿”为例，解释注意力的计算。

假设当前已经知道了其中每一个 token(在英文中，它是比单词更小的单位，如词根、词缀等子部分)的向量：v(峨眉山)，v(的)，v(猴子)，…，v(它)，…，如果需要计算句子中的“它”对于“猴子”的注意力，就可以计算v(猴子)和v(它)两个向量的内积，以此作为其注意力权重的依据。v(它)和每一个词向量计算内积，就可以得到“它”对于每一个词的权重。最后把这些权重归一化，就构成了最后的注意力权重向量 w。归一化就是计算“它”对于每一个词的权重之和 S，每一个权重再除以 S。归一化的一个目的是为了所有的权重向量之间可以比较。

例 5.1　注意力的计算。假设$\boldsymbol{v}$(峨眉山)=[1,3],$\boldsymbol{v}$(的)=[1,1],$\boldsymbol{v}$(猴子)=[3,1],$\boldsymbol{v}$(它)=[4,2],可以计算$\boldsymbol{v}$(它)与 4 个向量的内积如下：

内积($\boldsymbol{v}$(猴子),$\boldsymbol{v}$(它))=3×4+1×2=14；内积($\boldsymbol{v}$(峨眉山),$\boldsymbol{v}$(它))=1×4+3×2=10；

内积($\boldsymbol{v}$(的),$\boldsymbol{v}$(它))=1×4+1×2=6；内积($\boldsymbol{v}$(它),$\boldsymbol{v}$(它))=4×4+2×2=20

下面对内积做归一化。上面 4 个内积总和为 14+10+6+20=50。归一化后的权重为

内积($\boldsymbol{v}$(猴子),$\boldsymbol{v}$(它))/50=7/25；内积($\boldsymbol{v}$(峨眉山),$\boldsymbol{v}$(它))/50=1/5；

内积($\boldsymbol{v}$(的),$\boldsymbol{v}$(它))/50=3/25；内积($\boldsymbol{v}$(它),$\boldsymbol{v}$(它))/50=2/5

2/5,7/25,1/5,3/25 就是“它”分别对于“它”“猴子”“峨眉山”“的”的注意力权重。

在 Transformer 模型中，并不是利用两个 token 的向量直接计算内积，而是通过这两个向量向子空间投影，用投影后的两个低维向量计算内积。如上所示，再用这些内积计算权重向量 $\boldsymbol{w}$。最后，用这些权重对被注意的向量加权求和来更新“它”的向量$\boldsymbol{v}$(它)。

例 5.2　注意力的计算。假设$\boldsymbol{v}$(峨眉山)=[1,3],$\boldsymbol{v}$(的)=[1,1],$\boldsymbol{v}$(猴子)=[3,1],$\boldsymbol{v}$(它)=[4,2],如果这些向量向两个子空间 $\boldsymbol{s}_1$=[1,1],$\boldsymbol{s}_2$=[3,1]分别投影，分别计算“它”对于“峨眉山”“的”“猴子”“它”的注意力权重。

首先计算 4 个向量在子空间 $\boldsymbol{s}_1$=[1,1]上的投影向量(这里投影后的向量是一个一维标量)：

$\boldsymbol{v}_1$(峨眉山)=1×1+3×1=4；$\boldsymbol{v}_1$(的)=1×1+1×1=2；

$\boldsymbol{v}_1$(猴子)=3×1+1×1=4；$\boldsymbol{v}_1$(它)=4×1+2×1=6

计算$\boldsymbol{v}_1$(它)与其他向量的内积如下：

内积($\boldsymbol{v}_1$(猴子),$\boldsymbol{v}_1$(它))=4×6=24；内积($\boldsymbol{v}_1$(峨眉山),$\boldsymbol{v}_1$(它))=4×6=24；

内积($\boldsymbol{v}_1$(的),$\boldsymbol{v}_1$(它))=2×6=12；内积($\boldsymbol{v}_1$(它),$\boldsymbol{v}_1$(它))=6×6=36

对内积做归一化。上面 4 个内积总和为 96。归一化后的权重为

内积($\boldsymbol{v}_1$(猴子),$\boldsymbol{v}_1$(它))/96=1/4；内积($\boldsymbol{v}_1$(峨眉山),$\boldsymbol{v}_1$(它))/96=1/4；

内积($\boldsymbol{v}_1$(的),$\boldsymbol{v}_1$(它))/96=1/8；内积($\boldsymbol{v}_1$(它),$\boldsymbol{v}_1$(它))/96=3/8

3/8,1/4,1/8,1/4 就是“它”分别对于“它”“峨眉山”“的”“猴子”的注意力权重。

然后计算 4 个向量在子空间 $\boldsymbol{s}_2$=[3,1]上的投影向量(这里投影后的向量也是一个一维标量)：

$\boldsymbol{v}_2$(峨眉山)=1×3+3×1=6；$\boldsymbol{v}_2$(的)=1×3+1×1=4；

$\boldsymbol{v}_2$(猴子)=3×3+1×1=10；$\boldsymbol{v}_2$(它)=4×3+2×1=14

计算$\boldsymbol{v}_2$(它)与其他向量的内积如下：

内积($\boldsymbol{v}_2$(猴子),$\boldsymbol{v}_2$(它))=10×14=140；内积($\boldsymbol{v}_2$(峨眉山),$\boldsymbol{v}_2$(它))=6×14=84；

内积($\boldsymbol{v}_2$(的),$\boldsymbol{v}_2$(它))=4×14=56；内积($\boldsymbol{v}_2$(它),$\boldsymbol{v}_2$(它))=14×14=196

对内积做归一化。上面 4 个内积总和为 476。归一化后的权重为

内积($\boldsymbol{v}_2$(猴子),$\boldsymbol{v}_2$(它))/476=5/17；内积($\boldsymbol{v}_2$(峨眉山),$\boldsymbol{v}_2$(它))/476=3/17；

内积($\boldsymbol{v}_2$(的),$\boldsymbol{v}_2$(它))/476=2/17；内积($\boldsymbol{v}_2$(它),$\boldsymbol{v}_2$(它))/476=7/17

7/17,3/17,2/17,5/17 就是“它”分别对于“它”“峨眉山”“的”“猴子”的注意力权重。

根据上面例子可以知道，词向量向不同的子空间投影后得到的注意力权重是不同的。因此用这些权向量更新后的向量也就不同。

那么词向量应该向哪个子空间(对应于投影矩阵)投影才更合适呢？在 Transformer 模型中，投影的子空间是模型的参数，是在模型训练中学习得到的。

图 5-8 给出了“machine learning”这个短语的注意力计算过程和词向量的更新过程。首先“machine ”“learning”两个词向量 $\boldsymbol{x}_1$、$\boldsymbol{x}_2$ 分别向一个子空间投影，得到查询向量 $\boldsymbol{q}_1$、$\boldsymbol{q}_2$，然后再分别向两个子空间投影得到键向量 $\boldsymbol{k}_1$、$\boldsymbol{k}_2$ 和值向量$\boldsymbol{v}_1$、$\boldsymbol{v}_2$；用 $\boldsymbol{q}$、$\boldsymbol{k}$ 两个向量计算内积，得到图中的①② ；然后除以$\sqrt{d_k}$，其中，d_k 是 $\boldsymbol{q}$、$\boldsymbol{k}$，$\boldsymbol{v}$ 的维度($\boldsymbol{q}$、$\boldsymbol{k}$、$\boldsymbol{v}$ 的维度相同)，得到了各个权重③④ 。在 Transformer 模型中，使用 Softmax 函数对权重进行归一化，得到归一化后的权重⑤⑥，其计算公式如下：

$$\boldsymbol{z}=\text{Attention}(\boldsymbol{q},\boldsymbol{k},\boldsymbol{v})=\text{Softmax}\left(\frac{\boldsymbol{q}\cdot\boldsymbol{k}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{v}$$

$$\text{Softmax}(\boldsymbol{x}_i)=\frac{\mathrm{e}^{x_i}}{\sum_j \mathrm{e}^{x_j}}$$

最后使用归一化后的权重对值向量$\boldsymbol{v}_1$、$\boldsymbol{v}_2$ 加权求和，得到最后的更新向量 $\boldsymbol{z}$。

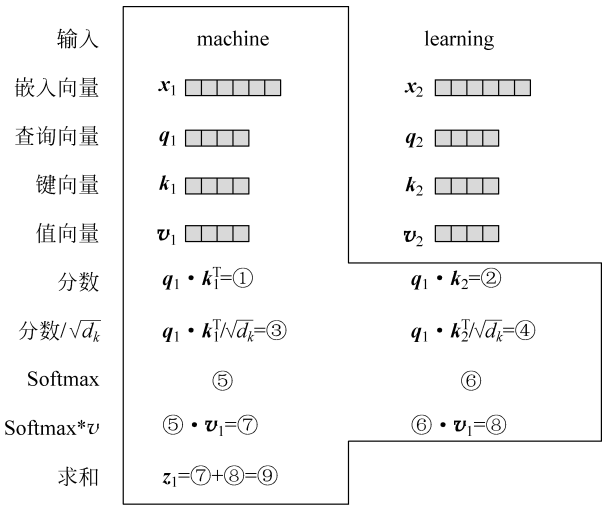

图 5-8　注意力权重的计算过程

5. 注意力与卷积操作的比较

在计算机视觉章节中介绍了卷积操作。一个 3×3 的卷积操作就是用 3×3 的模板上的数值和图像对应的像素灰度相乘，然后对所有的乘积求和。其公式也就是式(5-1)的形式。所以，卷积模板中的参数也可以看成注意力的权重。卷积操作就是按照每一个模板确定的注意力模式去寻找图像上对应的片段。简单地说，也就是在关注的一个图像的小的片段内，各个像素灰度之间的关系。

既然这二者之间的操作很相似，为什么在这里不直接使用卷积操作，而使用一个所谓新的“注意力”模块呢？

可以发现，卷积模板通常都比较小。这样做有两个原因：一方面是在提取特征时，图像中一个像素的灰度通常只和它一个小范围内的灰度有关。例如，一张桌子上有水杯、书和笔。书上一个像素周围的像素基本上仍然是书。但是和它比较远的地方可能是水杯、笔、桌面或者是其他物体。书上一些局部的特征和周围的物体(水杯、笔、桌面)的局部特征之间的关系一般不够紧密。因此，局部的卷积操作对于物体的识别通常更有意义。另一方面，做一个 3×3 的卷积，会导致图像的周边有一个像素宽的边无法得到有意义的卷积结果。如果卷积核非常大，卷积结果中有意义的区域会非常小。

而在一个句子中，一个词不只是和周围很近的词有关，也和比较远的词有关。如图 5-7 所示，“它”不只和“饿”有关，也和“猴子”有关。如果要采用卷积的方法寻找句首和句尾词的关系，这个卷积核就需要非常大，而卷积就无法操作和执行。特别是在一些长的文章中，如果要分析“首尾呼应”现象，就要分析更远的词之间的“注意”关系。

5.7　BERT

来自 Transformer 的双向编码器表示(bidirection encoder representations from Transformers，BERT)是一个基于 Transformer 的模型。其特点是利用了自然语言本身的特点，设计了两种**自监督学习**(self-supervised learning)方式，从而在不需要专门请人标注数据的情况下，做好了模型的训练，并能完成一些自然语言处理与理解的任务，如图 5-9 所示。

根据前面章节的内容可以知道，在模型训练阶段，通常需要对数据进行标注，利用标注好的数据训练一个神经网络，如图像分类就是这样。在自然语言处理中，对于机器翻译这个任务，可以利用已经存在的两种语言之间的对应例句作为标注数据训练模型。而自然语言处理的任务很多，如果要完成其他任务，如何训练一个语言模型？

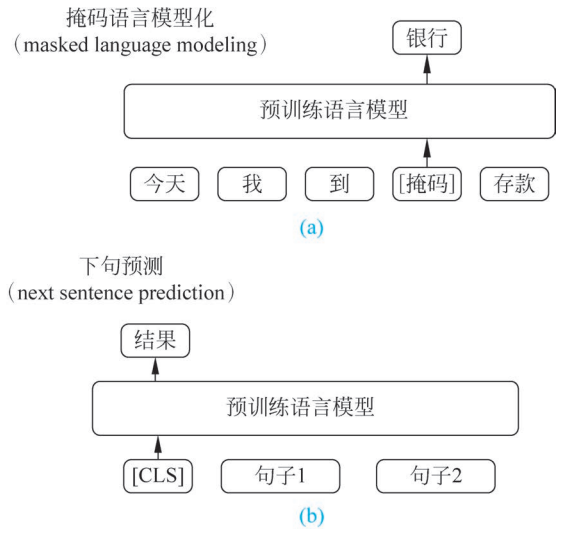

图 5-9　BERT 的两个自监督任务

BERT 的一个做法如下。对于一句话，如“今天我到银行存款”，如果随机选择其中的一个词，如“银行”，将其遮蔽掉，要求这个模型利用这个词周围已有的词来恢复这个词。在这样的任务中，被遮蔽的词实际是已经知道的，可以作为这个训练数据的标签。而现有的句子是大量的，就可以产生大量的训练数据。为了能够把被遮蔽的词恢复出来，“逼着”这个模型对于句子中各个词之间关系建模，如图 5-9(a)所示。

BERT 的另一个做法是又设计了一个新任务：判定两个句子是否是一前一后的两句话。为此，就从文本中选择了一前一后两句话，作为先后关系正确的数据，标签为 1。然后又随机选择了两个不是先后关系的句子，作为先后关系错误的数据，标签为 0。因为已有的文本非常多，因此就可以生成大量的包含正确顺序和错误顺序的句子对作为输入的训练样本，让模型在输出端输出 1(顺序正确)，或者 0(顺序错误)。这是一个两类分类问题。用这个任务训练的模型，就具有能力在第一句话后面很好地衔接第二句话，如图 5-9(b)所示。

这两个任务都是通过程序自动生成的训练数据，而不是由人工专门标注的数据，所以这样的**自监督学习**更容易实现。

因此，BERT 的第一个任务是为了一句话说得很顺畅，第二个任务是为了第二句话和第一句话衔接得好。这是自然语言处理与理解中非常核心的问题。所以，这个模型就具有了完成其他任务的基础。这类模型被称作**预训练模型**(pre-trainning model)。训练好这个模型后再在其他任务中**微调**(fine tune)，也就是利用其他任务的训练数据在这个模型上继续训练，从而具备了很好地完成其他任务的能力。

5.8　OpenAI 公司的 ChatGPT

OpenAI 公司在 2018 年研制了生成式预测训练 Transformer(generative pre-trained transformer,GPT)模型,并对它不断改进得到了 GPT-2、GPT-3 和 ChatGPT。

GPT 模型采用了自监督学习方式进行预训练。和 BERT 不同,给定一个句子或者几个连续的句子时,GPT 利用句子前面的词序列来预测下一个单词。当然,这样的数据非常多,能够满足训练好一个模型的要求。

此外,自然语言处理与理解中的任务非常多。看下面这个例子。当给了一句话,“我喜欢这本书”。如果要判断这句话的情感倾向是正面还是负面,这就是一个情感分类问题;如果要判断这句话是否符合语法,这就是一个语法判断分类问题;如果要判断这句话的主题,这就是一个句子主题分类问题。虽然输入的都是同一句话,但是由于分类的任务不同,通常就需要对每一个任务设计一个分类模型。

如何把这些不同的任务都用一个模型来完成? ChatGPT 使用了**提示学习**(prompt learning)技术解决了这个问题。

提示学习把上面举例中的分类问题转化成一个语言生成问题:输入“我喜欢这本书,我的情感是_____”输出是:正面。通过加入了“我的情感是”这样的提示信息,使得需要模型的输出有了具体方向。通过提示学习的方法,就可以让一个模型能够完成各种任务。当需要完成不同的分类任务时,就可以把任务的要求以“提示”的方式“告诉”模型。因此,ChatGPT 就可以用同一个模型实现机器翻译、问答、文本生成、文本摘要等各种任务。

ChatGPT 在 GPT-3 的基础上还采用了下面三个步骤继续训练网络。

(1) 取出提示学习的数据集的一部分数据,如“我喜欢这本书,我的情感是_____”,请人专门标注提示的答案,如“正面”。并用这些标注好的数据在 GPT-3 的基础上继续训练网络。

(2) 由于 GPT-3 是生成模型,所以对于提示数据集中其他的数据,模型可以生成多个答案。如对一个物理现象的解释可以有多种答案一样。这时,请人对这些答案的好坏排序。利用这些带有排序信息的数据训练另一个模型来自动评判答案的好坏,这被作为一个奖励模型用于下面一步。

(3) 采用强化学习(机器学习章节中的内容)方法继续优化 GPT-3 模型。

经过上述步骤,就可以大幅提高模型的性能。

除了 ChatGPT,被提出的其他一些预训练大模型都是以 Transformer 为基础模块,用大量数据进行自监督学习的预训练模型。

5.9 一个机器翻译的例子

下面看一段英译中的例子。给定下面这段英文：

“Meetings, seminars, lectures and discussions represent verbal forms of information exchange that frequently need to be retrieved and reviewed later on. Human-produced minutes typically provide a means for such retrieval, but are costly to produce and tend to be distorted by the personal bias of the minute taker or reporter. ”

在 2015 年以前，其一个翻译系统的中文译文如下：

“会议，研讨会，演讲和讨论代表频繁地需要以后被检索和被回顾信息交换的的口头形式。人被生产的分钟典型地提供手段为这样的检索，但是昂贵生产和倾向于由周详接受人或记者的个人偏心变形。”

2020 年这个系统的升级版本给出的译文如下：

“会议、研讨会、讲座和讨论是口头形式的信息交流，经常需要在以后检索和审查。人工制作的会议记录通常为这种检索提供了一种手段，但制作成本很高，而且容易被记录者或报告者的个人偏见所扭曲。”

从上面这个例子中可看出，2015 年以前翻译结果中的用词和词的顺序等都导致了翻译出来的句子无法被人理解，而 2020 年的翻译有了长足的进步。

5.10 机器对话和问答

机器对话和问答是人工智能里面特别重要的应用。和机器翻译的研究类似，机器对话和问答方法也有传统的方法和深度神经网络方法。

当前，预训练语言大模型已经能够很好地进行对话和问答。根据大量的对话和问答结果，人们发现，这些预训练大模型中具有了大量常识。同时也发现，它的对话内容不能保证是完全正确的。它是会出错的。


对话举例

请扫描二维码，阅读几个对话系统的对话举例。

5.11 文本生成

文本生成，包括写故事、写诗等。和机器翻译的研究类似，该方面的研究也有传统方法和深度神经网络方法。

下面列举几个文本生成系统的例子。

1. 诗歌

科普杂志《科学》(*Scientific American*)于 20 世纪 90 年代刊登的一个诗歌生成系统，它采用了固定的诗歌模板，然后按照模板添加词语生成诗歌。该系统生成了一首诗歌：

一位妇人藏了五只灰色的小猫

在破旧的汽车里

此时那位哀伤的乡农

唤起你痛苦的回忆

2. 九歌

清华大学计算机系研发了一个诗歌生成系统——九歌(http://jiuge.thunlp.org/)。它生成的一首诗如下：

诉别离

离别恨难分

琵琶不忍闻

断肠空有泪

明月已无魂

3. 微软对联生成器

微软公司研制了对联生成器(http://duilian.msra.cn/intro/intro.htm)。对联生成器可根据上联生成下联。

上联：苏堤春晓秀

下联：平湖秋月明

4. 薇薇诗歌生成器

这是清华大学语音与语言实验中心发布的一个系统。它生成的一首诗歌如下：

早梅

春信香深雪

冰肌瘦骨绝

梅花不可知

何处东风约

5. ChatGPT

2022 年发布的 ChatGPT 也可以根据要求进行写作，包括写小说、诗歌、总结报告等，表现出很好的性能。

5.12 生成的文本的评价

在回归问题和分类问题中，算法的自动评价相对简单。因为每一个训练数据都有标签，测试系统时，只要判断算法的输出与数据的标签是否一致（分类问题），或者相差多少（回归问题）就可以得出系统的性能指标，如分类准确率，或者回归的误差。

但是在文本生成这个问题上，算法的自动评价比较困难。对于文本生成任务，通常来说没有一个标准答案，如诗歌、小说、总结报告的写作没有唯一的标准答案。两篇都是很优秀的诗歌从谋篇布局到文字使用可能完全不同。因此，不能像回归和分类问题一样可以让算法自动给出一个非常合理的评价结果。不只是文本的写作，在对话和问答系统、机器翻译系统中都存在这样的困难。

评价一段文本的好坏，从不同角度看会有不同的标准。为此，研究人员设定了一些评价准则用于评判文本生成的结果的好坏。下面列举几个。

（1）正确性。生成的文本是否正确。

（2）风格。用于判断生成的文本是否满足要求的风格。如：生成的文本是否是要求的诗歌？是否是七言绝句？生成的文本是否具有鲁迅的风格？

（3）用词的多样性。生成的文本用词是否丰富和多样。在有些应用中，需要模型产生的文本包含丰富多样的词汇。

（4）和输入的相关性。举例来说，如果要求系统以“诉别离”为题写诗歌，系统生成的诗歌与题目不相关，这一准则的评分就比较低。

（5）针对性的一些指标。如生成摘要，评价生成的摘要长度是否符合要求，摘要长度和原文长度之比。如生成韵律诗歌，评价其韵律是否符合要求等。

上面这些指标仅仅是评价文本的一些方面。即使在这些方面评分都很高，生成的文本也未必是非常好的。当然上述这些指标不是都能够让算法自动计算的。因此，如何能够自动评价文本的质量就成为一个课题。

下面讨论其中一个指标：用词多样性的自动计算问题。

用词多样性可以有几种自动度量方法。熵是其中的一种度量方法：

$$d = -\sum_{i=0}^{n} p_i \log p_i \tag{5-2}$$

式中，p_i 是第 i 个词出现的频率；n 是出现的词的总数。d 越大，说明文本的用词多样性越强。

例 5.3　使用熵度量用词多样性的计算。

句子“峨眉山的猴子在吃香蕉因为它很饿”用到了 10 个词：峨眉山/的/猴子/在/吃/香蕉/因为/它/很/饿，每个词只出现了一次，所以，所有词的词频是 1/10。因此，熵的计算为

$$d = -10 \times (0.1 \times \log 0.1) = \log 10 = 1$$

句子“峨眉山的猴子在吃香蕉”用到了 6 个词：峨眉山/的/猴子/在/吃/香蕉，每个词只出现了一次，所以，所有词的词频是 1/6。因此，熵的计算为

$$d = -6 \times (0.6 \times \log 0.6) = \log 6$$

可以看到，第二个句子用词少，这个指标也就更小。

有些指标的自动计算有困难时，也可以采用人工评判方法。而人工评判也存在一些问题，下面列举几个。

(1) 速度和花费。人工评判时需要请人阅读生成的文本并给出评价。这通常比较慢，花时间比较多。另外还需要给评判人员付费，所以需要的花费比较多。

(2) 可能会出错。人会因为疲劳或其他原因导致评判出错。

(3) 评判结果可能不一致。评判人员可能因为个人的认知、情感、水平差异而给出不一致的结果。这不仅发生在不同的评判人员之间，同一个评判人员的两次评判也可能不同。

类似文本生成的自动评判问题在图像的生成、音乐自动生成中也会出现。

5.13　基于深度学习方法的优缺点

1. 优点

深度学习方法采用了大量的数据对大模型进行训练，取得了很好的性能。其优点如下：

(1) 生成的句子非常流畅。

(2) 能够很好地对已有的训练数据总结、归纳和综合，生成高质量的文本。

(3) 短语之间的相似性在隐含空间能够很好地体现。

(4) 具有端到端训练方法的优点。

(5) 当涉及新的语言或者新的应用领域(如科技领域)时，只需要提供相应的数据，而不需要人工设计和提取特征。

2. 缺点

上面这些优点让自然语言处理与理解算法可以落地变为产品。特别是 ChatGPT 的出现，进一步推动了技术的发展和应用。当然，以 ChatGPT 为代表的深度学习系统也有如下缺点。

（1）可解释性比较差。

以机器翻译任务为例，在基于规则的翻译方法中，当出现错误的时候，可以根据被误翻的句子在系统中经过的“轨迹”，找到出错的“位置”，从而发现原因，纠正错误和改进系统。但是在神经网络机器翻译方法中，神经网络模型从功能上可以完成翻译任务，但是其内部结构非常复杂，和人的翻译过程并不对应。因此，无法很好地解释在翻译过程中发生了什么。

（2）缺少符号和逻辑的学习机制。

一些预训练大模型在对话系统中表现出了一些推理功能，但是这些推理是对于已有的训练数据“综合”和“插值”的结果，而不是根据推理机制得到的。因此它不能对新的推理任务实现“外推”，不能保证推理的正确性。其根本原因是模型缺少纯逻辑的推理和计算机制。而对于很多问题，特别是一些数学问题，不采用逻辑的方法是不可能很好地解决的。

（3）新词问题。

当出现了系统没有见过的新词时，系统无法理解该词。从技术角度看，这是可以理解的。社会在发展，也因此不断会出现一些新词；科学与技术的进步也会创造出新的知识，也因此会出现一些新的术语，这给自然语言处理与理解系统带来了挑战。

（4）训练语料和应用环境不一致带来的问题。

举一个例子，如果在训练一个中英翻译系统时使用的是大量的文学作品中的中英对应句子（训练语料），而想把这个翻译系统用于翻译中文的工程技术文献（应用环境）时，这个系统的性能就会比较差。这个系统学会了文学作品的词汇、风格、表达和翻译方法，但是工程技术领域文献的用词、表达习惯、语言风格和文学作品相去甚远。因此，这个系统的翻译就达不到很好的效果。与之类似，如果用工程技术领域的语料训练了一个翻译系统，并将其用于翻译诗歌，系统的表现也会比较差。因此就要求训练的数据和应用时的数据在统计上是一致的。对于这个问题，可以增加应用环境的语料来训练翻译系统，这样这个问题可以得到缓解。

（5）知识的获取和使用。

在理解语言时需要很多知识。如何获取知识、表示知识和使用知识是重要的问题。这在知识表示章节会讨论。在当前的预训练大模型中已经包含了一些知识，特别是一些常

识。而且从知识的使用方面已经有了很好的表现。进一步的问题是如何保证它获得的知识是正确的。除了自然语料外，还有一些知识是人们总结出来的、以符号形式表示的，如何在自然语言处理与理解系统中使用这些知识也是重要的课题。

(6) 成语、俗语的理解。

语言中的成语、俗语的理解和翻译也往往是困难的课题。例如系统会把“我一出火车站就不知道东南西北了”翻译成“As soon as I leave the railway station, I don't know the southeast and northwest”。随着时间推移，系统可以使用的数据会越来越多，这个问题也会逐步得到缓解。

(7) 每种语言都可能存在一些特殊的语言现象。

下面举一个中文的例子。中文中有一个修辞手法叫互文见义，这给机器翻译带来了一些困难。例如，系统会把“他穿的左一件右一件”翻译成“He's wearing one on the left and one on the right.”(ChatGPT 2023,3)。另外，如果有一种语言“他/她”不分，那么类似这样一句话“ta 是一个护士”要翻译成英文该怎么办？

5.14　自然语言处理与理解模型成功的原因与给我们的启示

通过对自然语言处理与理解的研究，人们对于智能、智能任务、人工智能技术的发展有了一些新的认识。

自然语言处理的不同任务对文本的理解程度的要求是不同的。如果只需要对一篇文章做一个大致的归类，如经济、艺术、工程等类别，只使用文章中用到的一组关键词就能够取得一个不错的性能。如图 5-10 所示，不需要读文章的内容，只看其中频繁出现的一组特异性的词，就能够大致了解文章的主题。这就是自然语言处理中曾经使用过的**词袋**(bag of words)模型和**主题模型**(topic model)的思想来源。当前的词云(word cloud)技术就是利用了这一思想。

图 5-10　根据一组频繁使用的特异性的词来判断文章的主题

ChatGPT 低层的注意力模块获得了各个词之间的相关关系，更新后的词向量就包含了它关注到的词向量的信息，因而更像是一个词组或者短语等。再向上的一些层的注意力模块获得的各个词之间的相关关系实际上是词组或者短语之间的关系。这样，在模型高层，注意力模块就能得到句子和句子之

间的关系、一组句子和一组句子之间的关系。因为 ChatGPT 允许输入一万多个词，网络结构有几十层，所以模型可以注意到从词到篇章各个层次之间的语义的复杂关系。

根据研究发现，当模型的大小达到一定程度，训练数据达到一定程度的时候，预训练语言大模型性能会有一个明显提升。人们猜测，自然语言对世界的描述、表达方式等可能存在一个“边界”。当模型比较大并且数据足够多的时候，模型能够达到这个边界。

对于自然语言处理与理解的任务，可以使用和人不同的方式来完成。以 ChatGPT 为代表的模型和人的认知系统存在差异，但是也能较好地完成翻译、摘要、对话、写作等任务。

ChatGPT 的一个模型能够完成所有的自然语言处理与理解任务，打破了以前各任务单独研究，彼此分离的状态。

ChatGPT 可以以隐含的方式表示大量知识，特别是常识，并能够以一种更自然的方式使用知识。这是对于人工智能的一个贡献。

ChatGPT 可以实现简单的不确定性推理和常识推理。这也是对人工智能的一个贡献。

理解自然语言，包括要理解说自然语言的人的文化、习惯和价值观。从图 5-11 可以知道，ChatGPT 在这方面性能还不够好。文字不只是对客观世界的描述，还代表人的文化、观点、情感、价值观等。

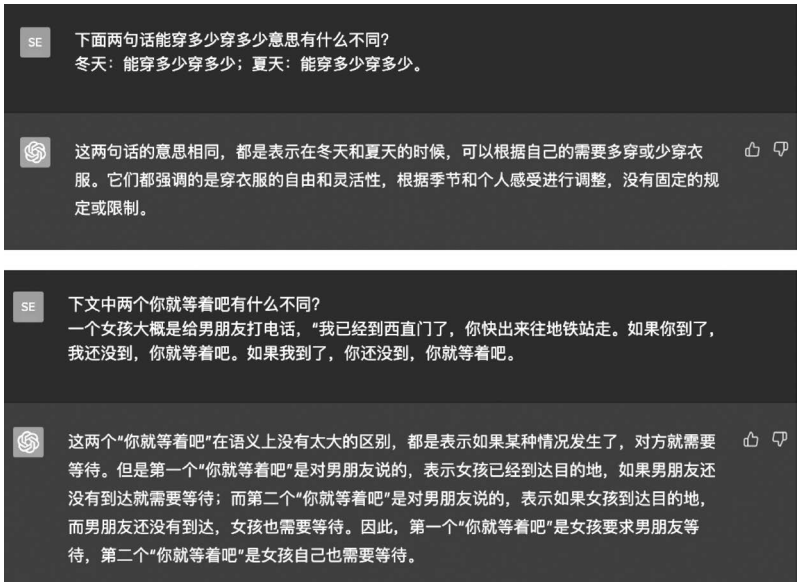

图 5-11　ChatGPT 对于一个问题的回答

理解自然语言，也包括理解语言和物理世界的对应。除了本章提到的机器翻译、对话和问答这样的任务外，人们还需要通过语言指挥机器人完成一定任务。如要求机器人到厨房做饭、在房间打扫。这些都需要机器人能够将自然语言指令和物理世界及机器人的操作相对应。而在这方面，只依靠语言模型是不够的。

5.15　语言的局限性

当前的预训练语言大模型在机器翻译、对话和问答、语言的生成方面取得的成果往往会给人一种错觉：这些大模型无所不知。特别是这些大模型使用了远远超过普通人所能够接触和阅读的语料量，超出了人所掌握的知识范围，这更会让人们产生这种错觉。而事实上，在智能这个问题上，语言本身具有局限性。仅仅通过语言训练的人工智能系统无法接近人类的智能。

1. 语言只承载了所有人类知识的一部分

人对于世界有很多的知识，其中只有一部分表现在语言上。很多知识不是用语言来表达的。下面举几个例子。

人们去九寨沟看到漂亮的自然风景会感叹“非常美”，看京剧演出会感叹“非常美”。但是，人对这两种美的感受是不一样的。但是在语言中并没有专门的词能够差异化地分别描述这两种情况。对于九寨沟，人们可能会用下面这些词来描述：“壮丽”“震撼”……但是这些词也只能描述感受的一部分，而不能描述人的完整的感受。为什么需要用一组词来描述？就是因为没有唯一的一个词能够准确地刻画这种感受。因此，只好使用一组词，从不同的角度描述。这只是说在人的感受方面。如果需要客观描述一张图，人们可能会用“碧蓝的湖水”“五彩的森林”等描述，或者用几百个字去很细致地描述。尽管如此，也仍然不能详尽准确地描绘一张图。这就是为什么“一图胜千言”。要想完整准确地描述一张图，最好就是用这张图，而不是用自然语言。

之所以会这样，根本原因在于语言是一个离散的符号系统。而人的感受或者对于图像的表达是连续的。要使用离散的符号系统表示一个连续空间范围，如果连续空间范围非常小、紧凑，并且有规律，就可以用一个单独的符号表示，这就是人们约定俗成的一些词汇。如果连续空间范围宽，没有规律，就无法用简单的一些符号准确表示。

对于图像是这样，对于声音、味觉、触觉也都有类似的问题存在。

2. 语言需要和物理世界结合

人们讨论的一个问题是这样的：假设有一个外星人到了地球，对地球一无所知。他阅读了用语言描绘地球世界的百科全书，是不是对地球无所不知了？

不是的。举个例子，虽然他读了书，知道红玫瑰、红牡丹是两种花。但是，如果不看到这两种花，他能区分出图 5-12 的两张图片吗？他能够理解玫瑰花的红色和牡丹花的红色的

差异吗？因此，要理解这个世界，仅仅用语言本身是不够的。换句话说，视觉、听觉、触觉、味觉和嗅觉都承载了信息，而这些是语言不能够替代的。

图 5-12　红玫瑰与红牡丹

3. 人对于语言的理解离不开客观世界

认知科学告诉我们，人们对于语言的理解需要通过在大脑中模拟语言描绘的内容。比如说“不要想像一只大象”，这件事人做不到。这是因为人在理解这句话的时候一定会有对大象的想像和模拟。没有想像和模拟，理解是完不成的。换句话说，人们对世界的理解从来都不是仅靠语言。

5.16　进一步学习的内容

有很多大学开设了自然语言处理方面的课程，系统地介绍自然语言处理的理论、方法等内容。也有很多自然语言处理方面的教材、书籍。另外，可以阅读一些文章了解自然语言处理方面的进展。

下列是自然语言处理方面很有影响力的会议。

- Annual Meeting of the Association for Computational Linguistics
- Conference on Empirical Methods in Natural Language Processing
- The Annual Conference of the North American Chapter of the Association for Computational Linguistics
- International Conference on Computational Linguistics

- Conference on Computational Natural Language Learning

下列是自然语言处理方面很有影响力的杂志。

- *Transactions of the Association for Computational Linguistics*

可以扫描二维码阅读关于课程、教材、文章等方面的信息。

进一步学习的内容

练习

1. 请计算下面两句话的用词多样性指标："我/喜欢/吃/苹果/吃/苹果/有利于/健康""我/喜欢/吃/苹果/吃/水果/有利于/健康"。解释这两句话在这个指标上的差异。

2. 请计算下面三句话的用词多样性指标："今天/很/热""今天/很/热/热/死/了""今天/很/热/热/死/了/热/死/了/热/死/了"，并解释这个指标在这三句话上的差异。

3. 考虑下面这句话："苹果/很/好吃"，假设$\boldsymbol{v}$（苹果）=[3,2]，$\boldsymbol{v}$（很）=[1,1]，$\boldsymbol{v}$（好吃）=[3,1]，请计算在向子空间[1,1] 投影后，"苹果"对于"很""好吃"的注意力权重。

4. 在练习 3 中，如果希望"苹果"对于"好吃"的注意力权重超过 0.7，可以找到合适的子空间吗？如何找到这个子空间？如果希望"苹果"对于"好吃"的注意力权重超过 0.99，可以找到合适的子空间吗？为什么？

5. 在自然语言处理问题中，常常需要一个单词库（vocabulary），也被称作字典。单词库可以是通过模型训练学习的，可以是人工输入的，也可以是用程序从数据集中自动抽取的。但是，在后续处理过程中，有可能遇到不在单词库中的单词，这被称作 OOV（out-of-vocabulary）问题。请问：应该如何处理 OOV 问题？为什么？

6. 自然语言处理有许多经典任务。例如，命名实体识别（named entity recognition，NER）就是一个重要的基本问题，自然语言处理的目标是识别文本中具有特定意义的实体，通常包括人名、地名、机构名、日期时间、专有名词等。自然语言处理中还有哪些经典问题？它们的定义是什么？可以用什么方法解决？

7. Transformer 由编码器和解码器组成。其中，编解码部分均采用了多层结构，如图 5-4(a)所示。在编码器中，数据首先经过自注意力层，得到加权向量 $\boldsymbol{z}$，如图 5-4(b)所示。对于每个单词，输入数据包含 3 个长度相等的向量：查询向量 $\boldsymbol{q}$，键向量 $\boldsymbol{k}$，值向量 $\boldsymbol{v}$，记它们共同的维度为 d_k。取 $\boldsymbol{q}_1$=[2,3,2,3]，$\boldsymbol{k}_1$=[1,4,1,4]，$\boldsymbol{k}_2$=[3,2,3,2]，$\boldsymbol{v}_1$=[1,1,1,1]，$\boldsymbol{v}_2$=[2,1,1,2]，易知 d_k=4，请补全图 5-7 中的①～⑨。

8. 有一个在线训练模型的网站：https://ai.baidu.com/easydl/。请利用这个网站对一个语言任务模型进行训练。可以扫描二维码阅读对这个网站的操作文件。

在线训练模拟网站