第 5 章 循环神经网络的原理和编程方法

深度学习网络在多个领域都展现出了强大的应用潜力，其中图像处理和自然语言处理是两个最为突出的领域。在图像处理领域，深度学习网络，特别是卷积神经网络，已经取得了显著的成功。通过学习和提取图像中的复杂特征，深度学习网络可以执行各种任务，如图像分类、目标检测和图像分割等。此外，生成对抗网络等模型还能够生成逼真的图像，进一步拓宽了深度学习网络在图像处理中的应用范围。在自然语言处理领域，深度学习网络也发挥了重要作用。循环神经网络能够处理序列数据，捕捉语言中的时序依赖关系，从而能够完成机器翻译、文本生成和情感分析等任务。近年来，随着基于 Transformer 的 BERT、ChatGPT 等模型的提出，深度学习网络在自然语言处理领域的应用得到了进一步的推动，实现了更高的性能和更广泛的应用。这些模型不仅可以理解文本的含义，还可以生成自然、流畅的文本，甚至参与到对话系统中，与人类进行交互。深度学习网络在自然语言处理领域中的应用，极大地推动了人工智能技术的发展，使机器能够更好地理解和处理人类语言。

本章介绍深度学习网络在自然语言处理领域的应用。首先，介绍循环神经网络的基本特点，包括序列数据的特点、循环神经网络的应用场合、简单循环神经网络的原理和特点、循环神经网络的其他结构等；然后，介绍词语嵌入编码的基本内容，包括语句的分词问题、词语的独热编码、词语的嵌入编码和编程方法；接着，介绍基于 Keras 深度学习框架的简单循环神经网络的编程方法；最后，介绍长短期记忆模型和门控循环单元网络的基本原理和特点，并介绍基于 Keras 深度学习框架的长短期记忆模型的编程方法。

5.1 循环神经网络

5.1.1 循环神经网络简介

深度学习网络可以分为 3 种类型：全连接多层前馈神经网络（也称为多层感知器）、卷积神经网络和循环神经网络。全连接多层前馈神经网络和卷积神经网络存在以下 3 个缺点。首先，这两种类型的网络只完成了信息的单向传递，网络的输出只和当前的输入有关系，和以前的输入没有关系，所以这两种类型的网络没有记忆功能。其次，在这两种类型的网络中，输入值之间相互独立，即当前的输入值和以前的输入值、未来的输入值没有关系。最后，这两种类型的网络不能处理序列数据。在进行数据的处理时，通常把沿着某一维存在很强相关性的数据称为序列数据，如聊天对话的序列数据、股票价格的序列数据等。“让风

尘刻画你的样子”这句话中的每个词之间都具有相关性，把它们联系起来作为一个整体才能够表达完整的含义，所以这句话的数据就是序列数据。

为了解决这两种类型网络的缺点，研究人员提出了循环神经网络。循环神经网络具有短期的记忆能力，这种能力通过隐藏层的神经元实现，这些神经元不仅接收当前时刻输入的信息，还接收来自它们自身上一时刻的输出信息。循环神经网络的这种机制使其能够捕捉序列数据中的依赖关系，特别是那些相距较近的单元之间的依赖关系。所以，循环神经网络能够处理文本、时间序列等序列数据，能够对序列中的每个单元分别进行处理，并在处理过程中考虑之前的信息。在进行文本处理时，循环神经网络把句子拆分成词语或单词的序列，并依次处理每个单元。例如，当循环神经网络处理“让风尘刻画你的样子”这句话时，这句话被分成了“让”“风尘”“刻画”“你的”“样子”这些基本单元，循环神经网络会依次读取这些单元，并在处理每个单元时考虑之前的信息。全连接多层前馈神经网络和卷积神经网络是静态的网络，没有考虑输入信息之间的时间依赖关系或序列结构。相比之下，循环神经网络是动态的网络，能够处理可变长度的输入序列，并通过隐藏状态来捕获序列数据中的依赖关系，这使得循环神经网络在处理时间序列、文本等具有序列结构的数据时更加有效。此外，循环神经网络的结构更接近生物神经网络的结构特点。生物神经网络中的神经元之间具有复杂的连接关系，并且神经元之间具有动态的通信方式，能够随着时间和经验的积累而发生变化。循环神经网络使用隐藏层进行状态的传递和更新，从而模拟了生物神经网络动态和连续的信息处理过程。

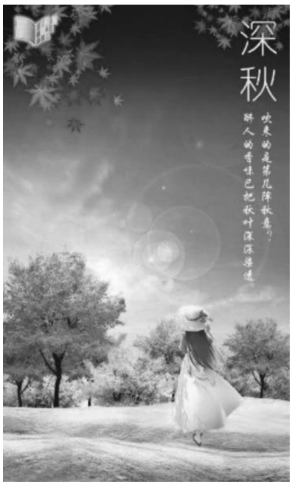
循环神经网络在很多领域得到了大量应用，如语音识别、通过人脸进行年龄的判断和自然语言处理等。循环神经网络在语音识别领域的应用非常广泛，它能够处理音频信号中的时序依赖关系，将连续的音频流转换为文字。例如，一个基于循环神经网络的语音识别系统可以接收一段语音输入，然后将其转换为对应的文本输出。通过训练大量的语音数据，循环神经网络可以学习到语音信号与文本之间的映射关系，从而实现准确的语音识别。

循环神经网络也可以根据人脸进行年龄的判断，它接收人脸的图像作为输入信号，并分析图像中面部的皱纹分布、皮肤松弛度、头发的灰白程度、面部的表情、眼神状态等特征，以预测人的年龄。在根据人脸判断年龄时，首先使用卷积神经网络提取特征，然后使用循环神经网络处理这些特征并输出年龄的预测值。例如，使用循环神经网络对图 5-1 中的人脸图像预测人的年龄，得到的年龄数值为 44 岁。

图 5-1　使用人脸图像预测年龄的示例

循环神经网络在自然语言处理领域有非常多的应用，如看图说话、摘要自动生成、机器翻译和自动作诗等。在处理看图说话问题时，循环神经网络接收一张图片作为输入值，并生成一段描述该图片的文本作为输出值。在处理这个问题时，通常首先使用卷积神经网络进行图像特征的提取，然后使用循环神经网络把这些特征转换为文本的描述。例如，根据图 5-2(a)中的图片看图说话，得到的文本如图 5-2(b)所示。

在处理摘要的自动生成问题时，循环神经网络接收一段较长的文字作为输入值，并生成一个简洁的摘要作为输出值。在处理这个问题时，循环神经网络首先分析文本中的关键信息和结构，然后提取重要的内容并生成摘要。例如，输入文本为：近年来，随着科技的快速



(a) 图片

在这张深秋的图片中，一位女性背对着我们，身着白色长裙和宽边帽，优雅地伫立在金黄色的草地上。她的长发随风飘动，似乎在诉说着秋天的故事。周围的树木被秋天的魔法点染得五彩斑斓，红叶和橙叶在阳光下闪烁着耀眼的光芒，与蔚蓝的天空形成了鲜明的对比。

阳光透过云层，洒在草地上，形成斑驳的光影。那些淡淡的云彩在天空中悠然漂浮，仿佛是秋天的使者，带来了一阵阵凉爽的秋意。在这宁静而美丽的场景中，我们仿佛可以闻到秋天特有的气息，那是醉人的香味，深深染透了每一片秋叶。

图片的上方，黑色的中文字迹“深秋”显得格外醒目，它不仅是对这幅画面最贴切的描述，也唤起了我们对秋天的无限遐想。我们不禁开始思考，这吹来的是第几阵秋意？是初秋的清新，还是深秋的浓郁？

这张图片以其独特的视角和细腻的描绘，成功捕捉了秋天的精髓和魅力。它让我们感受到了秋天的宁静与美丽，也让我们对大自然的力量和神奇产生了深深的敬畏和赞叹。

(b) 看图说话得到的文本

图 5-2　看图说话问题的示例

发展，人工智能技术在各个领域得到了广泛应用。在自然语言处理领域，人工智能技术已经实现了机器翻译、语音识别、情感分析等多种功能。此外，在医疗、金融、教育等领域，人工智能技术也发挥着越来越重要的作用。通过深度学习和大数据分析，人工智能系统能够识别并处理大量数据，为人类提供更加精准、高效的服务。然而，人工智能技术的发展也面临着一些挑战，如数据隐私、算法公平性和可解释性等问题。未来，随着技术的不断进步和完善，人工智能技术有望在更多领域发挥更大的作用。对此文本生成摘要“人工智能技术在自然语言处理、医疗、金融、教育等领域得到广泛应用，通过深度学习和大数据分析提供精准服务。然而，技术发展也面临数据隐私、算法公平性和可解释性等挑战。未来，人工智能有望在更多领域发挥更大作用。”

循环神经网络在机器翻译(Machine Translation，MT)中扮演着重要角色，它使用计算机将源语言的文本作为输入信号，并生成目标语言的对应文本作为输出信号。使用大量的双语语料库进行训练，循环神经网络能够学习到不同语言之间的翻译规则和模式。例如，对英语文本“The rapid development of artificial intelligence technology has led to its widespread application in various fields, including machine translation, speech recognition, and sentiment analysis.”进行机器翻译，得到的中文文本为“人工智能技术的快速发展导致了其在包括机器翻译、语音识别和情感分析在内的各个领域的广泛应用。”

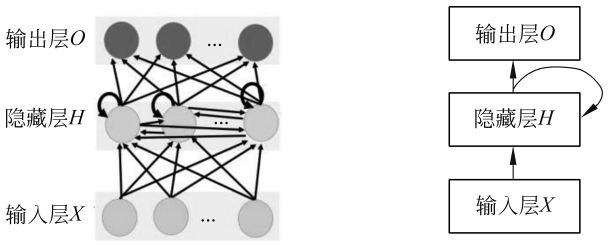
循环神经网络还能够用于自动作诗。通过训练大量的诗歌数据，循环神经网络能够学习到诗歌的韵律、格律和意境等要素，并根据给定的输入(如第一句、关键词或标题)生成完整的诗歌。例如，使用循环神经网络，生成一首以“人山人海”这 4 个字为藏头的诗歌：“人间烟火闹纷纭，山川湖海尽入眸，人海茫茫寻知己，海天一色共长流。”

总之，循环神经网络的应用范围还在不断地扩展和深化，为人工智能技术的发展提供了强大的支持。

5.1.2　简单循环神经网络的原理

循环神经网络有很多种类型，其中，简单循环神经网络(Simple Recurrent Neural

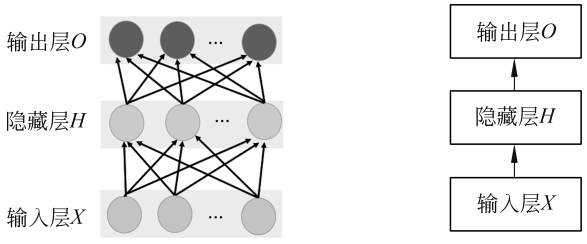
Network，SRNN)的结构相对比较简洁。SRNN只有一个隐藏层，其结构如图5-3(a)所示，图5-3(b)为图5-3(a)的简化图。具有一个隐藏层的全连接前馈神经网络的基本结构如图5-4(a)所示，图5-4(b)为图5-4(a)的简化结构。



(a) 结构图　　(b) 简化图

图5-3　简单循环神经网络的结构

这两种类型的神经网络具有相同的基本结构：输入层、一个隐藏层和输出层。但是，这两种类型的神经网络有明显的区别。在全连接前馈神经网络的信号流动方面，信号从输入层流向隐藏层，然后从隐藏层流向输出层；在层与层之间，神经元是全连接的方式；在隐藏层的内部，神经元之间则没有连接关系。在SRNN的信号流动方面，隐藏层的神经元不仅接收来自输入层的信号，还接收来自自身上一个时刻的信号。



(a) 结构图　　(b) 简化图

图5-4　全连接前馈神经网络的结构

在全连接前馈神经网络中，某个时刻输出层神经元的输出信号只和当前时刻输入层中的输入信号有关系，和以前时刻输入层的输入信号没有关系。所以，全连接前馈神经网络主要用于处理输入信号和输出信号之间互相独立的问题，即输出信号只依赖于当前的输入信号，而不受历史输入信号的影响。和全连接前馈神经网络相比，SRNN的运行机制更加复杂一些。在SRNN中，某个时刻输出层神经元的输出信号不仅仅和当前时刻输入层中的输入信号有关系，而且和以前时刻输入层神经元的输入信号有关系。所以，循环神经网络特别适用于处理具有时间依赖性的序列数据，如自然语言处理中的文本序列数据或时间序列数据，因为每个时刻的输出信号和之前所有时刻的输入信号都有关系。通过保存和传递内部的状态，循环神经网络能够捕捉序列数据中的长期依赖关系。也就是说，循环神经网络能够处理具有时间依赖性的数据，而全连接前馈神经网络则没有这个能力。循环神经网络使用隐藏层的自循环结构，能够记住并利用历史信息来影响当前的输出信号。在计算复杂度方面，由于循环神经网络需要处理序列数据，其计算通常比全连接前馈神经网络更复杂，尤其是在处理长序列数据时可能会遇到梯度消失或梯度爆炸的问题。

下面介绍SRNN的工作原理。按照序列数据中不同数据的先后顺序，使用SRNN处理

序列数据的基本过程如图 5-5 所示。根据此图，可以看出隐藏层中的神经元在 t 时刻的输出信号 H_t 不仅仅和 t 时刻输入层中输入信号 X_t 有关系，而且和 $t-1$ 时刻隐藏层神经元的输出信号 H_{t-1} 有关系。

在图 5-5 中，假设输入层 X 有 M 个神经元，则输入层有 M 个输入信号 $x_0 \sim x_{M-1}$，隐藏层有 N 个神经元，每个神经元的输出信号为 $h_0 \sim h_{N-1}$，输出层有 K 个神经元，每个神经元的输出信号为 $o_0 \sim o_{K-1}$，如图 5-6 所示。

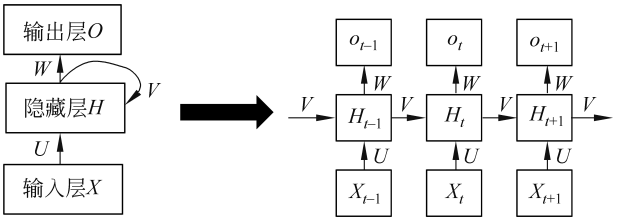


图 5-5 SRNN 处理序列数据的过程

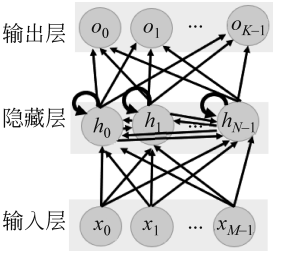


图 5-6 SRNN 中的神经元

在图 5-6 中，隐藏层神经元的输出值为

$$h_{t,j} = f\left(\sum_{i=0}^{M-1} x_{t,i}u_{i,j} + \sum_{i=0}^{N-1} h_{t-1,i}v_{i,j} + b_j\right) \tag{5-1}$$

其中，$h_{t,j}$ 表示 t 时刻隐藏层中第 j 个神经元的输出值，f 表示激活函数，M 表示输入层中输入信号的数量，$x_{t,i}$ 表示 t 时刻输入层的第 i 个输入信号，$u_{i,j}$ 表示输入层中第 i 个神经元到隐藏层第 j 个神经元的权值系数，N 表示隐藏层中神经元的数量，$h_{t-1,i}$ 表示 $t-1$ 时刻隐藏层中第 i 个神经元的输出值，$v_{i,j}$ 表示隐藏层中第 i 个神经元到隐藏层中第 j 个神经元的权值系数，b_j 表示隐藏层中第 j 个神经元的偏置系数。

把式(5-1)使用矩阵的形式表示，则隐藏层神经元的输出值为

$$\boldsymbol{H}_t = f(\boldsymbol{X}_t\boldsymbol{U} + \boldsymbol{H}_{t-1}\boldsymbol{V} + \boldsymbol{B}) \tag{5-2}$$

其中，$\boldsymbol{H}_t$ 和 $\boldsymbol{H}_{t-1}$ 分别表示 t 时刻、$t-1$ 时刻隐藏层所有神经元输出值组成的向量，它们的形状都是 $1\times N$，它们的表达式为

$$\boldsymbol{H}_t = [h_{t,0}\ h_{t,1}\ \cdots\ h_{t,N-1}] \tag{5-3}$$

$$\boldsymbol{H}_{t-1} = [h_{t-1,0}\ h_{t-1,1}\ \cdots\ h_{t-1,N-1}] \tag{5-4}$$

其中，$\boldsymbol{X}_t$ 表示 t 时刻输入层中 M 个输入信号组成的向量，其形状为 $1\times M$，它的表达式为

$$\boldsymbol{X}_t = [x_{t,0}\ x_{t,1}\ \cdots\ x_{t,M-1}] \tag{5-5}$$

$\boldsymbol{U}$ 表示输入层中的每个神经元到隐藏层中的每个神经元的权值系数矩阵，其形状为 $M\times N$，它的表达式为

$$\boldsymbol{U} = \begin{pmatrix} u_{0,0} & \cdots & u_{0,N-1} \\ \vdots & \ddots & \vdots \\ u_{M-1,0} & \cdots & u_{M-1,N-1} \end{pmatrix} \tag{5-6}$$

$\boldsymbol{V}$ 表示隐藏层中的全部神经元到自身的权值系数矩阵，其形状为 $N\times N$，它的表达式为

$$\boldsymbol{V} = \begin{pmatrix} v_{0,0} & \cdots & v_{0,N-1} \\ \vdots & \ddots & \vdots \\ v_{N-1,0} & \cdots & v_{N-1,N-1} \end{pmatrix} \tag{5-7}$$

$\boldsymbol{B}$ 表示隐藏层中每个神经元的偏置系数组成的向量，其形状为 $1\times N$，它的表达式为

$$\boldsymbol{B}=[b_0\ b_1\ \cdots\ b_{N-1}] \tag{5-8}$$

在图 5-6 中，输出层神经元的输出值为

$$o_{t,j}=f\left(\sum_{i=0}^{N-1}h_{t,i}w_{i,j}+c_j\right) \tag{5-9}$$

其中，$o_{t,j}$ 表示 t 时刻输出层中第 j 个神经元的输出值，f 表示激活函数，N 表示隐藏层中神经元的数量，$h_{t,i}$ 表示 t 时刻隐藏层中第 i 个神经元的输出值，$w_{i,j}$ 表示隐藏层中第 i 个神经元到输出层中第 j 个神经元的权值系数，c_j 表示输出层中第 j 个神经元的偏置系数。

把式(5-9)用矩阵的形式表示，则输出层神经元的输出值为

$$\boldsymbol{O}_t=\boldsymbol{H}_t\boldsymbol{W}+\boldsymbol{C} \tag{5-10}$$

其中，$\boldsymbol{O}_t$ 表示 t 时刻输出层中所有神经元输出值组成的向量，其形状为 $1\times K$，它的表达式为

$$\boldsymbol{O}_t=[o_{t,0}\ o_{t,1}\ \cdots\ o_{t,K-1}] \tag{5-11}$$

$\boldsymbol{H}_t$ 表示 t 时刻隐藏层所有神经元输出值组成的向量，其形状为 $1\times N$，它的表达式如式(5-3)所示。

$\boldsymbol{W}$ 表示隐藏层中的神经元到输出层神经元的偏置系数组成的矩阵，其形状为 $N\times K$，它的表达式为

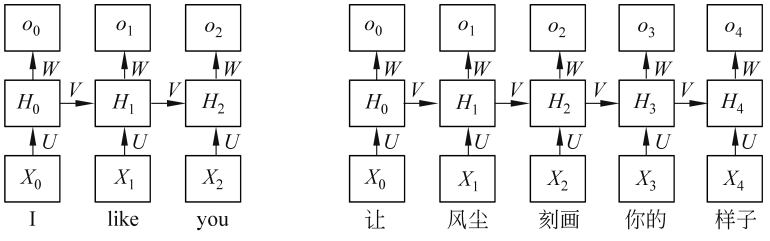
$$\boldsymbol{W}=\begin{pmatrix} w_{0,0} & \cdots & w_{0,K-1} \\ \vdots & \ddots & \vdots \\ w_{N-1,0} & \cdots & w_{N-1,K-1} \end{pmatrix} \tag{5-12}$$

$\boldsymbol{C}$ 表示输出层中神经元的偏置系数组成的向量，其形状为 $1\times K$，它的表达式为

$$\boldsymbol{C}=[c_0\ c_1\ \cdots\ c_{K-1}] \tag{5-13}$$

SRNN 具有参数共享的特点。SRNN 的隐藏层和输出层在不同时刻具有相同的网络参数，也就是说，当按照序列数据中不同数据的先后顺序输入每个数据时，SRNN 的权值系数矩阵 $\boldsymbol{U}$、$\boldsymbol{V}$、$\boldsymbol{W}$ 和偏置系数向量 $\boldsymbol{B}$、$\boldsymbol{C}$ 的值都是不变的。这种特点能够减少 SRNN 的网络参数数量，并且能够提高此网络模型的泛化能力。循环神经网络和卷积神经网络类似，使用梯度下降法或其改进方法训练网络参数，从而获得网络最优的预测性能。

下面介绍使用 SRNN 输入序列数据的示例。在第一个示例中，序列数据为“I like you”，分别给 SRNN 输入“I”“like”“you”的编码值，如图 5-7(a)所示。在第二个示例中，把歌曲《你的样子》的一句歌词“让风尘刻画你的样子”送入 SRNN，如图 5-7(b)所示。首先，对这句歌词进行分词处理，得到 5 个单元：“让”“风尘”“刻画”“你的”“样子”；然后，按照数据的输入顺序，分别输入“让”“风尘”“刻画”“你的”“样子”的编码值。



(a) 第一个示例　　(b) 第二个示例

图 5-7　简单循序神经网络处理序列数据的示例

5.1.3 循环神经网络的其他结构

除了 SRNN 之外，循环神经网络还有很多其他类型，如多输入单输出的循环神经网络、带有反馈的循环神经网络。每种类型的结构都不同，适合应用的场合也不同。

多输入单输出循环神经网络的结构如图 5-8 所示，这种网络只在最后一个时刻获得输出信号，在前面的时刻不会产生任何输出信号。在这种网络中，多个输入信号送入网络的内部单元，并更新网络的内部状态，最后才产生输出信号。多输入单输出循环神经网络适合处理序列数据到单一值的映射问题、文本情感分析问题等。例如，根据前 5 天的平均温度值预测今天的平均温度值，判断一句话是否有语法错误等，都属于序列数据到单一值的映射问题。在文本情感分析问题中，首先把文本的整体内容作为输入信号，然后获得单一的情感标签值，如热情、冷漠和平淡等。

带有反馈的循环神经网络的结构如图 5-9 所示。在每个时刻，这种网络都会产生输出信号，并把输出信号反馈到隐藏层。这种类型的网络能够处理序列数据到序列数据的映射问题，如机器翻译问题和文本生成问题等。在序列到序列的映射问题中，输入信号和输出信号都是序列数据。在机器翻译问题中，一种语言翻译成另外一种语言，把一种语言的句子作为输入序列，并在每个时刻输出另一种语言的部分词语，直到生成完整的句子。例如，把“I gave a talk in Beijing”翻译成“我在北京做了一个报告”。在文本生成问题中，根据给定的上下文生成新文本。首先，仔细阅读给定的上下文信息，并根据任务的要求确定要生成的文本类型（如产品推广文案、新闻稿等）和目标读者群体；然后，根据文本类型和目标读者的需求，构思文本的内容框架和要点，并按照构思的内容框架和要点撰写文本；接着，仔细检查文本，确保没有错误和疏漏，并根据需要进行修订和完善。使用以上步骤，根据给定的上下文就能够生成高质量、有吸引力的新文本。

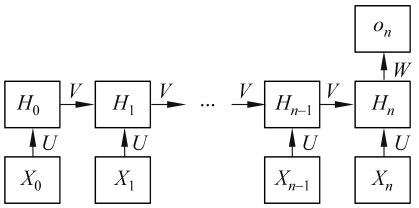


图 5-8 多输入单输出循环神经网络的结构

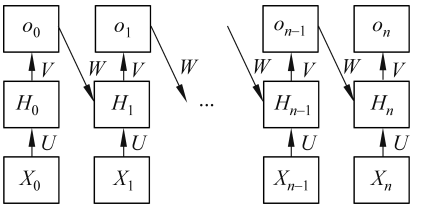


图 5-9 带有反馈的循环神经网络的结构

5.2 词语嵌入编码的原理和编程方法

5.2.1 语句的分词问题

语句的分词问题指把一句话分成最小的多个单元，也就是把一句话分解成多个孤立的词语。在使用英语作为语言的语句中，语句的分词问题比较简单，每个单词就是一个词语。例如，“I like you”这个英语语句包括 3 个单词，这句话的分词结果为“I / like / you”。在使用汉语作为语言的语句中，语句的分词问题相对比较复杂。对使用汉语作为语言的语句进行分词，得到的每个词语包括一个或多个汉字。汉语的分词可以分为两类：粗粒度分词和

细粒度分词，分词的粒度可以根据具体任务的需求选择。例如，对于“我路过南京市长江大桥”这句话，如果进行粗粒度的分词，分词结果为“我 / 路过 / 南京市 / 长江大桥”；如果进行细粒度的分词，分词结果为“我 / 路过/ 南京 / 市 / 长江 / 大桥”。

汉语语句的分词问题是汉语语言处理系统的基础，它在自然语言处理领域中有着重要的意义。首先，汉语语句的分词工作是自然语言处理领域中最基本和最底层的任务，它是文本分析、信息抽取、情感分析、机器翻译、自然语言理解等高级自然语言处理任务的前提。如果没有准确的分词结果，这些高级任务很难取得理想的效果。然后，汉语语句的分词工作能够消除歧义。在汉语语句中，汉字和汉字之间没有明显的空格等分隔符，会产生很多歧义。例如，“南京市长江大桥”如果不进行分词处理，可能会被误解为“南京市长 / 江大桥”，而实际上是指“南京市 / 长江大桥”。因此，中文分词有助于消除这种歧义，准确理解文本的含义。其次，在信息检索系统中，用户输入的查询词往往需要进行分词处理，以便与文档中的词汇进行匹配。准确的分词结果可以提高检索的准确率和召回率，使用户能够更快地找到所需要的信息。最后，汉语语句的分词问题能够促进语言的理解。分词工作把连续的字符序列切分成有意义的词汇单元，这对于机器理解文本内容具有重要意义。通过分词，机器可以识别出文本中的关键词、短语和句子结构，从而更好地理解文本的主题、情感和意图。随着深度学习等技术的不断发展，汉语语句的分词技术也在不断进步。准确的分词结果可以为其他人工智能任务提供高质量的数据支持，推动人工智能技术的整体进步。

汉语语句的分词问题存在两个挑战：歧义切分和未登录词语的识别。歧义切分指一个文本序列切分成多种不同但同样合理的词语序列，其原因是汉语语句中的词语可能存在不明显的边界。例如，对“门把手弄坏了”这句话进行分词处理，能够得到两个完全不同的分词结果：“门把手 / 弄 / 坏 / 了”和“门 / 把 / 手 / 弄 / 坏 / 了”。虽然这两个分词结果的意义完全不相同，但是它们在语义上都非常合理。未登录词语指在分词系统的词典中不存在的词语，它们包括中外人名、中国地名、机构组织名、事件名、货币名、缩略语、派生词、各种专业术语和不断发展的一些新词语。出现未登录词语的原因很多，包括词典的不完整性、语言的动态性（如新出现的词语）、特定的文本领域（如专业术语）等。例如，“PMI 互信息”是一个未登录词，因为它可能不存在于一般的分词系统词典。

汉语语句的分词方法有以下的 3 个类别。第一类是基于规则的分词方法，其基本思想是按照一定策略，依靠预定义的而且词语数量非常大的词典和一定的策略，将待分词的文本与词典中的词条进行匹配，从而实现分词。这类方法简单易行，只需要定义好词典和匹配策略，就可以进行分词。但是，对于复杂的文本，特别是存在歧义的情况，这类方法很难使用简单的规则来准确分词。在这类方法中，代表性的方法包括最大匹配方法、逆向最大匹配方法、双向最佳匹配方法和逐词遍历方法。在最大匹配方法中，从待分词文本的一端开始，取最大长度的字符串与词典中的词条进行匹配，若匹配成功则分词；否则，缩短长度继续匹配。在逆向最大匹配方法中，与最大匹配方法相反，从待分词文本的另一端开始匹配。在双向最佳匹配方法中，同时从文本的两端开始匹配，选择两端匹配长度之和最大的分词结果。在逐词遍历方法中，将待分词文本中所有可能的子串与词典进行匹配，找出所有可能的分词结果。

第二类方法是基于统计的分词方法。在这类方法中，上下文中相邻的字同时出现的次数越多，就越有可能构成一个词，使用相邻字的出现概率来评估它们成词的可信度。这类方

法需要非常多的语言知识和信息，并且需要大量的标注语料来训练统计模型。此外，这类方法使用了统计信息，能够更好地处理歧义。在这类方法中，代表性的方法包括 N 元文法模型方法、隐马尔可夫模型方法、最大熵模型方法和条件随机场模型方法等。N 元文法模型方法首先统计文本中连续 N 个字的出现频率，然后使用此概率评估其成词的可能性。隐马尔可夫模型方法将分词的过程看作一个隐马尔可夫过程，通过训练得到模型参数，并利用模型进行分词。最大熵模型方法利用最大熵原理建模，可以灵活地结合多种特征进行分词。条件随机场模型方法计算整个标签序列的联合概率分布，从而得到最可能的分词结果。

第三类方法是基于理解的分词方法，这类方法在分词的同时进行句法分析和语义分析，并利用句法信息和语义信息处理歧义现象，从而让计算机模拟人对句子的理解来进行分词。这类分词方法的效果依赖于训练语料的规模和质量，并且需要大规模的标注语料来训练复杂的分词模型。这类方法由于结合了句法、语义等深层信息，因此能够更加准确地处理分词中的歧义。在这类方法中，代表性的方法包括专家系统分词方法和神经网络分词方法。专家系统分词方法使用人工定义规则和专家知识进行分词。在神经网络分词方法中，首先使用神经网络模型来自动学习文本中的特征，然后使用特征进行分词。

在自然语言处理领域中，汉语语句的著名分词模型如表 5-1 所示。

表 5-1 汉语语句的著名分词模型

名　　称	网　　址
IKAnalyzer	http://www.oschina.net/p/ikanalyzer
SCWS 中文分词	http://www.xunsearch.com/scws/docs.php
jieba 分词	https://github.com/fxsjy/jieba
新浪云	http://www.sinacloud.com/doc/sae/python/segment.html
语言云	http://www.ltp-cloud.com/document

5.2.2 词语嵌入编码的原理

计算机处理的基本单元是数字，所以计算机不能直接处理词语，所以必须先对词语进行编码，即把词语使用数字表示。在词语的编码中，每个词语使用唯一的向量表示，不同词语的向量不同。词语的编码方法通常有两种：独热编码（One Hot Encoding）和嵌入编码（Embedding Encoding）。

1. 独热编码

对于词典中的每个词语，独热编码会创建一个向量，该向量的长度等于词典中词语的数量。在独热编码向量中，只有一个元素为 1，其余的元素都为 0。每个向量中的 1 都表示对应的词语在词典中的位置。例如，某中文词典只包括 5 个词语：让、风尘、刻画、你的、样子。这 5 个词语的索引值分别是 0、1、2、3 和 4，那么这 5 个词语的独热编码向量如表 5-2 所示，词语索引值所在位置的元素为 1，其他元素都为 0。例如，“你的”这个词语的索引值为 3，在其独热编码向量[0 0 0 1 0]中，索引值 3 所在位置的元素为 1，其他元素都为 0。再如，某英语词典中只有 5 个单词：he、she、it、cat 和 dog，这 5 个词的索引值分别是 0、1、2、3 和 4，它们的独热编码向量如表 5-3 所示。

表 5-2　独热编码的第一个示例

词　　语	让	风尘	刻画	你的	样子
独热编码向量	[1 0 0 0 0]	[0 1 0 0 0]	[0 0 1 0 0]	[0 0 0 1 0]	[0 0 0 0 1]

表 5-3　独热编码的第二个示例

词　　语	he	she	it	cat	dog
独热编码向量	[1 0 0 0 0]	[0 1 0 0 0]	[0 0 1 0 0]	[0 0 0 1 0]	[0 0 0 0 1]

词语的独热编码方法比较简单，但是这种方法有两个缺点。首先，在很多自然语言处理问题中，独热编码向量会变得非常稀疏，而且其维度通常非常大。在表 5-2 和表 5-3 这两个示例中，词典包含的词语数量比较少，由于独热编码向量中元素的数量等于词语的数量，因此独特编码向量的维度比较小。但是，在很多自然语言处理问题中，词典中词语的数量非常多，经常过万，此时独热编码向量中元素的数量也会非常大。也就是说，当词典中词语的数量很大时，独特编码向量的维度会非常高，从而导致“维度灾难”问题，带来非常大的计算量，使计算变得非常低效。例如，如果词典中有 10 000 个词语，那么每个词语的独热编码向量中就会有 10 000 个元素，此时独热编码向量的维度值为 10 000。在这种独热编码向量的 10 000 个元素中，只有一个元素为 1，其他元素都为 0。在这个词典中，如果“风尘”这个词语的索引值为 1，那么这个词语的独热编码值为“[0 1 0 … 0]”。在编码值“[0 1 0 … 0]”中，元素“1”的前面和后面分别有 1 个元素“0”和 9998 个元素“0”。

其次，独热编码无法描述词语和词语之间的关系，即词语之间的相似性。具体来说，独热编码会为每个唯一的词语创建一个维度，并在该维度上放置一个 1，其目的是表示该词语出现在该维度；而在其他所有维度上放置 0，其目的是表示其他词语没有出现在其他所有的维度上。这种编码方式将每个词语视为一个完全独立的实体，忽略了它们之间可能存在的任何关系或相似性。例如，如果词典只有 3 个词语：cat、dog 和 duck，cat、dog 和 duck 分别被编码为[1 0 0]、[0 1 0]和[0 0 1]。然而，这种编码方式无法表示出 cat、dog 和 duck 都具有动物属性这种特点，也无法表示它们之间的相似性。

2. 嵌入编码

为了克服独热编码的缺点，研究人员提出了嵌入编码。在嵌入编码中，把维数由所有词数量的高维空间转换到维数低得多的连续空间中，词典中每个词语映射到实数域中唯一的一个向量。与独热编码不同，嵌入编码不是简单地将每个词语使用一个由 0 和 1 组成的稀疏向量表示，而是将每个词语表示为一个密集向量，通过训练能够得到这些密集向量。嵌入编码向量的维度通常远远小于独热编码向量的维度，可以大大减少模型需要处理的特征数量，同时保留了足够的信息。词语的嵌入编码已经被广泛用于各种自然语言处理任务，如文本分类、情感分析和命名实体识别等。

与独热编码相比，词语的嵌入编码向量能够捕捉词语之间的相似性，并且它们的维度通常要小得多(如 100～300 维)。通常使用词语的相关性衡量词语之间的相似性。在两个语句中，如果把两个词语的位置互换，互换后的语句仍然是正常的语句，就认为这两个词语具有很强的相似性。例如，在两个语句“宿舍的环境优美”和“公寓楼的环境舒适”中，“宿舍”和“公寓楼”能够互换，“优美”和“舒适”能够互换，所以“宿舍”和“公寓楼”这两个词语具有很强