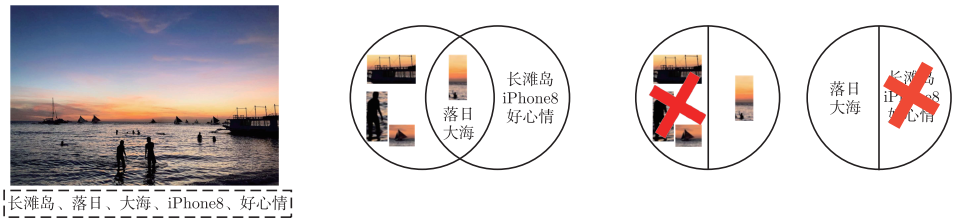
第 5 章　多模态表示

前面两章分别介绍了常用的文本和图像的单模态表示。为了完成多模态信息处理任务，需要在单模态表示的基础上学习多模态表示。在开始介绍多模态表示之前，先简单分析一下多模态数据的特点。通常，不同模态的数据既包含公共部分，也包含各个模态特有的部分。如图 5.1（a）所示的图文多模态数据：一幅图像及其文本描述。显然，“落日”和“大海”两个概念既能在图像中看到，也出现在文本描述中，是图像和文本两个模态的数据都包含的共同部分；而文本描述中的“长滩岛”“iPhone8”“好心情”无法直接从图像中获取，是文本模态特有的信息；“人”和“帆船”只能在图像中看到，无法从文本中获取，它们是图像模态特有的部分。图 5.1（b）利用文氏图对上述说明给出一个基于集合的描述。左边的圆代表图像信息的集合，右边的圆代表文本信息的集合，二者的交叉部分即两个模态信息的公共部分。



（a）图文多模态数据：图像和标签　　（b）共享层策略　　（c）对应层策略

图 5.1　多模态数据的特点以及多模态表示学习的两种策略

针对多模态数据的这一特点，研究者主要采取两种策略来学习多模态数据的表示。

第一种策略是为多个模态的数据学习一个共享的表示，多个模态的数据融合得到共享表示层，我们将该策略学习到的多模态表示称为**共享表示**（joint representation）。采用该策略的模型的目标是学习一个能够和图 5.1（b）中的文氏图对应的表示层，即表示层既包含图像和文本特有的信息，也包含公共部分信息。

2011 年，若干基于深度自编码器的多模态共享表示学习模型[102] 被提出，在视听语音分类任务上取得了当时的最优性能，首次将深度学习方法成功用于多模态信息处理任务。紧接着，2012 年，当时流行的两个单模态表示学习模型，即深度信念网络和深度玻尔兹曼机，被扩展成多模态共享表示学习模型[103-104]，并在图像标注和图文分类等任务中进行了评测。这些模型仅使用多模态对齐数据，以能够无监督地学习通用而强大的多模态表示为目标，期望结合简单的模型就可以完成多种任务。但是，由于各个模态均采用整体表示，且多模态表示融合方式过于简单，最终无法获得比为特定任务设计的模型更好的性能。

之后，大多数的多模态研究都转变成为特定多模态任务设计深度学习模型。这些模型以有监督的方式学习针对特定任务的多模态表示，并不以学习到通用的多模态表示为目标。因此，多模态表示仅是这些模型的附属品，更多的时候是扮演解释模型的角色。但是，在这期间，多模态对齐、融合和转换技术都取得了巨大进展。加上预训练语言模型的研究在自然语言处理领域的突破，2019 年开始，研究人员陆续提出多个多模态预训练模型学习通用的多模态表示。这些模型大多利用 transformer 融合多个模态的数据，<CLS> 符号对应的输出表示即可作为多模态共享表示。

第二种策略为每个模态数据单独学习相应的表示，但是在不同模态的数据的表示空间中增加相似性约束以建立多模态数据间的对应关联，我们将该策略学习到的多模态表示称为**对应表示**（coordinated representation）。采用该策略的模型的目标是学习一个能够和图 5.1（c）所示对应的表示层，即表示层只包含图像和文本数据的公共部分。

和共享表示学习模型一样，早期的对应表示学习模型同样是基于自编码器和玻尔兹曼机的。比如基于自编码器的 Corr-AEs[3]、MSAE[105]、DCCAE[106]、基于受限玻尔兹曼机的 Corr-RBMs[107]。由于可以直接将不同模态对应表示之间的距离视为多个模态数据的关联度，因此这些模型大多也直接用于跨模态检索任务。

之后，基于排序损失的方法采用了更贴近跨模态检索任务目标的损失函数，即通过引入不匹配的图像和文本作为负例，使得匹配的图文数据对之间的相似度大于不匹配的图文数据对之间的相似度，成为最主流的跨模态检索方法。使用这种损失的模型有 MNLM[108]、GXN[109]、VSE++[110]。

而一些研究人员通过可视化技术发现基于排序损失方法学习到的对应表示空间中的图像和文本是分离的，于是利用对抗学习的思想，提出基于对抗损失的方法。该类方法通过引入模态分类器使得图文数据在对应表示空间中能够充分融合，消除不同模态的差异。使用该损失的模型有 UCAL[111]、ACMR[112]。

上述对应表示学习模型都是针对跨模态检索任务而构建的，所习得的表示并不具有通

用性。2021 年，Radford 等提出基于排序损失的对应表示学习多模态预训练模型 CLIP[70]，并在由 4 亿条图文对组成的数据集上完成训练，其所学图像模态对应表示在零样本、小样本和常规设定下的图像分类任务上都取得了极其优异的性能。之后，CLIP 模型习得的对应表示广泛应用于各种多模态模型中，比如用于文本视频跨模态检索任务的 Clip4clip[113]、图像描述任务的 ClipCap[114]、用于文本引导图像编辑任务的 Styleclip[115]、文本生成图像任务的 DALL·E[72] 和 unCLIP[116]。

本章将介绍这两种多模态表示。需要说明的是，本章仅介绍早期的无监督的基于整体表示的多模态表示学习技术，多模态预训练技术将在之后的章节单独介绍。

5.1　共享表示

最直接的获取共享表示的方法是简单地拼接多个模态的表示，这样就可以获得多个模态数据的全部信息。然而，拼接的共享表示没有去除冗余信息，表达效率低。为此，深度学习方法一般使用如图 5.2 所示的网络结构学习共享表示，即先使用若干网络层对每个模态的输入分别建模，获取每个模态的抽象表示，然后使用一个网络层连接所有模态的抽象表示，获得所有模态的共享表示。这样的表示能够较为充分地融合各个模态的信息，表达较为紧凑。其可直接用于分类任务或作为下一个针对特定任务的神经网络模型的输入，可以被端到端地训练。

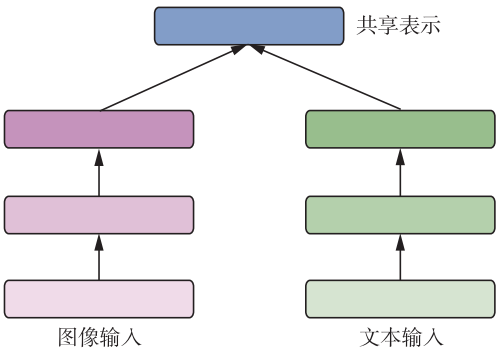


图 5.2　共享表示网络结构示意图

下面将介绍两类经典的共享表示学习模型：多模态深度自编码器和多模态深度生成模型。

5.1.1 多模态深度自编码器

如图 5.3 所示，多模态深度自编码器 (multimodal deep autoencoder, MDAE)[102] 的输入和输出均包含两个模态的数据。MDAE 的训练需要重新构造训练数据。除了对齐的图像文本训练数据，还需要增加两组训练数据：一组是仅有图像输入，文本输入全部置 0；另一组是仅有文本输入，图像输入全部置 0。但是，这两组数据也需要多模态自编码器重构图像和文本两个模态的数据。这样，三分之一的训练数据输入仅包含图像数据，三分之一的训练数据输入仅包含文本数据，另外三分之一的训练数据输入包含对齐的图像和文本数据。这里借鉴了去噪自编码器的思想，即要求从损坏的输入中重构完整输入，以学习更加鲁棒的表示。

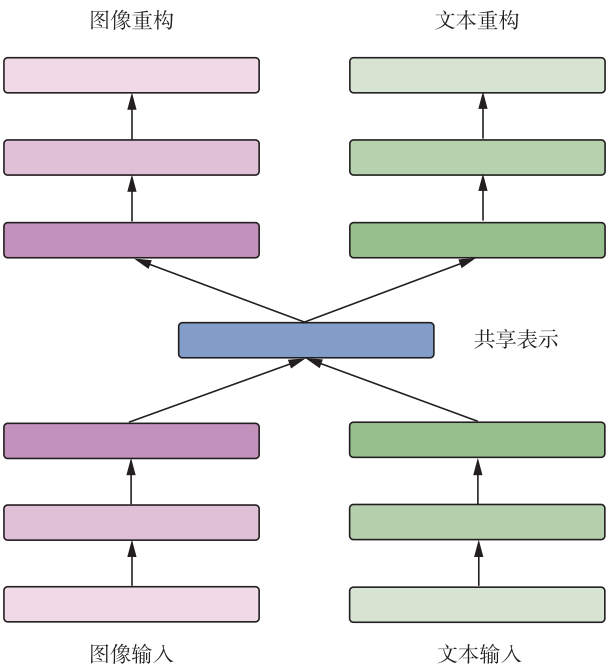


图 5.3　多模态深度自编码器模型结构示意图

在训练阶段，该模型可以可选地使用受限玻尔兹曼机（稍后介绍）或自编码器对每个层次进行预训练，再使用标准的反向传播算法训练。训练完成后，两个模态的数据就可以被多模态自编码器的编码模块映射到一个共享层中。

5.1.2　多模态深度生成模型

多模态深度信念网络[103] 和多模态深度玻尔兹曼机[104] 是两个典型的学习图文共享表示的多模态深度生成模型。二者都是以受限玻尔兹曼机为基础构建。为此，首先介绍受限玻尔兹曼机的结构、优化算法和评估方法，然后介绍以受限玻尔兹曼机为基础构建的两个单模态深度生成模型（深度信念网络和深度玻尔兹曼机），最后介绍相应的多模态生成模型。

1. 受限玻尔兹曼机

受限玻尔兹曼机 (restricted Boltzmann machine, RBM)[117] 是一个基于能量的模型 (energy based model, EBM)，它是玻尔兹曼机 (Boltzmann machine, BM)[118] 的一种特殊形式。

如图 5.4 所示，RBM 是一个两层的概率无向图模型，它由一个输入层和一个表示层组成。与 BM 中所有节点之间都有连接不同，RBM 的输入层内、表示层内的节点之间不存在连接，输入层和表示层的节点之间全连接。这里用 v 代表输入层，用 h 代表表示层，如果 RBM 的输入层和表示层都是二值随机变量，即 $\forall i,j,v_i \in \{0,1\}, h_j \in \{0,1\}$，则它被称为伯努利 RBM。该模型的输入层和表示层的联合概率分布 $p(\boldsymbol{v},\boldsymbol{h})$ 和输入层的概率分布 $p(\boldsymbol{v})$ 分别被定义为

$$\begin{aligned} p(\boldsymbol{v},\boldsymbol{h}) &= \frac{\exp(-E(\boldsymbol{v},\boldsymbol{h}))}{Z} \\ p(\boldsymbol{v}) &= \frac{\sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v},\boldsymbol{h}))}{Z} \end{aligned} \tag{5.1.1}$$

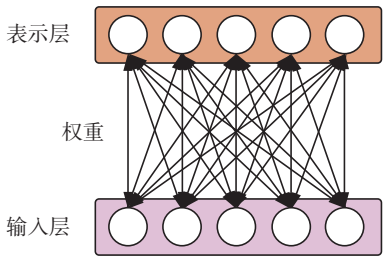


图 5.4　受限玻尔兹曼机模型结构示意图

其中，$Z = \sum_{\boldsymbol{v}} \sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v},\boldsymbol{h}))$，是概率归一化因子，也称为配分函数 (partition function)，

E 是能量函数，如果输入层单元和表示层单元是二值随机变量，则能量函数定义为

$$E(\boldsymbol{v},\boldsymbol{h}) = -\sum_{i,j} v_i W_{ij} h_j - \sum_i c_i v_i - \sum_j b_j h_j \tag{5.1.2}$$

其中，v_i 是输入层的第 i 个节点的值，h_j 是表示层的第 j 个节点的值，W_{ij} 代表它们之间的权重，c_i 是输入层的第 i 个节点的偏置，b_j 是表示层的第 j 个节点的偏置。由概率分布和能量函数的定义，容易推得以下条件概率：

$$\begin{aligned}
p(h_j=1|\boldsymbol{v}) &= \frac{\exp(-E(\boldsymbol{v},h_j=1))}{\exp(-E(\boldsymbol{v},h_j=1))+\exp(-E(\boldsymbol{v},h_j=0))} \\
&= \frac{\exp(\sum_i W_{ij}v_i + \sum_i c_i v_i + b_j)}{\exp(\sum_i W_{ij}v_i + \sum_i c_i v_i + b_j)+\exp(\sum_i c_i v_i)} \\
&= \frac{\exp(b_j + \sum_i W_{ij}v_i)}{1+\exp(b_j + \sum_i W_{ij}v_i)} \\
&= s\left(b_j + \sum_i W_{ij}v_i\right) \\
p(v_i=1|\boldsymbol{h}) &= \frac{\exp(-E(v_i=1,\boldsymbol{h}))}{\exp(-E(v_i=1,\boldsymbol{h}))+\exp(-E(v_i=0,\boldsymbol{h}))} \\
&= \frac{\exp(\sum_j W_{ij}h_j + \sum_j b_j h_j + c_i)}{\exp(\sum_j W_{ij}h_j + \sum_j b_j h_j + c_i)+\exp(\sum_j b_j h_j)} \\
&= \frac{\exp(c_i + \sum_j W_{ij}h_j)}{1+\exp(c_i + \sum_j W_{ij}h_j)} = s\left(c_i + \sum_j W_{ij}h_j\right)
\end{aligned} \tag{5.1.3}$$

其中，$s(x)=\dfrac{1}{1+\mathrm{e}^{-x}}$，为 Logistic 函数。

为了求解 RBM 的参数 $\boldsymbol{W}$、$\boldsymbol{b}$、$\boldsymbol{c}$，记作 θ，研究者提出了很多高效的优化算法，这里仅介绍其中最常用的对比散度 (contrastive divergence, CD)[119] 算法。

RBM 的训练目标是最大化训练集上的对数似然，即最大化 $\log p(\boldsymbol{v})$。对 $p(\boldsymbol{v})$ 的对数，求关于参数 θ 的偏导数，可以得到

$$\begin{aligned}
\frac{\partial \log p(\boldsymbol{v})}{\partial \theta} =& \frac{\partial \log \sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v},\boldsymbol{h})) - \log(\boldsymbol{Z})}{\partial \theta} \\
=& \frac{1}{\sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v},\boldsymbol{h}))} \cdot -\exp(-E(\boldsymbol{v},\boldsymbol{h})) \cdot \frac{\partial E(\boldsymbol{v},\boldsymbol{h})}{\partial \theta} \\
& - \frac{1}{\boldsymbol{Z}} \cdot -\exp(-E(\boldsymbol{v},\boldsymbol{h})) \cdot \frac{\partial E(\boldsymbol{v},\boldsymbol{h})}{\partial \theta}
\end{aligned}$$

$$
\begin{aligned}
&= -\frac{\exp(-E(\boldsymbol{v},\boldsymbol{h}))}{\sum_{\boldsymbol{h}}\exp(-E(\boldsymbol{v},\boldsymbol{h}))}\frac{\partial E(\boldsymbol{v},\boldsymbol{h})}{\partial\theta}+\frac{\exp(-E(\boldsymbol{v},\boldsymbol{h}))}{\boldsymbol{Z}}\frac{\partial E(\boldsymbol{v},\boldsymbol{h})}{\partial\theta}\\
&= -<\frac{\partial E(\boldsymbol{v},\boldsymbol{h})}{\partial\theta}>_{p(\boldsymbol{h}|\boldsymbol{v})}+<\frac{\partial E(\boldsymbol{v},\boldsymbol{h})}{\partial\theta}>_{p(\boldsymbol{v},\boldsymbol{h})}
\end{aligned}
\tag{5.1.4}
$$

其中，$<\cdot>$ 为期望算子，下标表示相应的概率分布。第一项 $p(\boldsymbol{h}|\boldsymbol{v})$ 代表在输入数据条件下表示层的概率分布，比较容易计算，而第二项 $p(\boldsymbol{v},\boldsymbol{h})$ 由于归一化因子 $\boldsymbol{Z}$ 的存在，无法有效地计算该分布。因此，只能通过采样算法获取近似值。

由式(5.1.2)、式(5.1.3)和式(5.1.4)可以得到如下的参数 W、b、c 的梯度计算公式：

$$
\begin{aligned}
\frac{\partial \log p(\boldsymbol{v})}{\partial W_{ij}} &= p(h_j=1|\boldsymbol{v})v_i-\sum_v p(\boldsymbol{v})p(h_j=1|\boldsymbol{v})v_i\\
\frac{\partial \log p(\boldsymbol{v})}{\partial c_i} &= v_i-\sum_v p(\boldsymbol{v})v_i\\
\frac{\partial \log p(\boldsymbol{v})}{\partial b_j} &= p(h_j=1|\boldsymbol{v})-\sum_v p(\boldsymbol{v})p(h_j=1|\boldsymbol{v})
\end{aligned}
\tag{5.1.5}
$$

RBM 的对称结构和其中神经元节点状态的条件独立性，使得 CD 算法可以通过若干步吉布斯 (Gibbs) 采样[120-121] 计算第二项的期望。抽样 k 步的具体过程如下。

$$
\boldsymbol{v}^0 \xrightarrow{p(\boldsymbol{h}|\boldsymbol{v}^0)} h^0 \xrightarrow{p(\boldsymbol{v}|\boldsymbol{h}^0)} \boldsymbol{v}^1 \xrightarrow{p(\boldsymbol{h}|\boldsymbol{v}^1)} \boldsymbol{h}^1 \cdots \boldsymbol{h}^{k-1} \xrightarrow{p(\boldsymbol{v}|\boldsymbol{h}^{k-1})} \boldsymbol{v}^k \xrightarrow{p(\boldsymbol{h}|\boldsymbol{v}^k)} \boldsymbol{h}^k \tag{5.1.6}
$$

k 步抽样的 CD 算法被称为 CD-k 算法。完成 k 步抽样之后，就可以近似计算模型参数的梯度，具体如下。

$$
\begin{aligned}
\delta W_{ij} &= v_i^0 h_j^0 - v_i^k h_j^k\\
\delta c_i &= v_i^0 - v_i^k\\
\delta b_j &= h_j^0 - h_j^k
\end{aligned}
\tag{5.1.7}
$$

有了参数的梯度，就可以使用梯度上升方法更新 RBM 的参数。尽管 CD-k 算法对梯度的近似十分粗略，也被证明并不是任何函数的梯度，但是诸多实验表明了其在训练 RBM 时的有效性。

计算 RBM 在训练数据上的似然是其训练效果的评估最直接的方法。然而，由于其计算涉及归一化因子，计算复杂度非常高，因此，通常只能采用近似方法评估 RBM 的训练质量。例如，退火式重要性抽样 (annealed improtance sampling, AIS) 算法[122] 通过引入

容易计算的辅助分布近似计算归一化因子。尽管该方法能够较准确地计算数据似然，但是其较大的计算量还是不能很好地满足监视 RBM 训练质量的速度需求。在实践中，最常用的评估方法为重构输入误差，和自编器的评估类似。具体而言，即计算输入数据 v^0 和 k 步抽样后的输入数据 v^k 之间的差异值。尽管这一方法并不可靠，但是其因计算简单而在实践中被广泛采用。本文也依据重构误差的大小，来确定 RBM 的训练质量。

2. 建模实数值

上文介绍的基本 RBM 只能建模二值随机变量输入，面对其他形式的输入，如实数值、离散值等，基本的 RBM 将不再适用。大量研究者通过改变能量函数扩展标准 RBM，使其能够建模各种各样的数据分布。对于实数值，一种最直接的方法是使用高斯分布建模输入层，这种 RBM 因此被称为高斯 RBM(Gaussian RBM, GRBM)[123-124]。该模型的能量函数定义为

$$E(\boldsymbol{v},\boldsymbol{h})=\frac{1}{2}\sum_i\frac{(v_i-c_i)^2}{2\sigma_i^2}-\sum_{i,j}\frac{v_i}{\sigma_i}W_{i,j}h_j-\sum_i c_iv_i-\sum_j b_jh_j \tag{5.1.8}$$

其中，σ_i 为输入数据第 i 维的标准差，其他参数和基本 RBM 公式中的参数含义相同。相对应的概率分布如下。

$$\begin{aligned}p(h_j=1|\boldsymbol{v})&=s(b_j+\sum_i\frac{v_i}{\sigma_i}W_{ij})\\p(v_i=1|\boldsymbol{h})&=\mathcal{N}(c_i+\sigma_i\sum_j W_{ij}h_j,\sigma_i^2)\end{aligned} \tag{5.1.9}$$

其中，$\mathcal{N}$ 代表高斯分布。在实际应用中，通常将输入数据特征表示的每一维都归一化成均值为 0、方差为 1 的形式。GRBM 的模型参数求解同样采用 CD 算法。

3. 建模离散值

当输入形式为稀疏的离散值时，通常使用神经主题模型 replicated softmax RBM (RSRBM)[125]。该模型的能量函数定义为

$$E(\boldsymbol{v},\boldsymbol{h})=-\sum_{i,j}\frac{v_i}{\sigma_i}W_{i,j}h_j-\sum_i c_iv_i-D\sum_j b_jh_j \tag{5.1.10}$$

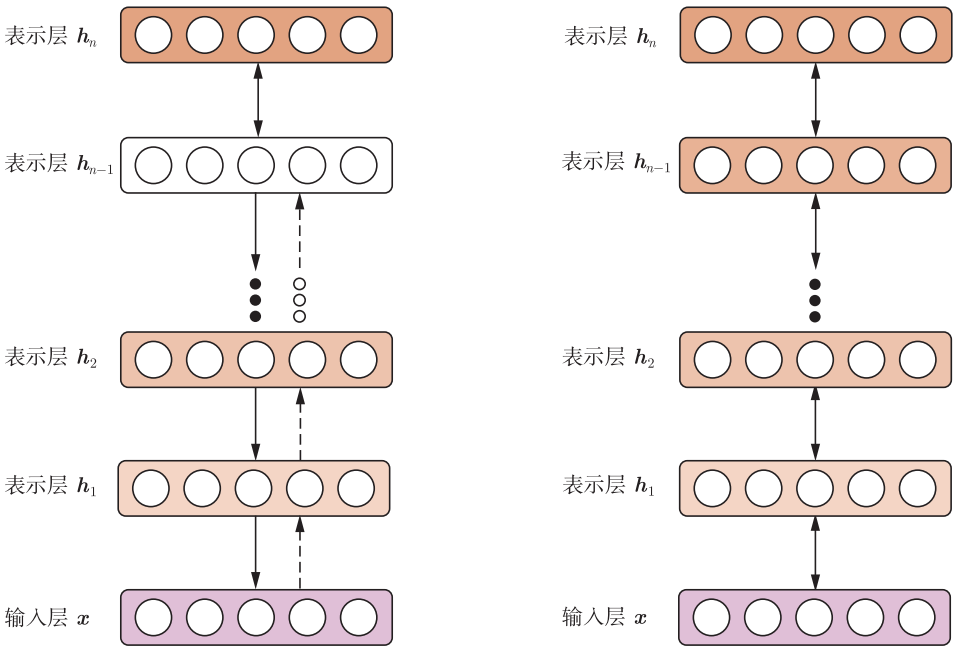
其中，D 是输入层离散值之和，对一篇文档而言，就是文档中的总词数。模型 RSRBM 的参数也采用 CD 算法求解。RSRBM 的输入层可以看作一个多项式随机变量。本文使用 RSRBM 建模文本模态的词袋特征。

4. 深度信念网络和深度玻尔兹曼机

深度信念网络 (deep belief networks, DBN)[126] 和深度玻尔兹曼机 (deep Boltzmann machine, DBM)[127] 是两个典型的基于 RBM 的深度生成模型。

DBN 是 Hinton 等提出的一个包含多个表示层的概率生成模型，它的结构如图 5.5（a）所示，最底部是输入层，其余部分包含 n 个表示层，实线部分代表网络的连接结构，虚线部分是得到输入的多层表示的过程。DBN 是一个混合的网络结构，最上面两层是一个无向图结构的 RBM，其余层自上而下是有向图结构的网络。因此，DBN 的所有层次的联合概率分布可表示为

$$p(\boldsymbol{x}, \boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_{n-1}, \boldsymbol{h}_n) = p(\boldsymbol{h}_n, \boldsymbol{h}_{n-1}) \left(\prod_{k=0}^{n-2} p(\boldsymbol{h}_k | \boldsymbol{h}_{k+1}) \right) \tag{5.1.11}$$



（a）深度信念网络　　（b）深度玻尔兹曼机

图 5.5　基于受限玻尔兹曼机的深层模型结构示意图

其中，$\boldsymbol{x} = \boldsymbol{h}_0$，$p(\boldsymbol{h}_n, \boldsymbol{h}_{n-1})$ 是最上面的 RBM 中变量的联合概率分布，$p(\boldsymbol{h}_k | \boldsymbol{h}_{k+1})$ 是其他 RBM 的表示层到输入层（这里把 $\boldsymbol{h}_{k+1}$ 看作表示层，把 $\boldsymbol{h}_k$ 看作输入层）的条件概

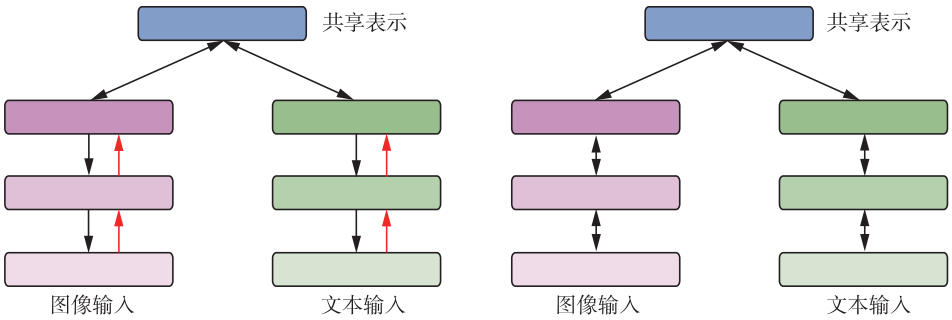
率分布。DBN 的生成过程是一个自顶向下的过程：首先，随机初始化最顶部的 RBM 的表示层 $\boldsymbol{h}_n$；然后利用吉布斯采样进行多次采样，得到 $\boldsymbol{h}_{n-1}$；最后，依次利用条件概率分布 $p(\boldsymbol{h}_{n-2}|\boldsymbol{h}_{n-1}), \cdots, p(\boldsymbol{x}|\boldsymbol{h}_1)$ 采样得到输入 $\boldsymbol{x}$。

由于无法有效地计算后验概率分布 $p(\boldsymbol{h}_{k+1}|\boldsymbol{h}_k)$，DBN 的训练是非常困难的。为了高效地训练 DBN，Hinton 等提出一种贪婪的逐层学习算法[126,128]。该算法的核心思想是利用 RBM 中的输入层到表示层的条件概率分布近似后验概率分布。具体的学习算法为：首先，使用 CD 算法训练最底部的 RBM；然后利用条件概率分布 $p(\boldsymbol{h}_1|\boldsymbol{x})$ 得到表示 $\boldsymbol{h}_1$；接着利用类似的方法，依次完成所有层次 RBM 的训练；最后选择性使用 wake-sleep 算法[126,129]微调整个网络的权值。

DBM 通过直接栈式堆叠多个 RBM 得到，它的结构如图 5.5（b）所示，所有层次之间都是无向连接，因此，DBM 是一个标准的无向图模型。与 DBN 类似，DBM 也具有学习不同层次的复杂表示的能力。但是，在生成和训练过程中，DBM 包含了自顶向下和自底向上两个通道，而 DBN 仅包含自顶向下通道，因此，DBM 学习表达的能力要强于 DBN，相应的高效的训练算法也就更加复杂。为了有效地学习 DBM 的参数，研究者提出大量优秀的训练算法[125,127,130-131]，感兴趣的读者可以自行查阅这些文献。

5. 多模态深度信念网络和多模态深度玻尔兹曼机

多模态深度信念网络（multimodal deep belief networks, MDBN）和多模态深度玻尔兹曼机（multimodal deep Boltzmann machine, MDBM）分别由 DBN 和 DBM 扩展而来，二者的结构如图 5.6 所示。和 MDAE 不同，MDBN 和 MDBM 均为概率生成模型，训练目标为最大化图像和文本模态数据的联合概率分布。模型的训练方法与单模态的 DBN 和 DBM 相似，均包含了单层 RBM 预训练和整体调权两个过程。



（a）多模态深度信念网络　　（b）多模态深度玻尔兹曼机

图 5.6　基于受限玻尔兹曼机的深层模型结构示意图