自动化学报
Acta Automatica Sinica
ISSN 0254-4156,CN 11-2109/TP

《自动化学报》网络首发论文

题目：深度生成模型综述
作者：胡铭菲，刘建伟，左信
DOI：10.16383/j.aas.c190866
收稿日期：2019-12-19
网络首发日期：2020-09-21
引用格式：胡铭菲，刘建伟，左信．深度生成模型综述．自动化学报．
https://doi.org/10.16383/j.aas.c190866

自 动 化 学 报
ACTA AUTOMATICA SINICA

深度生成模型综述

胡铭菲 [1] 刘建伟 [1] 左信 [1]

摘要 通过学习可观测数据的概率密度而随机生成样本的生成模型在近年来受到人们的广泛关注，网络结构中包含多个隐藏层的深度生成式模型以更出色的生成能力成为研究热点，深度生成模型在计算机视觉、密度估计、自然语言和语音识别、半监督学习等领域得到成功应用，并给无监督学习提供了良好的范式。本文根据深度生成模型处理似然函数的不同方法将模型分为三类：第一类方法是近似方法，包括采用抽样方法近似计算似然函数的受限玻尔兹曼机和以受限玻尔兹曼机为基础模块的深度置信网络、深度玻尔兹曼机和亥姆霍兹机，与之对应的另一种模型是直接优化似然函数变分下界的变分自编码器以及其重要的改进模型，包括重要性加权自编码和可用于半监督学习的深度辅助深度模型；第二类方法是避开求极大似然过程的隐式方法，其代表模型是通过生成器和判别器之间的对抗行为来优化模型参数从而巧妙避开求解似然函数的生成对抗网络以及重要的改进模型，包括 WGAN、深度卷积生成对抗网络和当前最顶级的深度生成模型 BigGAN；第三类方法是对似然函数进行适当变形的流模型和自回归模型，流模型利用可逆函数构造似然函数后直接优化模型参数，包括以 NICE 为基础的常规流模型、变分流模型和可逆残差网络（i-ResNet），自回归模型（NADE）将目标函数分解为条件概率乘积的形式，包括神经自回归密度估计（NADE）、像素循环神经网络（PixelRNN）、掩码自编码器（MADE）以及 WaveNet 等。详细描述上述模型的原理和结构以及模型变形后，阐述各个模型的研究进展和应用，最后对深度生成式模型进行展望和总结。

关键词 深度生成式模型, 受限玻尔兹曼机, 变分自编码器, 流模型, 生成对抗网络, 自回归分布估计
引用格式 胡铭菲，刘建伟，左信. 深度生成模型综述. 自动化学报, 20XX,XX(X): X—X
DOI 10.16383/j.aas.c190866

Survey on Deep Generative Model

HU Ming-Fei[1] LIU Jian-Wei[1] ZUO Xin[1]

Abstract The generative model, which can generate samples randomly by learning the probability density of observable data, has been widely concerned for the past few years. It has been successfully applied in a wide range of fields, such as image generation, image restoration, density estimation, natural language and speech recognition, style transfer and super resolution, and so on. Deep generative model with multiple hidden layers in the network structure becomes a research hotspot because of its better generation ability. Depending on the different methods of calculating the maximum likelihood function, we divide the models into three types: the first kind of method is the approximate method, which use the sampling method to calculate approximately the likelihood function, such as Restricted Boltzmann machines and Deep Belief Network, Deep Boltzmann Machines, helmholtz machine based on RBM. The alternatives are to optimize directly the variational lower bound of likelihood function, it is named as variational autoencoder. The important improvements to these variants include Importance Weighted Autoencoders and Auxiliary Deep Generative Models; the second kind is implicit methods, the representative model is Generative Adversarial Nets, GAN's model parameters is optimized by the adversaring behavior between the generator and the discriminator. The principal instantiations of GAN include Wasserstein GAN, Deep Convolutional Generative Adversarial Networks and BigGAN. The third kind involve Flow and Neural Autoregressive Net, the main variations of the Flow paradigm include Normalizing Flow based on Nonlinear Independent Components Estimation, Invertible Residual Networks and Variational Inference with Flow. The successful improvements to the Neural Autoregressive Net include Neural Autoregressive Distribution Estimation, Pixel Recurrent Neural

收稿日期 2019-12-19 录用日期 2020-07-27
Manuscript received December 19, 2019; accepted July 27, 2020
国家重点研发计划项目(2016YFC0303703)资助
Supported by National Key Research and Development Program of China (2016YFC0303703)
1. 中国石油大学（北京）自动化系 北京 102249
1. Department of Automation, China University of Petroleum, Beijing 102249

Network, Masked Autoencoder for Distribution Estimation and WaveNet. We outline the principle and structure of these deep generative models, and look forward to the future work.

Key words deep generative models, restricted boltzmann machine, variational auto-encoder, flow, generative adversarial nets, neural autoregressive distribution estimator

Citation Hu Ming-Fei,Liu Jian-Wei,Zuo Xin. Survey on deep generate model.*Acta Automatica Sinica*, 20XX, XX(X): X—X

受益于当前计算机性能的快速提升，学习可观测样本的概率密度并随机生成新样本的生成模型成为热点。相比于需要学习条件概率分布的判别模型，生成模型的训练难度大、模型结构复杂，但除了能够生成新样本外，生成模型在图像重构、缺失数据填充、密度估计、风格迁移和半监督学习等应用领域也获得了巨大的成功。当前可观测样本的数量和维数都大幅度增加，浅层的生成模型受到性能瓶颈的限制而无法满足应用需求，从而被含有多个隐藏层的深度生成模型替代，深度生成模型能够学习到更好的隐表示，模型性能更好。 本文对有重要意义的深度生成模型进行全面的分析和讨论，对各大类模型的结构和基本原理进行梳理和分类。本文第 1 节介绍深度生成模型的概念和分类；第 2 节介绍受限玻尔兹曼机和以受限玻尔兹曼机为基础模块的几种深度生成模型，重点内容是各种模型的不同训练算法;第 3 节介绍变分自编码器的基本结构、变分下界的推理和重参数化方法；第 4 节介绍生成对抗网络，主要内容为模型原理、训练方法和稳定性研究，以及两种重要的模型结构；第 5 节总结了流模型的结构，详细介绍了流模型的技术特点；第 6 节分析了自回归模型的模型结构以及几种重要分支的研究进展；第 7 节将介绍生成模型中的两个小分支：矩阵匹配模型和随机生成模型；第 8 节对深度生成模型存在的问题进行分析讨论，并对未来的研究方向和发展趋势做出了展望。

1 深度生成模型概述

深度生成模型的目标函数是数据分布与模型分布之间的距离，可以用极大似然法进行求解。从处理极大似然函数的方法的角度，可将深度生成模型分成如下三种，分类内容如图 1 所示。具体分类方式如下：

第一种方法是通过变分或抽样的方法求似然函数的近似分布，这种方法可称为近似方法，主要包括受限玻尔兹曼机[1]和变分自编码器[2]。用抽样方法近似求解似然函数的受限玻尔兹曼机属于浅层模型，以该模型为基础模块的深度生成模型包括深度玻尔兹曼机和深度置信网络两种；变分自编码器用似然函数的变分下界作为目标函数，这种使用变分下界替代似然函数的近似方法的效率比受限玻尔兹曼机的抽样方法高很多，实际效果也更好，变分自编码器具有代表性的模型包括重要性加权自编码、辅助深度生成模型等。

第二种方法是避开求极大似然过程的隐式方法，其代表模型是生成对抗网络[3]。生成对抗网络利用神经网络的学习能力来拟合两个分布之间的距离，巧妙地避开了求解似然函数的难题，是目前最成功、最有影响力的生成模型，其具有代表性的模型很多，例如深度卷积生成对抗网络、WGAN 和当前生成能力最好的 BigGAN；另外利用参数化马尔科夫过程代替直接参数化似然函数的生成随机网络[4]也属于此类方法。

第三类方法是对似然函数进行适当变形，变形的目的是为了简化计算，此类方法包括流模型[5]和自回归模型[6]两种模型。流模型利用可逆网络构造似然函数之后直接优化模型参数，训练出的编码器利用可逆结构的特点直接得到生成模型。流模型包括常规流模型、变分流模型和可逆残差网络三种；自回归模型将目标函数分解为条件概率乘积的形式，这类模型有很多，具有代表性的包括像素循环神经网络、掩码自编码器以及成功生成逼真的人类语音样本的 WaveNet 等。

2 受限玻尔兹曼机

玻尔兹曼机（Boltzmann machines，BM）是由能量函数定义的结构化无向图概率模型，用于学习二值向量上的任意概率分布，广义上把基于能量的模型都称作 BM。BM 层内各单元之间和各层之间均为全连接关系，权值大小表示单元之间的相互作用关系。BM 变种的流行程度早已超过了该模型本身，其中最主要的衍生模型是属于生成模型的受限玻尔兹曼机 RBM[1]（Restricted Boltzmann machines，RBM）和以 RBM 为基础模块的深度置信网络（Deep Belief Network，DBN）和深度玻尔兹曼机（Deep Boltzmann Machines，DBM）等深度生成模型，是深度学习中的典型代表，曾受到广泛研究，国内外均有关于该模型的综述文章[7-8]。这类

模型能够学习高维特征和高阶概率依赖关系并成功应用在降维、特征提取等领域，是最早出现的深度生成模型。

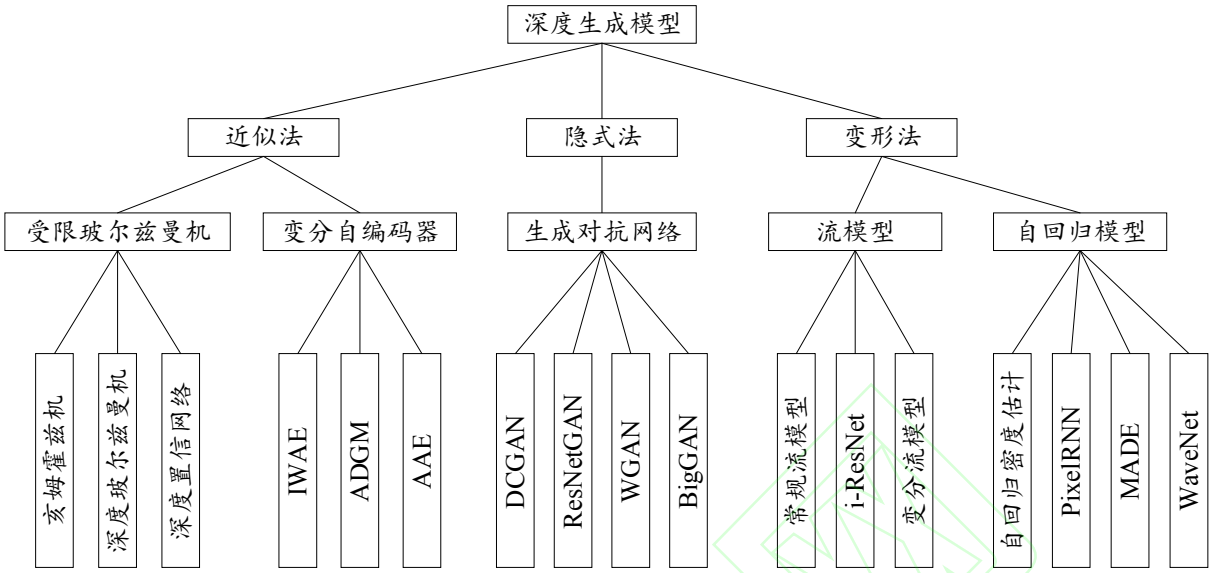

图 1 深度生成模型分类

Fig.1 Deep generative models classification

本节将详细介绍训练过程中存在的问题以及解决方法，然后以该模型为基础分别介绍 DBN 和 DBM 的结构和相关算法，最后总结近些年来重要的相关模型和算法改进。

2.1 受限玻尔兹曼机

RBM 面世之初因其结构特点而命名为簧风琴，曾是深度概率模型中的常见组件。RBM 是 BM 的一种特殊拓扑结构，能够描述变量之间的高阶相互作用，其模型结构具有完备的物理解释，训练算法有严谨的数理统计基础[7]。

2.1.1 模型结构

RBM 的单元被分成两组，每个组称作一层，层之间的连接由权值矩阵描述。RBM 与 BM 均是包含一层可见变量和一层隐藏变量的浅层模型，两者的区别是 RBM 的层内神经元之间没有连接。RBM 的上层是不可观测的隐藏层，下层是可观测的输入层，两层的所有神经元只取 1 或 0，这两个值分别对应该神经元激活或未激活的两种状态，模型结构如图 2 所示。

图中 x 表示可见层神经元（输入）；隐藏层神经元 z 表示输入的映射； a,b,W 分别表示可见层偏置向量、隐藏层偏置向量和权重矩阵。RBM 的能量函数由下式给出：

$$E(x_i,z_j) = -\sum a_i x_i - \sum b_j z_j - \sum\sum x_i W_{ij} z_j \quad (1)$$

式中 $E(x_i,z_j)$ 表示能量函数，这种形式使得模型中任意变量的概率可以无限趋于 0 但无法达到 0。

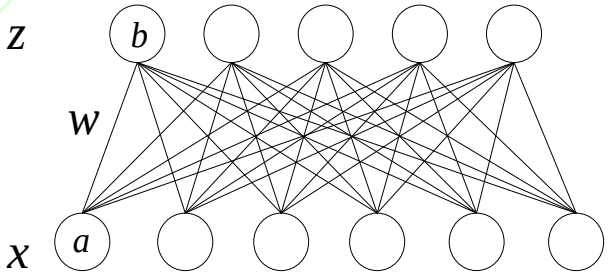

图 2 受限玻尔兹曼机

Fig.2 Restricted Boltzmann machines

RBM 的联合概率分布由能量函数指定：

$$P(x_i,z_j) = \frac{1}{Z}\exp(-E(x_i,z_j)) \quad (2)$$

其中 Z 是被称为配分函数的归一化常数：

$$Z = \sum\sum \exp(-E(x_i,z_j)) \quad (3)$$

二分图结构特有的性质使 RBM 的条件分布 $P(z_j \mid \boldsymbol{x})$ 和 $P(x_i \mid \boldsymbol{z})$ 是可因式分解的，使条件分布的计算和抽样都比 BM 简单。从联合分布中可以推导

出条件分布：

$$P(z_j=1|\boldsymbol{x})=\frac{1}{Z'}\prod\exp(b_j z_j+z_j W_{ji}x_i) \quad (4)$$

根据条件分布因式相乘的原理可将可见变量的联合概率写成单个神经元分布的乘积：

$$\begin{aligned}&P(z_j=1|\boldsymbol{x})\\&=\frac{P(z_j=1|\boldsymbol{x})}{P(z_j=0|\boldsymbol{x})+P(z_j=1|\boldsymbol{x})}\\&=\mathrm{sigm}(b_j+\sum_i x_i W_{ij})\end{aligned} \quad (5)$$

训练 RBM 模型使用极大似然法，似然函数的对数如下表示：

$$\mathrm{In}L(\theta|\boldsymbol{x})=\mathrm{In}\sum_z e^{-E(x_i,z_j)}-\mathrm{In}\sum_{x,z}e^{-E(x_i,z_j)} \quad (6)$$

2.1.2 配分函数

训练 RBM 时需要计算边缘概率分布，而无向图模型中未归一化的概率必须除以配分函数进行归一化，以获得有效的概率分布 $P(x)$：

$$P(x)=\frac{1}{Z}\tilde{P}(x) \quad (7)$$

其中 $\tilde{P}(x)$ 表示未归一化的概率， Z 表示配分函数。

配分函数是 $\tilde{P}(x)$ 所有状态的积分，理论上难以求解。配分函数的计算依赖于模型参数，对数似然关于模型参数的梯度可以分解为：

$$\nabla_\theta \log P(x;\theta)=\nabla_\theta \log\tilde{P}(x;\theta)-\nabla_\theta \log Z(\theta) \quad (8)$$

式（8）中的前后两项分别对应训练的正相（positive phase）和负相（negative phase）。大部分无向图模型都有计算简单的正相和难以计算的负相，RBM 的隐藏单元在给定可见单元时是条件独立的，属于典型的负相，不容易计算。对负相的进一步分析可以推导出如下结果：

$$\nabla_\theta \log Z=\mathrm{E}_{x\sim P(x)}\nabla_\theta \log\tilde{P}(x) \quad (9)$$

式（9）是使用各类蒙特卡洛方法近似最大化似然的基础。负相涉及从模型分布中抽样，一般被认为代表了模型不正确的信念，类似人类做梦的过程：大脑在清醒时经历的真实事件会按照训练数据分布的梯度更新模型参数，在睡觉时按照模型分布的负梯度最小化配分函数，然后更新模型参数。但在 RBM 学习中需要交替执行正相和负相的计算才能完成参数更新。

2.1.3 配分函数的估计方法

计算 RBM 的配分函数是训练模型的主要难点，主要体现为计算困难、计算量大，主要计算方法可以分为三类，其中对比散度算法以计算量相对最小、精度尚可而成为 RBM 的主要算法，三类算法具体如下：

第一类算法是通过引入中间分布直接估计配分函数的值，中间分布的计算需要使用蒙特卡洛马尔科夫链或重要性采样，代表算法是退火重要性抽样算法[18]（Annealed Importance Sampling，AIS）。AIS 是无向图模型中直接估计配分函数的常用方法，擅长估计高位空间中复杂分布上的配分函数，缺点是用蒙特卡洛估计配分函数的方法效率较低。

退火重要性抽样 AIS 通过引入中间分布来缩小模型分布和数据分布之间的距离，从而估计高维空间上多峰分布的配分函数。该方法先定义一个已知配分函数的简单模型，然后估计给定的简单模型和需要估计模型的配分函数之间的比值，例如在权重为 0 的 RBM 和学习到的权重之间插值一组权重不同的 RBM，此时配分函数比值为：

$$\frac{Z_1}{Z_0}=\frac{Z_{\eta_1}}{Z_0}\frac{Z_{\eta_2}}{Z_{\eta_1}}\cdots\frac{Z_{\eta_{n-1}}}{Z_{\eta_{n-2}}}\frac{Z_1}{Z_{\eta_{n-1}}}=\prod_{j=0}^{n-1}\frac{Z_{\eta_{j+1}}}{Z_{\eta_j}} \quad (10)$$

如果对于任意的 $0\le j\le n-1$ 都能使分布 P_{η_j} 和 $P_{\eta_{j+1}}$ 足够接近，则可以用重要性抽样估计每个因子的值，然后使用这些值计算配分函数比值的估计值。中间分布一般采用目标分布的加权几何平均：$P_{\eta_j}\propto P_1^{\eta_j}P_0^{1-\eta_j}$，考虑到重要性权重，最终的配分函数比值为：

$$\frac{Z_1}{Z_0}\approx\frac{1}{K}\sum_{k=1}^{K}W^k \quad (11)$$

其中 W^k 表示第 k 次抽样时的重要性权重，W^k 的值可从转移算子乘积得到。

第二类计算配分函数的算法是构造新目标函数替代配分函数，避免直接求解配分函数的过程，主要包括得分匹配[14]（Score Matching，SM）和噪声对比估计[16]（Noise Contrastive Estimation, NCE）。SM 算法精度很高，但缺点是计算量较大，只稍低于使用 MCMC 采样的 AIS 算法；NCE 算法的精度一般且计算量也很大，但该算法将无监督学习的似

然函数估计问题转化为学习概率的二值分类问题，巧妙的避开求解配分函数和估计数据分布的难题，是一个具有创造性的方法。

得分匹配 得分表示对数概率密度关于模型参数的导数，SM 算法用模型分布和数据分布的对数对输入求导后的差的平方代替边缘概率分布作为 RBM 的新目标函数：

$$L(x)=\frac{1}{2}\left\|\nabla_x \log P_g(x)-\nabla_x \log P_r(x)\right\|_2^2 \quad (12)$$

SM 算法的思路与 CD 算法类似，但该算法以计算量为代价得到更精确的数据概率分布估计。过大的计算量使 SM 算法通常只用于单层网络或者深层网络的最下层，另外算法中的求导过程说明该算法只能应用在连续数据中，比率匹配（Ratio Matching, RM）将该方法推广到离散数据中[15]。

噪声对比估计 NCE[16]在每个样例中引入类标签 y，规定训练数据上的样本属于一类，然后引入噪声分布，从噪声分布抽样得到的样本属于另一类，噪声样本和训练样本以及类标签合在一起作为新的训练样本，在训练过程中指定类先验概率：

$$P(y=1)=\frac{1}{2} \quad (13)$$

其条件概率可以表示为：

$$\begin{aligned} P(x\mid y=1)&=P_g(x)\\ P(x\mid y=0)&=P_r(x)\end{aligned} \quad (14)$$

然后用相同方法构造其他类的联合分布。NCE 这种转换的技巧促使了后来生成对抗网络的诞生，但因为训练困难使该算法更适合随机变量较少的问题，在给定单词上下文，计算单词的条件概率分布的任务上获得了很好的效果。

第三类算法是直接估计配分函数关于参数的近似梯度，这种基于马尔科夫链的近似方法主要包括对比散度[9]（Contrastive Divergence，CD）、持续对比散度[12]（Persistent Contrastive Divergen，PCD）和快速持续对比散度（Fast Persistent Contrastive Divergen，FPCD）[13]三种。CD 算法是一种计算量很低的算法，计算效率远高于其他两类方法，缺点是精度不高，需要使用 MCMC 等算法精调模型；PCD 算法利用持续马尔科夫链提高 CD 算法的精度；FPCD 用单独的混合机制改善 PCD 的混合过程，提高算法的训练速度和稳定性。

对比散度 CD 算法在每个步骤用数据分布中抽取的样本初始化马尔科夫链，有效减少抽样次数、提高计算效率。该算法用估计的模型概率分布与数据分布之间的距离作为度量函数，首先从训练样本抽样，利用 n 步 Gibbs 抽样达到平稳分布后再固定概率分布的参数，从该平稳分布中抽样，用这些样本计算权重 $w_{i,j}$ 的梯度为：

$$\frac{1}{l}\sum_{x\in S}\frac{\partial \ln L(\theta\mid x)}{\partial W_{i,j}}=\left(P_r-P_g\right) \quad (15)$$

其中 $S=\{x_1,\cdots,x_l\}$ 表示训练集，r 表示数据参数，g 表示模型参数。CD 算法是一种近似算法，Perpiñan 等人[10]证明 CD 算法最终会收敛到与极大似然估计不同的点，因此该算法更适合作为一种计算代价低的参数初始化方法。Bengio[11]证明了 CD 等价于去掉最小项的 MCMC 算法，解释了算法中偏差的由来。

实验显示使用 1 步 Gibbs 抽样的 CD-1 算法就能得到不错的学习效果。尽管 CD-1 算法的偏差较大，但可以用于浅层模型的预训练，再把浅层模型堆叠起来形成深度模型，这个优势也使得快速高效的 CD 算法成为目前训练 RBM 的标准算法，并用于 DBN 等深层模型的预训练。

持续对比散度 PCD 算法在应用数学领域被称为随机最大似然（stochastic maximum likelihood, SML）算法，是 CD 的一种改进算法，在每个步骤中用先前梯度步骤的状态值初始化马尔可夫链，有效提高算法精度，并弥补了 CD 算法无法最大化似然函数的缺陷。

PCD 用持续马尔科夫链得到负相的近似梯度，令 t 步的持续马尔科夫链状态为 x_t，则参数的梯度可以近似为：

$$\frac{\partial \ln P(x)}{\partial \theta}\approx\lambda\left(\mathrm{E}[xz^{\mathrm{T}}]-\mathrm{E}[x_{t+k}z_{t+k}^{\mathrm{T}}]\right) \quad (16)$$

每个马尔科夫链在整个学习过程中都在不断更新，这种做法可以更容易找到模型的所有峰值。除了计算量比较大以外，PCD 的另一个缺点是超参数 k 的取值高度依赖于具体问题，如果学习率或训练时间不合适会使马尔科夫链的遍历性下降进而产生不稳定的误差。

快速持续对比散度 FPCD 引入了单独的混合机制以改善 PCD 持续马尔科夫链的混合过程，使算法性能不会因为训练时间过长等的影响而恶化。

FPCD 使用一组额外的权值 θ^- 提高样本的混合速率，原本的权值当作慢速权值 θ^+ 估计数据的期望值，然后使用混合参数 $\theta=\theta^+ +\theta^-$ 作为持续马尔科夫链的更新样本。FPCD 中的快速权重通常使用较大的学习率并保持足够长的迭代次数，马尔可夫链改变峰值之后再对快速权重使用权重衰减，促使学习率收敛到较小的值，从而提高混合速率。

2.1.4 RBM 模型研究进展

基础 RBM 的二值神经元很容易扩展到实值或向量形式，即高斯-伯努利受限玻尔兹曼机。该模型假设可见层数据是服从高斯分布的实数用以处理实值数据，其能量函数可以定义成如下形式：

$$E(x,z)=\sum_{i=1}^{n}\frac{(z_i-a_i)^2}{2\sigma_i^2}-\sum_{i=1}^{n}\sum_{j=1}^{n}W_{ij}z_j\frac{x_i}{\sigma_i}-\sum_{j=1}^{n}z_jb_j \quad (17)$$

根据能量函数得到似然函数和条件概率分布的过程与二值 RBM 相同。Salakhutdinov[18]利用受限玻尔兹曼机在高维数据中提取特征的功能，从大型文本中提取主题，此时可见层神经元服从伯努利分布。在条件概率的计算中需要归一化指数函数 Softmax 调整概率分布，归一化概率分布并有效避免数值溢出。

Montufar 等人[19]从理论上证明了 RBM 与[0,1]之间的任意概率分布的相对熵上界。Sutskever 等人[20]提出了循环回火 RBM，通过让隐藏层神经元之间的权重传递更多信息，使模型能容易准确的推断和更新梯度。Nair 和 Hinton[21]提出用阶梯 S 型神经元近似替换 RBM 的二值神经元，这种神经元更加自然地学习光照等自然界变化，提高了模型学习特征的能力。基于因式分解的三值 RBM[22]将实值图像映射到因式分解的输出，令隐藏层神经元表示可见层神经元的协方差和阈值，提高模型对小图形数据集的识别准确性。

2.2 深度置信网络

深度置信网络[23]（Deep Belief Network，DBN）的出现使深度学习再次受到人们的关注，缓解了深度模型很难优化的缺陷，在 MNIST 数据集上的表现超过当时占统治地位的支持向量机。尽管 DBN 与后来出现的深度生成式模型相比已没有优势，但它以深度学习历史中里程碑式的重要模型而得到认可和广泛研究。

2.2.1 DBN 模型结构

置信网络一般是有向图模型，而深度置信网络的最低层具有无向连接边，其它各层之间为有向连接边。DBN 具有多个隐藏层，隐藏层神经元通常只取 0 和 1，可见层单元取二值或实数，除顶部两层之间是无向连接外，其余层是有向边连接的置信网络，箭头指向可见层，因此 DBN 属于有向概率图模型，其结构如图 3 所示。

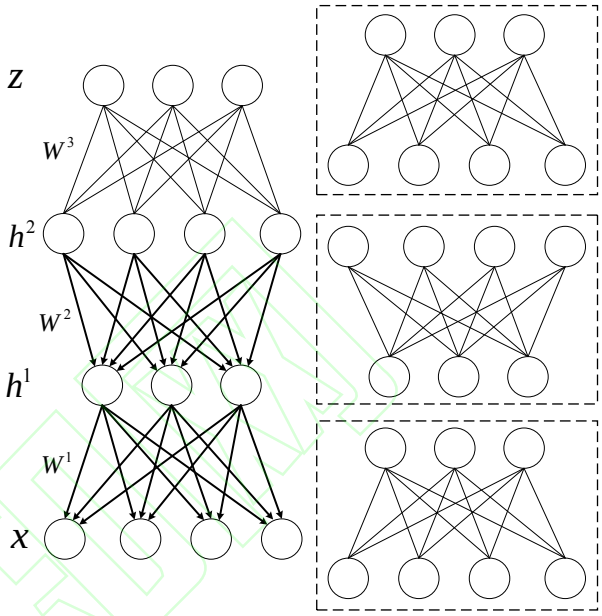

图 3 深度置信网络结构

Fig.3 The structure of deep belief networks

以 DBN 的前两个隐藏层 h^1 和 h^2 为例说明模型结构。此时 DBN 的联合概率分布定义为

$$P(x,h^1,h^2;\theta)=P(x\mid h^1;W^1)P(h^1,h^2;W^2) \quad (18)$$

其中 $\theta=\{W^1,W^2\}$ 表示模型参数，$P(x\mid h^1;W^1)$ 表示有向的置信网络。$P(h^1,h^2;W^2)$ 可以用训练 RBM 的方法预训练，因为可见层和第一个隐藏层的联合分布与 RBM 的联合概率分布形式相同：

$$P(x,h^1;\theta)=\sum_{h^2}P(x,h^1,h^2;\theta) \quad (19)$$

因此 DBN 在训练过程中可以将任意相邻两层看作一个 RBM。从 DBN 生成样本时，先在顶部隐藏层内运行几步 Gibbs 抽样，然后按照条件概率由上至下依次计算各层的值，最后就能得到从 DBN 产生的样本。

2.2.2 DBN 的目标函数

隐藏层 h^1 到可见层是有向网络，无法直接得到隐藏层条件概率，因此假设条件概率的近似分布为

$Q(h^1 \mid x)$。利用 Jensen 不等式可以得到 DBN 的似然函数：

$$\begin{aligned} &\log P(x) \\ &\geq \sum_{h^1} Q(h^1 \mid x) \log \frac{P(x, h^1)}{Q(h^1 \mid x)} \\ &= \sum_{h^1} Q(h^1 \mid x) \log P(\cdot) + H(Q(h^1 \mid x)) \end{aligned} \quad (20)$$

其中 $\log P(\cdot) = \log P(h^1) + \log P(x \mid h^1)$， $H(\cdot)$ 表示熵函数。该目标函数本质上是 RBM 目标函数的变分下界。

2.2.3 贪婪逐层预训练算法

随机初始化参数下的 DBN 很难训练，需要使用贪婪学习算法调整模型参数使模型有容易训练的初始值。贪婪学习算法采用逐层预训练的方式，首先训练 DBN 的可见层和隐藏层 h^1 之间的参数，固定训练好的参数并使权重 W_2 等于 W_1 的转置。在训练权重 W_3 时，从条件概率中抽样获得 h^2， 训练方法与 RBM 的训练方式相同。贪婪逐层预训练算法提供了两种获得 h^2 和 z 的方法，如图 4 所示。

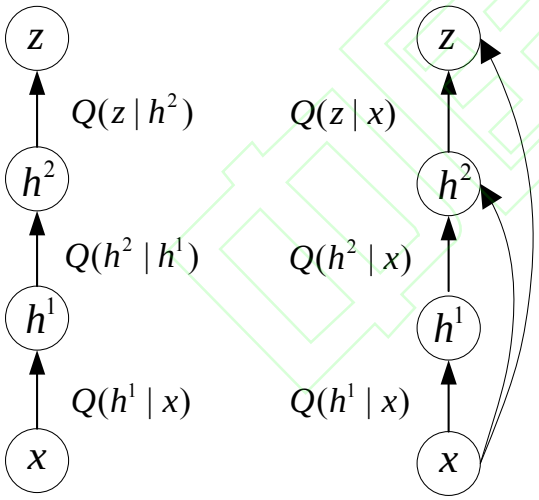

图 4 两种贪恋逐层学习算法

Fig.4 Two kinds of greedy layer-wise pre-training

一种方法是从条件概率 $Q(h^2 \mid h^1)$ 中抽样获得 h^2，另一种算法直接从条件概率 $Q(h^2 \mid x)$ 中抽样。使用贪婪算法训练有向概率图时通常使用第二种方法抽样，因为它在实际应用中表现更好[24]。

2.2.4 亥姆霍兹机

DBN 的出现是深度模型的巨大进步，但从数学的角度推导出变量之间的真实关系需要很大的计算量，降低了模型的训练速度。Hinton 和 Neal 等人提出的亥姆霍兹机（Helmholtz Machine，HM）[25]可以看成是另一种连接形式的 DBN。HM 的基本思想是保持 DBN 有向性的同时，在层之间增加单独的权重，使最上层的隐变量可以和可见层进行通信，能有效提高模型训练速度，HM 模型结构如图 5 所示。

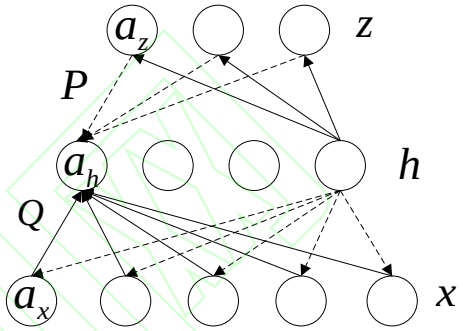

图 5 亥姆霍兹机

Fig.5 Helmholtz Machine

图 5 展示了全连接网络 HM 中的部分连接，其中 a_x 表示可见层 x 的偏置，a_h 和 a_z 表示两个隐藏层 h 和 z 的偏置，指向向上的实线连接 Q 表示认知权重，指向向下的虚线 P 连接表示生成权重。Hinton 等人提出了用于训练 HM 的醒睡算法[26]。醒睡算法将训练过程分为清醒和睡眠两个阶段：清醒阶段是认知过程，通过真实数据和认知权重由下至上得到神经元状态，然后用梯度下降调整生成权重；睡眠阶段是生成过程，通过顶层的特征表示和生成权重生成底层状态，然后用训练过的生成过程得到的权值和偏置调节认知权重。

2.2.5 DBN 研究进展

DBN 模型的扩展研究可以归纳为两方面，一方面是研究如何高效处理大规模无标签数据，更好的进行无监督学习，另一方面是利用基本的反向传播等方法微调参数使其获得更好的性能，改动模型结构以适应更多的应用领域。

Mohamed 等人[27]描述并分析了用连续判别训练准则优化 DBN 权值等参数的方法，说明基于序列训练准则学习的 DBN 性能优于原始 DBN。Dahl 等人[28]提出 DBN-HMM 系统用于语音元素识别，是一种基于 DBN 的隐马尔科夫模型，该模型利用

最小化语音元素误差率和极大似然改善预测精度，模型性能明显优于高斯混合的 HMM。

用贪心算法训练的 DBN 提取结构复杂的无标签高维数据的特征，然后再用高斯核对特征进行分类或回归，这种方法的效果比使用原输入的核方法好[29]，利用反向传播精调 DBN 参数可以进一步提高精度。Hinton[30]利用 DBN 将高维数据转为维数较低的数据，用这种类似深度自编码器的功能对数据降维，效果比主成分分析和奇异值分析更好。

用 DBN 直接处理高维图片很困难， Lee[31]提出在 DBN 加入卷积层，利用卷积神经网络缩小真实图片的维数、提取初级特征，作者还提出了概率最大池化的池化层，使带有卷积层的 DBN 能够正常训练，从而具有高级视觉特征的学习能力。卷积 DBN 可以用于处理序列图像，从连续的高维图像中提取有用特征，并将在多层结构中提取的底层动作特征用于动作识别。

2.3 深度玻尔兹曼机

深度玻尔兹曼机[32]（Deep Boltzmann Machines，DBM）是以 RBM 为基础模块的另一种深度生成式模型，与 DBN 的区别是 DBM 为无向概率图模型，属于马尔科夫随机场模型，通过简单的自下而上的传播就能快速初始化模型参数，并结合从上至下的反馈处理数据的不确定性问题，缺点是在生成样本时需要一定计算量。

2.3.1 模型结构

二值 DBM 的神经元只取 0 和 1，同样容易扩展到实值，层内每个神经元相互独立，条件于相邻层中的神经元，包含三个隐藏层的 DBM 结构如图 6 所示。

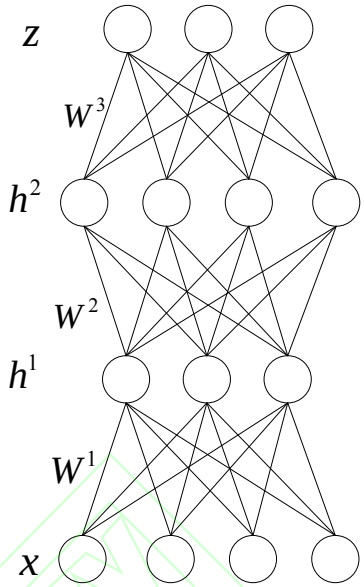

图 6 深度玻尔兹曼机

Fig.6 Deep Boltzmann machines

DBM 是基于能量的模型，因此模型的联合概率分布由能量函数定义，DBM 前三层的联合概率分布为：

$$P(x,h^1,h^2)=\frac{1}{Z_\theta}e^{-E(x,h^1,h^2)} \qquad (21)$$

为了简化表示，省略了偏置参数的能量函数形式如下：

$$E(x,h^1,h^2)=-x^{\mathrm{T}}W^1h^1-(h^1)^{\mathrm{T}}W^2h^2 \qquad (22)$$

DBM 的二分结构意味着可以用 RBM 相同形式的条件独立分布假设，给定相邻层神经元值时，层内单元彼此条件独立。这里给出 DBM 中可见层和隐藏层 h^1 的条件概率公式：

$$\begin{aligned} P(x_i=1\mid h^1)&=\mathrm{sigm}(W^1_{i,:}h^1)\\ P(h^1_i=1\mid x,h^2)&=\mathrm{sigm}(x^{\mathrm{T}}W^1_{:,i}+W^2_{i,:}h^2)\end{aligned} \qquad (23)$$

DBM 在训练过程中使用 Gibbs 抽样更新参数，每次以一层为一个模块进行更新。例如给定偶数层，此时关于奇数层的分布是独立的，则可将该层相邻的奇数层作为条件进行抽样，与 DBN 相比这种训练方法的效率更高。

2.3.2 DBM 的训练方法

深度玻尔兹曼机所有层之间均为无向传播，因此训练方法与 DBN 有差异，但用随机初始化马尔科夫链近似似然函数梯度的过程很慢，因此采用变分的方法之后用平均推断估计数据期望、用 MCMC

随机近似过程近似模型期望。

平均场近似是变分推断的一种简单形式，该方法假设 DBM 后验分布的近似分布限制为完全因子的，通过一些简单分布族的乘积近似特定的目标分布。令 $P(h^1,h^2\mid x)$ 的近似分布为 $Q(h^1,h^2\mid x)$，则条件概率的平均场分布意味着：

$$Q(h^1,h^2\mid x)=\prod_j Q(h_j^1\mid x)\prod_k Q(h_k^2\mid x) \quad (24)$$

平均场推断是通过最小化近似分布和真实分布的 KL 散度以使两者尽量接近，以求得模型参数和状态的值：

$$D_{\mathrm{KL}}(Q\,\|\,P)=\sum_h Q(h^1,h^2\mid x)\log\left(\frac{Q(h^1,h^2\mid x)}{P(h^1,h^2\mid x)}\right) \quad (25)$$

为了方便计算，MCMC 方法假设真实后验概率由隐藏层组成的全因式分解的均匀分布：

$$Q(h\mid x)=\prod\prod\prod Q_1(\cdot)Q_2(\cdot)Q_3(\cdot) \quad (26)$$

为了更好的提取有效特征，减少精调时的计算负担，DBM 的预训练过程是必要的。DBM 预训练过程的使用方法与 DBN 略有差异，经过预训练的 DBM 能够生成 MNIST 数据集的样本，在视觉目标识别等任务上也有不错的效果，但作为含有多个隐藏层的深度网络，频繁使用蒙特卡洛抽样会造成参数的不确定性和计算负担，因此在实际应用中最多使用三层网络。

2.3.3 DBM 研究进展

训练 DBM 时需要用贪婪逐层预训练算法对模型参数进行预训练，这一过程的缺点是无法监督模型参数的训练进度，不同模块使用不同结构的算法也使 DBM 失去了很多优点。Montavon 和 Muller[33]提出中心化深度玻尔兹曼机，通过重参数化模型使其在开始学习过程时代价函数的 Hessian 矩阵具有更好的条件数。该模型不需要预训练就能直接训练，并生成高质量的新样本，缺点是分类效果比较差。

Melchior 等人[34]通过实验证实了 Hessian 矩阵条件数的改善，并观察到中心化技巧等价于玻尔兹曼机中的增强梯度。即使在训练困难的情况下，例如训练多层的深度玻尔兹曼机，Hessian 矩阵条件数的改善也能使学习成功。Goodfellow 提出了另一种训练方法：多预测深度玻尔兹曼机[35]，该模型的核心思想是使用反向传播并避免 MCMC 抽样，使模型的分类能力比中心化 DBM 更好，且保持了良好的推断能力，但与基础 DBM 相比该模型不能得到更好的似然函数。

在给定可见单元时 DBM 不能对隐单元进行高效抽样，因此另一类改善 DBM 训练算法的方向是用退火重要性抽样估计对数似然函数的下界。Salakhutdinov[36]提出了基于回火变换 MCMC 因子的 T-SAP 算法和基于自适应 MCMC 算法的耦合自适应模拟回火（Coupled Adaptive Simulated Tempering, CAST）算法。T-SAP 算法采用不同温度的多个链抽样，将样本在原始目标分布与高温分布之间移动，并引入多个中间分布使模型能够产生合理的接受概率。用 CAST 算法训练 DBM 能有效改进参数估计，使该模型能得到更好的多模态能量图，论文中解释了自适应 MCMC 算法和 FPCD 算法的关系。

2.4 应用、分析和小结

在 DBN 出现之初，受限玻尔兹曼机结构的模型受到广泛的关注，并应用于多个领域，除了生成新样本，比较成功的应用还包括目标识别和人体运动行为识别[93-94]、语音识别系统[95-96]、机器翻译[97]以及协同过滤[98]等。随着 GAN 和 VAE 等效果更好、训练效率更高的新型生成模型的出现，RBM 类模型的相关研究逐渐减少，近两年的应用扩展包括数据流缺失识别[99]和汇率预测[100]。

表 1 介绍了以 RBM 为基础模块的重要改进模型及训练算法，并分别介绍了每个模型的改进目的及核心方法。其中最具有代表性的改进模型是与卷积结构结合的卷积深度置信网络 CDBN[101]，该模型利用卷积层提取空间信息，提升模型处理图像样本的能力，是最早引入卷积结构的深度生成模型；另一个有较大影响力的是利用标签信息实现监督学习的条件受限玻尔兹曼机 cRBM[92]，该模型在人体运动系统中成功合成了多种不同的运动序列并实现在线填充运动捕捉过程中丢失的数据。cRBM 不仅是该类模型中最成功的应用之一，也为监督模型提供了新的范式。

与后来出现的具有统治地位的深度生成模型（如变分自编码器和生成对抗网络）相比，RBM 类模型训练效率低、理论复杂，严重限制了 RBM 的理论发展和应用。另外当前优秀的深度生成模型通常都有几十个隐藏层甚至更多，相比之下 RBM 最多只能有三层，这使得模型的适用范围比较窄，很

容易到达性能瓶颈，各种训练算法和模型结构的改进也无法解决这个问题。

从表 1 可以看出学者们对此类模型的研究以 RBM 为主，提出了很多种不同类型的训练方法、模型结构和应用领域，而针对 DBN 和 DBM 的后续研究相对较少，这也从侧面说明了 RBM 对深度学习复兴的贡献以及在后来的发展中因训练效率低下、性能不佳等问题导致多层 RBM 逐渐失去人们的关注。

表 1 基于 RBM 的模型
Table 1 RBM based models

方法名称	改进方式	改进目的	核心方法
rtRBM	训练算法	提高模型性能	改进回火 RBM，加入循环机制
ReLU-RBM	激活函数	改善训练效果	将线性修正单元引入到 RBM 中
3-Order RBM	模型结构	提高模型性能	将可见单元和隐单元分解成三元交互 隐单元控制可见单元协方差和阈值
PGBM	模型结构	结构扩展	在 RBM 中使用门控单元用于特征选择
RBM-SVM	模型结构	提高模型性能	上层 RBM 用于特征提取 下层 SVM 进行回归
RNN-RBM	模型结构	结构扩展	RBM 与循环网络结合
apRBM	模型结构	结构扩展	构造层权重之间的确定性函数
cRBM	模型结构	实现监督学习	将自回归结构和标签信息应用到 RBM
Factored- cRBM	模型结构	提高模型性能	将三元交互方法用在条件 RBM 中
Gaussian-Bernoulli RBM	数据类型	将 RBM 推广到实值	可见单元为参数化高斯分布， 隐藏单元为参数化伯努利分布
mcRBM	模型结构	捕获同层神经元之间的关系	在隐藏层中添加协方差单元 对条件协方差结构建模
ssRBM	模型结构	捕获同层神经元之间的关系	使用辅助实值变量编码条件协方差
mPoT	模型结构	捕获同层神经元之间的关系	添加非零高斯均值的隐变量 条件分布为条件独立的 Gamma 分布
fBMMI-DBN	训练算法	改进预训练算法	用梅尔频率倒谱系数训练DBN产生特征以 预测 HMM 状态上的后验分布
CDBN	模型结构	结构扩展	DBN 与卷积结构结合
3-Order DBN	模型结构	提高模型性能	将三元交互方法用在 DBN 中
fsDBN	训练算法	提高模型性能	用连续判别训练准则优化权值、状态变换 参数和语言模型分数
DBN-HMM	模型结构	提高模型性能	DBN 与隐马尔科夫模型结合
CAST	训练算法	改进训练算法	将自适应算法和 MCMC 结合训练 DBN
Trans-SAP	训练算法	改进训练算法	将回火算法和 MCMC 结合训练 DBN
aiDBM	训练算法	改进训练算法	提出一种近似推断算法，用单独的识别模 型加速 DBN 训练速度
centered DBM	训练算法	改进训练算法	通过重参数化模型使开始学习时 代价函数的 Hessian 具有更好的条件数
MP-DBM	训练算法	改进训练算法	允许反向传播算法，避免 MCMC 估计梯度 带来的训练问题
CDBM	模型结构	结构扩展	DBM 与卷积结构结合

3 变分自编码器

变分自编码器[2]（Variational Auto-Encoder，VAE）是以自编码器结构为基础的深度生成模型。自编码器在降维和特征提取等领域应用广泛，基本结构是通过编码（encoder）过程将样本映射到低维空间的隐变量，然后通过解码（decoder）过程将隐变量还原为重构样本。VAE 模型的基本结构与自编码器相似，两者的区别可以总结为如下三点：

①隐藏变量 z 是随机变量。普通的自编码器没有生成能力，为了使解码过程（生成模型）具有生成能力而不是唯一的映射过程，VAE 假设隐藏变量 z 为服从正态分布的随机变量。

②构造似然函数的变分下界。变分自编码器的目标函数是使输入样本的概率分布和重构样本的概率分布尽量接近，但输入样本的概率是未知的，因此引入建议分布，用变分下界将数据概率分布的数学期望转化为建议分布的数学期望。

③重参数化。变分下界的计算需要在后验分布中抽样，但直接抽样得到的是离散变量，无法进行反向传播。VAE 对编码器输出的均值和方差进行线性变换，解决了训练过程的最后一个障碍。

本节主要介绍 VAE 的模型结构和基本原理，包括变分下界的不同推导方式、重参数化方法以及模型的不足，然后介绍几种有重要意义的扩展模型。

3.1 VAE的模型结构和基本原理

3.1.1 VAE 模型结构

VAE 通过编码过程 $P(z|x)$ 将样本映射为隐藏变量 z，并假设隐藏变量服从于多元正态分布 $P(z)\sim N(0,I)$，从隐藏变量中抽取样本，这种方法可以将似然函数转化为隐藏变量分布下的数学期望：

$$P(x)=\int P(x|z)P(z)\mathrm{d}z \qquad (27)$$

由隐藏变量产生样本的解码过程就是我们需要的生成模型。编码器和解码器可以采用多种结构，现在通常使用 RNN 或 CNN 来处理序列样本或图片样本。

VAE 整体结构图如图 7 所示。为了将样本和重构后的样本进行一一对应，每个样本 x 都必须有其单独对应的后验分布，才能通过生成器将从该后验分布中抽样出来的随机隐藏变量还原成对应的重构样本 $\hat{x}$，每个批次的 n 组样本将由神经网络拟合出 n 组对应的参数以方便用生成器进行样本重构。假设该分布是正态分布，因此 VAE 中存在两个编码器，分别产生样本在隐藏变量空间的均值 $\boldsymbol{\mu}=g_1(x)$ 和方差 $\log\boldsymbol{\sigma}^2=g_2(x)$。

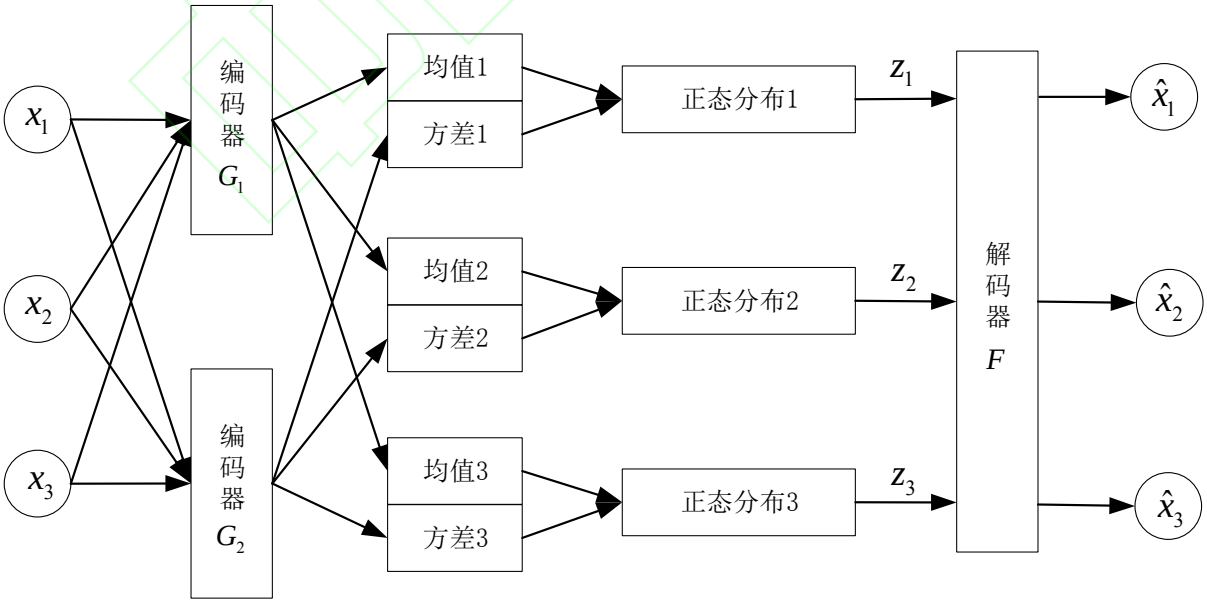

图 7 VAE 结构图

Fig.7 The structure of VAE

3.1.2 变分下界的求法

VAE 的目标函数是数据分布 $P(x)$ 和重构的样本分布 $P(\hat{x})$ 之间距离的最小化，一般用 KL 散度来衡量这两个分布之间的距离：

$$D_{\mathrm{KL}}\left(P(x)\,\|\,P(\hat{x})\right)=\int P(x)\frac{P(x)}{P(\hat{x})}\mathrm{d}x \tag{28}$$

KL 散度的值是非负的，其值越大两个分布之间的距离越远，当且仅当值为 0 时两分布相等。但数据分布的未知性使得 KL 散度无法直接计算，因此 VAE 引入建议分布 $Q(x)$（近似分布）和近似后验分布 $Q(z\,|\,x)$，用极大似然法优化目标函数可得到对数似然函数为：

$$\log P(x)=D_{\mathrm{KL}}(Q(z\,|\,x)\,\|\,P(z\,|\,x))+L(x) \tag{29}$$

根据 KL 散度非负的性质可得 $\log P(x)\geq L(x)$，因而称 $L(x)$ 为似然函数的变分下界。变分下界可以由如下公式推导得到：

$$\begin{aligned}&D_{\mathrm{KL}}(Q(z\,|\,x)\|P(z\,|\,x))\\&=E_{Q(z|x)}(\log Q(z\,|\,x)-\log P(x\,|\,z))\\&=E_{Q(z|x)}(\log Q(z\,|\,x)-\log P(z,x))+\log P(x)\end{aligned} \tag{30}$$

将公式（26）带入到似然函数中可以得到变分下界的最终形式：

$$\begin{aligned}&L(x)\\&=E_{Q(z|x)}(-\log Q(z\,|\,x)+\log P(x,z))\\&=E_{Q(z|x)}(-\log Q(z\,|\,x)+\log P(z)+\log P(x\,|\,z))\\&=-D_{\mathrm{KL}}(Q(z\,|\,x)\,\|\,P(z))+E_{Q(z|x)}(\log P(x\,|\,z))\end{aligned} \tag{31}$$

3.1.3 变分下界的解析解

似然函数的变分下界 $L(x)$ 可以分成两部分理解，第一项 $D_{\mathrm{KL}}(Q(z|x)\,\|\,P(z))$ 是近似后验分布 $Q(z\,|\,x)$ 和先验分布 $P(z)$ 的 KL 散度，两个分布均为高斯分布，因此该项可以写成：

$$\begin{aligned}&D_{\mathrm{KL}}(Q(z|x)\,\|\,P(z))\\&=D_{\mathrm{KL}}(N(\mu,\sigma^2)\,\|\,N(0,1))\\&=\frac{1}{2}\left(-\log\sigma^2+x^2+\sigma^2-1\right)\end{aligned} \tag{32}$$

式（32）中的均值和方差分别为两个编码器的输出值，因此该项可以计算出解析解。由于 $P(z)$ 是标准正态分布，该项可以看成是使 $Q(z\,|\,x)$ 逼近标准正态分布的正则化项，这个正则化项代表 VAE 和普通自编码器的本质区别：自编码器没有生成新样本的能力，为了使 VAE 能够生成样本，需要对编码的结果添加高斯噪声，因此 VAE 假设后验分布为高斯分布，从而使生成器对噪声有鲁棒性；用该项做正则化项，即使 VAE 的第一个编码器得到的均值趋于 0，又能迫使第二个编码器算出带有强度为 1 的噪声，两者的共同作用使得 VAE 的解码器拥有生成能力。

变分下界的第二项是生成模型 $P(x\,|\,z)$，其中 z 服从 $Q(z\,|\,x)$。为了得到该部分的解析解，需要定义生成模型的分布，针对二值样本和实值样本，VAE 分别采用了最简单的伯努利分布和正态分布：

伯努利分布 当生成模型服从于伯努利分布，$P(x\,|\,z)$ 可以表示为 $\prod_{i=1}^{D}y_i^{x_i}(1-y_i)^{1-x_i}$，$y=f(z)$ 表示生成模型的输出，此时 $\log P(x\,|\,z)$ 为：

$$\begin{aligned}&\log P(x\,|\,z)\\&=\sum_{i=1}^{D}x_i\log y_i+(1-x_i)\log(1-y_i)\end{aligned} \tag{33}$$

正态分布 当生成模型服从于正态分布时，$P(x\,|\,z)$ 可以表示成 $N(\mu,\sigma^2\mathrm{I})$，当方差固定为常数 σ^2 时，$\log P(x\,|\,z)$ 可以表示成：

$$\log P(x\,|\,z)\approx-\frac{1}{2\sigma^2}\|x-\mu\|^2 \tag{34}$$

根据式（33）和式（34）可以看出，当样本为二值数据时，用 sigmoid 当作解码器最后一层的激活函数，则变分下界的第二项是交叉熵函数；当样本为实值数据时，变分下界的第二项是均方误差。

3.1.4 重参数化

计算变分下界的第二项时需要从 $Q(z\,|\,x)$ 中抽样，尽管知道该分布是正态分布且其参数已经由编码器计算出来，但直接使用 MCMC 估计近似梯度会产生很大的方差，实际应用中是不可行的[37]，另外抽样操作无法求导，不能用反向传播优化参数，因而 VAE 提出了重参数化方法。

重参数化将该分布中抽样得到的不确定性样本转化成确定性的样本，从简单分布中抽样可以降低抽样的计算复杂性：选择相同概率分布族的 $P(\varepsilon)$，对 $P(\varepsilon)$ 抽样得到的样本 ε 进行若干次线性变换就能获得在原始分布抽样的等价结果。

令 $P(\varepsilon)\sim N(0,1)$，在 $P(\varepsilon)$ 中抽取 L 个样本 ε^i，则 $z^i=\mu+\varepsilon^i\times\sigma$，这种只涉及线性运算的重参数化过程可以用蒙特卡洛方法估计，避免了直接抽样，此时变分下界第二项的估计式可以写成如下形式：

$$\mathrm{E}_{Q(z|x)}(\log P(x\mid z))\simeq\frac{1}{L}\sum_{l=1}^{L}F(\mu+\varepsilon^i\times\sigma) \quad (35)$$

其中 $\varepsilon\sim N(0,1)$。通常取 $L=1$ 就足够精确，这一原理类似于受限玻尔兹曼机的训练方法：每个运行周期抽样出的隐变量都是随机生成的，当运行周期足够多时可以在一定程度上满足抽样的充分性，因此 $L=1$ 就可以满足 VAE 的训练目标。

VAE 的结构可以分为三个阶段，如图 8 所示：

①第一个阶段是编码过程，样本通过两个神经网络分别获得正态分布的均值和方差；

②第二个阶段是重参数化，以便从后验分布中抽样并能够用反向传播训练模型参数；

③第三个阶段是解码过程，将重参数化后的变量通过生成模型生成新样本。

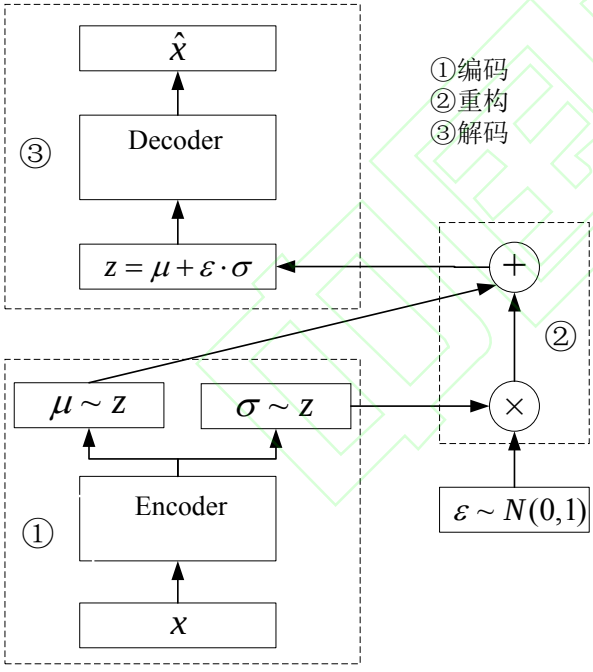

图 8　VAE 训练流程

Fig.8　The training process of VAE

图像样本训练出的 VAE 生成的样本不够清晰，第一个原因是由 KL 散度的固有性质造成的，另一个可能原因是后验分布过于简单，不符合实际学习场景。Theis 等人[38]指出该问题并非 VAE 特有，在使用优化 KL 散度的深度生成式模型中均会发生类似的问题。

3.2　几种重要的VAE结构

3.2.1　重要性加权自编码

VAE 假设后验分布是正态分布只是为了方便计算，并没有考虑这种假设的合理性，必然会影响模型的生成能力。重要性加权自编码[39]（Importance Weighted Autoencoders， IWAE）是 VAE 模型最重要的改进方法之一，IWAE 从变分下界的角度出发，通过弱化变分下界中编码器的作用，一定程度上缓解了后验分布的问题，提高了生成模型的性能。IWAE 将变分下界 $L(x)$ 改写成：

$$\begin{aligned}L(x)&=\mathrm{E}_{Q(z|x)}(-\log Q(z\mid x)+\log P(x,z))\\&=\mathrm{E}_{Q(z|x)}\left[\log\frac{P(x,z)}{Q(z\mid x)}\right]\end{aligned} \quad (36)$$

此时需要在 $Q(z\mid x)$ 中抽样，当抽取 K 个点时式（36）可以写成：

$$L(x)=\mathrm{E}_{Q(z|x)}\left[\log\frac{1}{K}\sum_{i=1}^{K}\frac{Q(x\mid z_i)P(z_i)}{Q(z_i\mid x)}\right] \quad (37)$$

当 $L=1$ 时上式刚好为 VAE 的变分下界，因此可以将 VAE 看成是 IWAE 的一个特例。文章中证明了该式是一个比式（35）更紧的下界，当 L 取较大的值时，模型对 $Q(z\mid x)$ 形式的依赖会趋于弱化，这相当于削弱了后验分布的作用，变相的提高生成模型部分的影响。

IWAE 以降低编码器性能为代价提高生成模型的能力，生成能力明显提高，后来的 VAE 模型大多以 IWAE 为基准，但如果需要同时训练出好的编码器和生成器，IWAE 将不再适用。

3.2.2　监督结构的变分自编码器

VAE 是无监督模型，将标签信息融入到模型中，使 VAE 能够处理监督问题或半监督问题的方法有很多，包括用于监督学习的条件变分自编码器[148-149]（Conditional Variational Auto-Encoder，CVAE）和用于半监督学习的半监督变分自编码器[42,150]（Semisupervised Variational Auto-Encoder，SSVAE）。辅助深度生成模型（Auxiliary Deep Generative Models， ADGM）[40]是效果最好且最有影响力的条件变分自编码器，并同时兼顾监督和半监督学习。ADGM 对标签信息 y 的处理方法与 Kingma 等人[41]提出的将深度生成模型应用于半监督学习的方法类似，都是分别构造有标签数据和无标签数据的似然函数然后求和。ADGM 在 VAE 的

基本结构上使用了辅助变量 a，辅助变量 a 可以用条件概率的形式增加近似后验分布的复杂度，即 $Q(z,a\mid x)=Q(z\mid a,x)Q(a\mid x)$，其结构如图 9 所示。

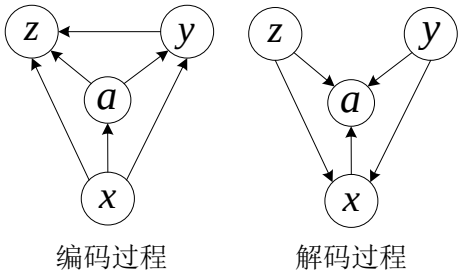

图 9 深度辅助生成模型

Fig.9 Auxiliary deep generative models

ADGM 的编码过程由三个神经网络构成，分别为：

①生成辅助变量 a 的模型 $Q(a\mid x)=N\left(\mu(x),\sigma^2(x)\right)$

②生成标签信息 y 的模型 $Q(y\mid x,a)=\mathrm{Cat}(x,a)$

③生成隐藏变量 z 的模型 $Q(z\mid C)=N(\mu(C),\sigma^2(C))$

其中 $\mathrm{Cat}(\cdot)$ 表示多项分布，$C=x,a,y$ 表示模型输入。解码过程有两个神经网络，分别为：

①重构样本 x 的模型 $P(x\mid z,y)=F(z,y)$

②重构辅助变量 a 的模型 $P(a\mid x,z,y)=F(x,z,y)$

处理无标签样本时，ADGM 的变分下界为：

$$\begin{aligned}&\log P(x)\\&=\log\iiint P(x,y,a,z)dzdyda\\&\geq E_{Q(a,z\mid x,y)}\left[\log\frac{P(x,y,a,z)}{Q(a,y,z\mid x)}\right]\\&=-L(x)\end{aligned}\tag{38}$$

其中 $Q(a,y,z\mid x)=Q(z\mid a,y,x)Q(y\mid z,x)Q(a\mid x)$。处理有标签样本时，变分下界为：

$$\begin{aligned}&\log P(x,y)\\&=\log\iint P(x,y,a,z)dzda\\&\geq E_{Q(a,z\mid x,y)}\left[\log\frac{P(x,y,a,z)}{Q(a,z\mid x,y)}\right]\\&=-L(x,y)\end{aligned}\tag{39}$$

其中 $Q(a,z\mid x,y)=Q(z\mid a,y,x)Q(a\mid x)$。因此 ADGM 的目标函数是无标签样本和有标签样本变分下界之和：$L=L(x)+L(x,y)$。

ADGM 可以用于监督学习或半监督学习，该模型和 IWAE 用不同的方法解决后验分布过于简单的问题，ADGM 的优势是没有削弱编码器，代价是需要 5 个神经网络，计算量更大。用 one-hot 向量表示标签信息可以使 VAE 能够处理监督数据，本质上是在编码器中加入一个条件约束，模型在学习样本时加入标签因素，使 VAE 可以按照指定的标签生成相应类型的样本。

Abbasnejad 等人[42]提出用无限混合（Infinite Mixture）模型处理半监督数据，无限混合模型是指一定数量的带有不同混合系数的 VAE 的混合，模型数量可以由样本自动学习确定，混合系数由狄利克雷过程获得。但这两种方法的效果都不如 ADGM。

3.2.3 卷积变分自编码器

很多深度生成模型想要生成高分辨率的图片样本都会与卷积网络结合。将卷积层融入到 VAE 的模型被称为深度卷积逆图形网络[43]（Deep Convolutional Inverse Graphics Network，DC-IGN）。DC-IGN 的结构与 VAE 相同，只是将 VAE 中的编码器和解码器由原来的全连接网络替换成卷积网络和逆卷积网络，并对模型的部分结构和训练过程做出针对性调整。DC-IGN 确立了在 VAE 结构中使用卷积层的结构框架，这种结构在三维图像生成领域获得了不错的效果。

3.2.4 对抗自编码器

对抗自编码器（Adversarial Autoencoders，AAE）[44]是将对抗的思想应用到 VAE 训练过程的生成模型，两者的主要区别是 VAE 用先验分布和后验分布的 KL 散度约束隐藏变量，AAE 则构造了一个聚合先验分布（伪先验分布）来匹配可以是任意分布的真实后验分布，为了实现对两个分布的匹配，AAE 在隐变量处附加一个对抗网络，其结构如图 10 所示。

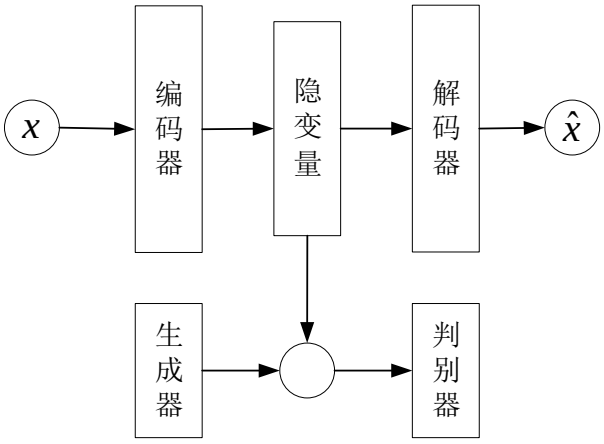

图 10 对抗自编码器

Fig.10 Adversarial Autoencoders

AAE 的生成器采用和编码器相同的神经网络，用来伪造接近真实隐藏变量的分布，判别器负责区分真假分布中得到的样本。

生成器和判别器共同组成了附加的对抗网络，对抗网络的目的是使生成器生成的任意复杂度的分布可以足够接近真实隐藏变量，该部分的目标函数与生成对抗网络的目标函数相同，在 AAE 的整体损失函数中相当于替换了 VAE 中 KL 散度的正则化项。图中展示的 AAE 结构可以用于无监督学习，以该结构为基础，AAE 构造出适用于监督学习、半监督学习和风格迁移等三种不同的模型结构，并在各自的领域内都获得了不错的实验效果，大大增加了 VAE 模型的应用范围。

Zhao 等人[45]从不同的角度得到了更一般的似然函数，该论文认为 VAE 的解码器同样需要约束，为了适当的限制这种灵活性而在目标函数中引入互信息，即 InfoVAE。InfoVAE 给出一个目标函数框架：

$$\begin{aligned} L(x) &= \mathrm{E}_{Q(z|x)}[\log P(x|z)] + \alpha I(x,z) \\ &\quad - \lambda D_{\mathrm{KL}}(Q(z|x)\|P(z)) \end{aligned} \tag{40}$$

其中 $I(x,z)$ 表示可见变量 x 和隐变量 z 之间的互信息，λ,α 是模型参数，不同参数可以得到不同 VAE 的目标函数：

1）$\lambda=1,\alpha=0$ 时该目标函数等价于 VAE；

2）$\lambda=1,\alpha=1$ 且使用 JS（Jensen Shannon）散度，得到对抗自编码 AAE 的目标函数；

3）$\lambda>0$ 且 $\alpha+\lambda=1$ 时，可以得到 β-VAE 的目标函数[46]。

3.2.5 阶梯变分自编码器

采用阶梯结构的阶梯变分自编码器[47]（Ladder Variational Autoencoders， LVAE）逐层、递归的修正隐藏变量的分布，利用多层网络构造更复杂的隐藏变量分布，并在变分下界中使用预热法，这两种方法使得 LVAE 的损失比重要性加权自编码 IWAE 和辅助深度生成模型 ADGM 更低，此后出现的 VAE 论文经常用 LVAE 的结果进行比较。

LVAE 将隐藏变量 z 分割成 L 层，多层隐藏变量的每一层的条件概率分布均为正态分布，因此先验分布可以表示为第一个隐藏层的分布 $P(z_L)$ 乘以前面隐藏层与相邻层之间的条件概率之积，条件概率和第一个隐藏层的分布为

$$\begin{aligned} P(z_i|z_{i+1}) &= N\left(\mu(z_{i+1}),\sigma^2(z_{i+1})\right) \\ P(z_L) &= N(0,\mathrm{I}) \end{aligned} \tag{41}$$

解码器的分布与 VAE 相同，使 $P(x|z)$ 为正态分布或伯努利分布，变分下界采用重要性加权下界以获得更紧的变分下界，并设计了一个可以逐渐改变权重的目标函数：

$$\begin{aligned} &L(x) \\ &= -\beta D_{\mathrm{KL}}(Q(z|x)\|P(z)) + \mathrm{E}_{Q(z|x)}(\log P(x|z)) \end{aligned} \tag{42}$$

其中权衡参数 $\beta=[0\to 1]$ 在模型训练过程中逐渐增加值的大小，即预热法（Warm-Up, WU）。WU 是处理变分函数的一种常用方法，在模型训练的初始阶段设置 $\beta=0$，使模型集中注意力去减少重构误差，然后在训练过程中逐渐增大 β 的值，使目标函数的权重平衡到惩罚项 KL 散度上直到 $\beta=1$，此时该目标函数与 VAE 相同。论文中指出，将 WU 算法直接应用在 VAE 中也能得到损失更低的目标函数。

Cai 等人[48]提出了使用了双层解码器的多阶段变分自编码器（Multi-Stage Variational Auto-Encoders， Multi-Stage VAE）：用原始解码器作为第一阶段生成粗略的样本，然后在模型第二阶段使用超分辨率模型将模糊样本作为输入生成高清样本，使用残差网络使多阶段模型更容易训练。该模型的双层解码器一定程度上克服了生成样本不够清晰的缺点，生成的样本尽管有所改进，但依然比较模糊。

3.2.6 向量量化变分自编码器

向量量化变分自编码器（Vector Quantised Variational Auto-Encoders， VQ-VAE）是首个使用离散化隐藏变量的 VAE 模型。离散表示经常是更高效的表示方式，例如自然语言处理中文字的隐表示通常是离散化形式，图像编码过程的离散表示也可以提高压缩效果，因此 VQ-VAE 意图训练出表示能力更强大的离散变量的先验分布，使模型有能力生成有意义的样本以及扩展 VAE 的应用领域。

VQ-VAE 受到向量量化（Vector Quantization，VQ）方法的启发而提出了新的训练方法：后验概率分布和先验概率分布有明确分类，从这些分类明确的概率分布中提取样本，利用嵌入表示进行索引，得到的嵌入表示输入到解码器中。这种训练方法和有效的离散表达形式共同限制了解码器的学习过程，避免后验崩溃（posterior collapse）现象。

后验崩溃是指当解码器的能力过强时，会迫使编码器学习到无用的隐表示，是 VAE 模型中经常出现的训练问题。

VQ-VAE 结构简单，与适当的先验一起出现时能够在图像、视频、音频样本上生成连续的高质量样本，证明了 VAE 类模型可以应用于无监督对话中，为 VAE 开创了新的应用范例。

3.3 应用、分析和小结

在生成样本方面，VAE 类模型可以生成高清晰度的手写体数字[102]、自然图像[103]和人脸[43,104]等基础数据，并成功生成静态图片的未来预测图片[105]，其中最有影响力的应用是在 VAE 的编码器和解码器中使用循环神经网络 RNN 的 DRAW 网络[103]，DRAW 扩展了 VAE 的结构，并且生成了逼真的门牌号码图片(SVHN 数据集)，是 2016 年出现的效果最好的生成模型之一。DRAW 的作者随后在该模型中加入卷积网络提取空间信息[106]，进一步提高了模型的生成能力，并生成了清晰的自然图像样本。

除了生成图片样本，VAE 还可以在自然语言处理领域生成文本[107-109]、在天文学中模拟对遥远星系的观测[110]、在推荐系统中融合不同信息[111]，在图像合成领域生成不同属性的图像样本[112]以及在化工领域中设计分子的结构[113]等领域均有使用。

作为当前最常用的深度生成模型之一，VAE 由于自身结构的固有缺点使模型生成的图片样本带有大量的噪声，大部分 VAE 结构很难生成高清的图片样本，在图像生成领域的效果不如基于 GAN 和 FLOW 的生成模型，所以在图像领域 VAE 通常被当作特征提取器。但在自然语言处理领域，VAE 类模型生成的语言样本比生成对抗网络更合理，只需要简单的结构就能生成出较流畅的语言，因此更应该在自然语言处理领域寻找 VAE 的优势之处。

VAE 通过编码、重构和解码三个过程完成了由输入样本到隐藏变量的编码过程和隐藏变量到新样本的生成过程。表 2 列举了 VAE 中有影响力的模型并简略介绍了各种模型的核心方法，从表中可以看出，VAE 模型的重要文献中，大部分都是对变分下界的改动，所以学习 VAE 模型的关键是充分理解变分下界的意义。

表 2 重要的 VAE 模型

Table 2 Important VAE models

方法名称	主要贡献	核心方法
CVAE	使 VAE 实现监督学习	在输入数据中加入 one-hot 向量用于表示标签信息
ADGM	提高 CVAE 处理标签信息的能力	在 VAE 中同时引入标签信息和辅助变量 用五个神经网络构造各变量之间的关系
kg-CVAE	提高生成样本的多样性	在 ADGM 上引入额外损失（bag-of-words loss） 使隐变量包含单词出现概率的信息
hybrid-CVAE	用 CVAE 建立鲁棒的结构化预测算法	输入中加入噪声、使用随机前馈推断 构造带有随机高斯网络的混合变分下界： $L(x)=\alpha L_{\mathrm{CVAE}}+(1-\alpha)L_{\mathrm{GSNN}}$
SSVAE	使 VAE 实现半监督学习	构造两个模型：M2 为半监督模型 M1 模型为 VAE 用于提升 M2 的能力
IMVAE	提高 SSVAE 处理混合信息的能力	用非参数贝叶斯方法构造无限混合模型 混合系数由 Dirichlet 过程获得
AAE	使模型可以学习出后验分布	构造聚合的伪先验分布匹配真实分布 在隐变量处附加一个对抗网络学习伪先验分布
ARAE	使 AAE 能够处理离散结构	编码器和解码器采用循环神经网络里 变分下界中添加额外的正则项
IWAE	使后验分布的假设更符合真实后验分布	构造比 VAE 更紧的变分下界形式，通过弱化变分下界中编码器的作用提升变分推断的能力
DC-IGN	保留图片样本中的局部相关性	用卷积层和池化层替代原来的全连接网络

infoVAE	提高隐变量和可观测变量之间的互信息，使近似后验更逼近真实后验分布	在变分下界中引入互信息： $\alpha I_q(x)$
β-VAE	从原始数据中获取解开纠缠的可解释隐表示	在变分下界中添加正则系数： $L(x)=\mathrm{E}_{Q(z\|x)}(\log P(x\|z))-\beta D_{\mathrm{KL}}(Q(z\|x)\|P(z))$
β-TCVAE	解释 β-VAE 能够解开纠缠的原因并提升模型性能	在 β-VAE 变分下界中引入互信息和额外正则项： $-I_q(z)$ 和 $-D_{\mathrm{KL}}(Q(x)\|P(x))$
HFVAE	使 VAE 对离散变量解开纠缠 总结主流 VAE 的变分下界	对变分下界分解成四项并逐一解释作用： $L(x)=\mathrm{E}_{Q(z\|x)}[\log(P(x\|z)/P(x))-\log(Q(z\|x)/Q(z))]$ $-D_{\mathrm{KL}}(Q(z)\|P(z))-D_{\mathrm{KL}}(Q(x)\|P(z))$
DRAM	处理时间序列样本	在 VAE 框架中引入注意力机制和 长短时记忆网络结构
MMD-VAE	用最大平均差异替换 KL 散度	将变分下界中的 KL 散度项替换成： $D_{\mathrm{MMD}}(Q(x)\|P(x))$
HVI	使用精度更高的抽样法替代重参数方法	用 Hamiltonian Monte Carlo 抽样替换重参数化方法直接对后验分布抽样以获得更精确的后验近似
VFAE	学习敏感或异常数据时使隐变量保留更多的信息	在变分下界中附加基于最大平均差异的惩罚项： $\sqrt{2/D}\cos(\sqrt{2/r}xW+b)$
LVAE	逐层、递归的修正隐变量的分布，使变分下界更紧	利用多层的隐变量逐层构造更复杂的分布 在变分下界中使用预热法
wd-VAE	解决输入缺失词情况下的语言生成	将输入文本转换成 UNK 格式并进行 dropout 操作 使解码器的 RNN 更依赖隐变量表示
VLAE	用流模型学习出更准确的后验分布	用流模型学习的后验分布替代高斯分布， 根据循环网络学到的全局表示抛弃无关信息
PixelVAE	捕获样本元素间的关系以生成更清晰锐利的图片样本	将隐变量转成卷积结构，解码器使用 PixelCNN CNN 只需要很少几层，压缩了计算量
DCVAE	通过调整卷积核的宽度改善解码器理解编码器信息的能力	在解码器中使用扩张卷积加大感受野 对上下文容量与有效的编码信息进行权衡
MSVAE	用双层解码器提高模型生成高清图像的能力	第一层解码器生成粗略的样本 第二层解码器使用残差方法和跳跃连接的超分模型 将模糊样本作为输入生成高清样本

4 生成对抗网络

生成对抗网络（Generative Adversarial Nets, GANs）[3]是当前机器学习领域最热门的研究方向，在图像生成领域占有绝对优势。GAN 本质上是将难以求解的似然函数转化成神经网络，让模型自己训练出合适的参数拟合似然函数，这个神经网络就是 GAN 中的判别器。

GAN 内部对抗的结构可以看成是一个训练框架，原理上可以训练任意的生成模型，通过两类模型之间的对抗行为来优化模型参数，巧妙的避开求解似然函数的过程。这个优势使 GAN 具有很强的适用性和可塑性，可以根据不同的需求改变生成器

和和判别器，尽管模型本身具有很多训练上的难点，但随着各种解决方法的出现，逐渐解决了模型的训练问题，使 GAN 受到很多关注，几年内就出现数以千计的相关论文。

GAN 原理简单易理解，生成的图片清晰度和分辨率超过其他生成模型，缺点是训练不稳定，因此人们最关注的是模型的生成效果和训练稳定性两方面。本节首先介绍 GAN 的基本原理和模型结构，然后介绍基于 WGAN 的稳定性研究和 GAN 模型框架的发展。

4.1 GAN模型

4.1.1 GAN 的基本原理

GAN 中的博弈方是一个生成器和一个判别器，生成器的目标是生成逼真的伪样本让判别器无法判别出真伪，判别器的目标是正确区分数据是真实样本还是来自生成器的伪样本，在博弈的过程中，两个竞争者需要不断优化自身的生成能力和判别能力，而博弈的结果是找到两者之间的纳什均衡，当判别器的识别能力达到一定程度却无法正确判断数据来源时，就获得了一个学习到真实数据分布的生成器，GAN 的模型结构如图 11 所示。

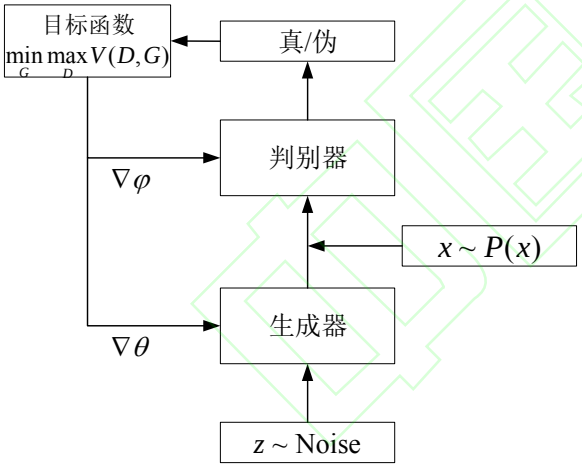

图 11 GAN 模型结构

Fig.11 The structure of GANs

GAN 中的生成器和判别器可以是任意可微函数，通常用多层的神经网络表示。生成器 $G(z;\theta)$ 是输入为随机噪声、输出伪样本、参数为 θ 的网络，判别器 $D(x;\varphi)$ 是输入为真实样本和伪样本、输出为 0 或 1（分别对应伪样本和真实样本）、参数为 φ 的二分类网络。GAN 根据生成器和判别器不同的损失函数分别优化生成器和判别器的参数，避免了计算似然函数的过程。

4.1.2 训练方法

GAN 的训练机制由生成器优化和判别器优化两部分构成，下面分析两者的目标函数和优化过程。

优化判别器 固定生成器 $G(z;\theta)$ 后优化判别器 $D(x;\varphi)$，由于判别器是二分类模型，目标函数选用交叉熵函数：

$$\max_D V(D) = \mathrm{E}_{x\sim P_r}[\log D(x)] + \mathrm{E}_{x\sim P_g}[\log(1-D(x))] \quad (43)$$

其中 P_r 是真实样本分布，P_g 表示由生成器产生的样本分布。判别器的目标是正确分辨出所有样本的真伪，该目标函数由两部分组成：

1) 对于所有的真实样本，判别器应该将其判定为真样本使输出 $D(x)$ 趋近 1，即最大化 $\mathrm{E}_{x\sim P_r}[\log D(x)]$；

2) 对于生成器伪造的所有假样本，判别器应该将其判定为假样本使输出尽量接近 0，即最大化 $\mathrm{E}_{x\sim P_g}[\log(1-D(x))]$。

优化生成器 固定训练好的判别器参数，考虑优化生成器模型参数。生成器希望学习到真实样本分布，因此优化目的是生成的样本可以让判别器误判为 1，即最大化 $\mathrm{E}_{x\sim P_g}[\log(D(x))]$，所有生成器的目标函数为：

$$\min_G V(G)\mathrm{E}_{x\sim P_g}[\log(1-D(x))] \quad (44)$$

后来又提出了一个改进的函数为：

$$\min_G V(G)\mathrm{E}_{x\sim P_g}[-\log D(x)] \quad (45)$$

从该目标函数可以看出，生成器的梯度更新信息来自判别器的结果而不是来自数据样本，相当于用神经网络拟合出数据分布和模型分布之间的距离，从根本上回避了似然函数的难点，这一思想对深度学习领域产生了深远的影响，也是 GAN 模型的优势。

固定生成器参数，根据判别器目标函数可得到：

$$-P_r(x)\log D(x) - P_g(x)\log[1-D(x)] \quad (46)$$

令式（46）对 $D(x)$ 的导数为 0 可以得到判别器最优解的表达式：

$$D^*(x)=\frac{P_r(x)}{P_r(x)+P_g(x)} \quad (47)$$

然后固定最优判别器 D^* 参数，训练好的生成器参数就是最优生成器。此时 $P_r(x)=P_g(x)$，判别器认为该样本是真样本还是假样本的概率均为 0.5，说明此时的生成器可以生成足够逼真的样本。

4.1.3 GAN 存在的问题

GAN 模型刚提出时存在很多严重缺陷，效果也不突出，从模型结构到稳定性、收敛性等都处于探索阶段，导致一段时间内没有展现出应有的能力。GAN 模型的不足之处可以总结为以下几点：

①模型难以训练，经常出现梯度消失导致模型无法继续训练；生成器形式过于自由，训练时梯度波动极大造成训练不稳定；需要小心地平衡生成器和判别器的训练程度，使用更新一次判别器后更新 k 次生成器的交替训练法并不能很好的缓解训练问题；

②出现模式崩溃（model collapse），具体表现为生成样本单一，无法生成其它类别的样本；

③目标函数的形式导致模型在训练过程没有任何可以指示训练进度的指标。

Arjovsky 在论文[49]中用一系列的公式推导证明了 GAN 出现各种问题的原因并给出了初步的解决方案。论文分析了生成器目标函数的两种形式，对式（44）进行恒等变形使之成为如下形式：

$$\mathrm{E}_{x\sim P_r}[\log D(x)]+\mathrm{E}_{x\sim P_g}[\log(1-D(x))] \quad (48)$$

将其带入到最优判别器式（47）中后得到的最终结果为：

$$V(G)=2D_{\mathrm{JS}}(P_r\,\|\,P_g)-2\log 2 \quad (49)$$

根据 GAN 定义的判别器目标函数的最优解，可以将生成器等价为真实分布和生成分布的 JS 散度，所以优化生成器时相当于最小化两个分布之间的距离。但 JS 散度的定义导致当两个分布之间没有任何重合部分时，JS 散度的值将固定是常数 $\log 2$，这就会产生梯度消失的现象。

论文中指出，当两个分布的支撑集是高维空间中的低维流形时，两者之间没有重叠部分的概率是 1。如图片样本等高维数据，其分布只是高维空间中的一个低维流形，两个不同分布之间几乎不存在重叠，导致生成器目标函数的导数为 0 从而出现梯度消失的现象。

根据 Huszar[50]给出的推导可将两个分布之间的 KL 散度变换为含最优判别器 D^* 的形式：

$$\begin{aligned}&D_{\mathrm{KL}}(P_g\,\|\,P_r)\\&=\mathrm{E}_{x\sim P_g}\log[1-D^*(x)]-\mathrm{E}_{x\sim P_g}[\log D^*(x)]\end{aligned} \quad (50)$$

对式（45）进行等价变换，得到的最终结果表示为：

$$V(G)=D_{\mathrm{KL}}(P_g\,\|\,P_r)-2D_{\mathrm{JS}}(P_r\,\|\,P_g)+c \quad (51)$$

其中 $c=2\log 2+\mathrm{E}_{x\sim P_r}[\log D^*(x)]$ 是常数。生成器的改进目标函数可以变换为两个分布的 KL 散度减去两个分布的 JS 散度，说明该函数在优化过程中一边拉近两者的 KL 散度，一边增大 JS 散度，优化中的矛盾造成了梯度的不稳定。

KL 散度的不对称性使生成的样本缺乏多样性和准确性，这种惩罚程度的巨大差别使生成器无限趋向于生成准确样本而不去生成多样性的样本，以免产生巨大的惩罚，所以生成器目标函数中的 KL 散度项是造成模型模式崩溃的原因。

4.2 基于WGAN的稳定性研究

WGAN[51]（Wasserstein GAN）从理论上分析了原始 GAN 存在的缺陷，提出用 Wasserstein 距离替代 KL 散度和 JS 散度，改变了生成器和判别器的目标函数，并对判别器施加 Lipschitz 约束以限制判别器的梯度，后来对 GAN 稳定性研究的很多论文都以 Lipschitz 约束为出发点。WGAN 只用几处微小的改动就解决了 GAN 不稳定的问题，基本消除了简单数据集上的模型崩溃问题，是 GAN 最重要的改进论文之一。

4.2.1 Wasserstein 距离的优点

fGAN[52]以局部变分法为理论基础证明了任何散度都可以应用到 GAN 的结构中，并给出了用一般化的 f 散度构造 GAN 的统一方法。GAN 的缺点源自 KL 散度和 JS 散度本身性质的问题，因此 WGAN 提出用 Wasserstein 距离替代 KL 散度和 JS 散度，解决两种散度造成的训练不稳定问题。Wasserstein 距离又被称为推土机距离（Earth Mover， EM），最初用于解决最优运输问题，可以解释为将分布 P_r 沿某个规划路径转移到分布 P_g 需要的最小消耗，该问题对应的最优化问题可以表示成如下形式：

$$D_{\mathrm{W}}(P_r \| P_g) = \inf_{\gamma \sim \prod(P_r, P_g)} \mathrm{E}_{(x,y)\sim\gamma}\left[c(x,y)\right] \quad (52)$$

其中 x, y 分别表示付出分布 P_r 和 P_g 的样本，$c(x,y)$ 表示输运成本，$\prod(P_r, P_g)$ 表示两个分布的所有联合分布集合，该集合中的任意分布的边缘分布均为 P_r 和 P_g，即 $\sum_x \gamma(x,y) = P_r(y)$、$\sum_y \gamma(x,y) = P_g(x)$。对于任意一个可能的联合分布 $\gamma(x,y)$，从分布中抽样得到真实样本 x 和生成样本 y，这两个样本之间的距离在联合分布期望下的值为 $\mathrm{E}_{(x,y)\sim\gamma}\left[\| x - y \|\right]$，在所有联合分布中选取对该期望值能够取到下界的分布，该下界就定义为 Wasserstein 距离。

如果高维空间中的两个分布之间没有重叠，KL 散度和 JS 散度无法正确反映出分布的距离，也就不能为模型提供梯度，但 Wasserstein 距离可以准确的反映这两个分布的距离，从而提供更可靠的梯度信息，避免训练过程中出现梯度消失和不稳定等现象。

4.2.2 WGAN 的判别器形式

Wasserstein 距离很难直接优化，但可以将式（52）变换成对偶问题：

$$\begin{aligned} & K \cdot D_{\mathrm{W}}(P_r, P_g) \\ = & \max_{w:|F|_L \le K} \mathrm{E}_{x\sim P_r}\left[F(x)\right] - \mathrm{E}_{x\sim P_g}\left[F(x)\right] \end{aligned} \quad (53)$$

式（53）相当于在连续函数 $F(\cdot)$ 上施加 Lipschitz 常数为 K 的约束，使得定义域内任意两个元素 x_1 和 x_2 满足：

$$\left|F(x_1) - F(x_2)\right| \le K\left|x_1 - x_2\right| \quad (54)$$

这就将 Wasserstein 距离转化成求解满足 Lipschitz 约束的所有函数条件下两个分布期望之差的上界。WGAN 对判别器施加 Lipschitz 约束的具体方法是限制网络中的权重，控制所有权重的绝对值不超过固定常数，否则对参数进行截断，这种方法叫做权重裁剪（weight clipping）。判别器通过权重裁剪后得到：

$$L = \mathrm{E}_{x\sim P_g}\left[F(x)\right] - \mathrm{E}_{x\sim P_r}\left[F(x)\right] \quad (55)$$

该式可以指示模型的训练进度，真实分布与生成分布的 Wasserstein 距离越小，说明 GAN 中生成器的生成能力越好。根据式（55）可以得到 WGAN 中生成器和判别器的目标函数分别为：

$$\begin{aligned} V(G) &= \min -\mathrm{E}_{x\sim P_g}\left[F(x)\right] \\ V(D) &= \min \mathrm{E}_{x\sim P_g}\left[F(x)\right] - \mathrm{E}_{x\sim P_r}\left[F(x)\right] \end{aligned} \quad (56)$$

通过对判别器参数设置阈值保证判别器对差别微小的样本不会得到差别过大的值，满足对距离基本的稳定性要求，避免判别器在训练时产生不稳定现象。判别器的输出是真实分布和生成分布之间的 Wasserstein 距离的近似值而不再是 GAN 的二分类任务，因此去掉判别器最后一层的激活函数。

4.2.3 梯度惩罚

WGAN 在训练过程中会出现收敛速度慢、梯度消失或梯度爆炸等现象，原因在于对判别器施加 Lipschitz 约束的方式不合理。实验发现权重裁剪会使判别器所有参数趋于极端，全部集中在阈值的最大值和最小值这两个点上，这使得判别器退化成一个二值神经网络，严重影响了判别器的能力；阈值的调参过程比较困难，经常出现梯度爆炸或梯度消失的现象。

WGAN-GP[53]直接将判别器的梯度作为正则项加入到判别器的损失函数中，该正则项通过惩罚梯度使判别器梯度在充分训练后达到 Lipschitz 常数 K 附近，因此该正则项被称为梯度惩罚（Gradient penalty）。加入梯度惩罚的判别器的目标函数为：

$$\begin{aligned} L = & \mathrm{E}_{x\sim P_g}\left[F(x)\right] - \mathrm{E}_{x\sim P_r}\left[F(x)\right] \\ & + \lambda \mathrm{E}_{x\sim P_{\hat{x}}}\left[\left(\| \nabla_x D(x) \|_2 - 1\right)^2\right] \end{aligned} \quad (57)$$

其中 Lipschitz 常数 K 取 1，$P_{\hat{x}}$ 表示整个样本空间的概率分布。梯度惩罚要求梯度在整个样本空间内都满足 $\|\nabla_x D(x)\|_2 \le 1$，这种约束条件难以做到，所以采用真假样本以及两者之间的随机插值的方法，使该约束能近似地遍布真实样本和生成样本之间的所有空间，即：

$$\begin{aligned} L = & \mathrm{E}_{x\sim P_g}\left[F(x)\right] - \mathrm{E}_{x\sim P_r}\left[F(x)\right] \\ & + \frac{\lambda}{N}\sum_{i=1}^{N}\left(\| \nabla_x D(x) \|_{x=\varepsilon_i x_r + (1-\varepsilon_i) x_g} - 1\right)^2 \end{aligned} \quad (58)$$

其中 ε 是服从 U[0,1] 的随机数，x_r 和 x_g 分别表示真

实样本和生成样本。梯度惩罚只对真假样本集中区域以及两者之间的区域生效，就能够很好的控制梯度，使 WGAN-GP 避免出现梯度消失或梯度爆炸，显著提高训练速度和收敛速度，可以训练多种不同种类的网络结构，首次实现了 GAN 模型的无监督文本生成。

4.2.4 谱归一化生成对抗网络

梯度惩罚的缺点是惩罚只能在局部生效，如果样本类别较多，随机插值方法会导致判别器的约束失效。谱归一化（Spectral Normalization）方法则将判别器中的所有参数都替换为$W \to W/\|W\|_2$，如果激活函数导数的绝对值都小于等于某个常数，就能保证判别器满足 Lipschitz 约束。用这种更精确的方法实现$|F|_L \le K$约束的模型叫谱归一化生成对抗网络[54]（Spectral Normalization for GAN，SNGAN），该模型实现方法简单，只需把谱范数的平方作为正则化项，填加到判别器的目标函数中，此时该目标函数可以表示为：

$$L = V(D) + \lambda \|W\|_2^2 \qquad (59)$$

SNGAN 的收敛速度比 WGAN-GP 更快，且效果更好，是目前对模型施加 Lipschitz 约束的最好方法。

4.3 GAN框架结构发展

4.3.1 基于卷积层的结构

深度卷积生成对抗网络（Deep Convolutional Generative Adversarial Networks， DCGAN）[55]是 GAN 的第一个重要改进，在多种结构中筛选出效果最好的一组生成器和判别器，使 GAN 训练时的稳定性明显提高，至今仍然是常用的架构。正因为 DCGAN 的出现，让人们不必过多纠结模型的结构，而是把注意力放在综合性的任务上，使 DCGAN 迅速应用到图像生成、风格迁移和监督任务等多个领域。

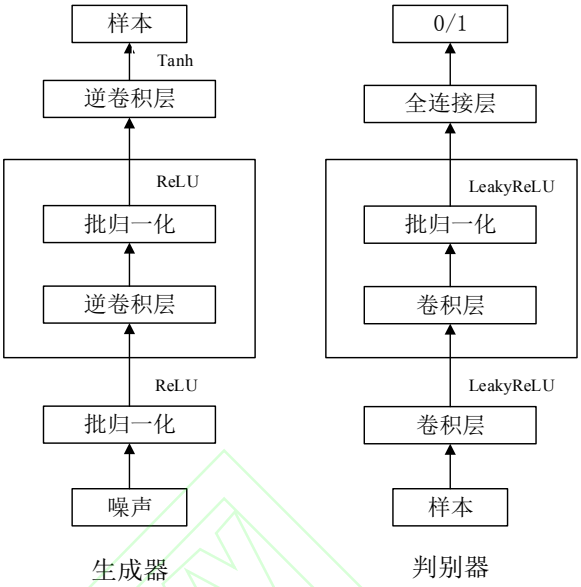

图 12 DCGAN 结构

Fig.12 The structure of DCGANs

DCGAN 的结构如图 12 所示。模型架构最主要的特点是判别器和生成器采用卷积网络和反卷积网络，各层均使用批归一化。DCGAN 训练速度很快，内存占用量小，是快速实验最常用的结构，缺点是生成器中的反卷积结构存在固有的棋盘效应（checkerboard artifacts），具体表现为图片放大之后能看到如象棋棋盘一样的交错纹理，严重影响生成图片的质量，限制了 DCGAN 结构的重构能力。

基于卷积层的最新生成对抗网络是针对单个自然图像的 SinGAN[91]，该模型按照不同比例的下采样将单个图像样本分割成多个不同尺度的子样本，然后使用多个 DCGAN 组成金字塔结构，各个 DCGAN 负责学习不同子样本的数据概率分布，这种结构使模型能够生成给定图像中相同视觉内容的高质量且具有任意大小和比例的新图像。

4.3.2 基于残差网络的结构

生成器中的反卷积结构在图像上的映射区域大小有限，使得 DCGAN 难以生成高分辨率的图像样本，而渐进式增长生成对抗网络[56]（Progressive Growing of GAN, PGGAN）等生成能力突出的新一代结构均选用残差网络作为生成器和判别器，其基本结构如图 13 所示。基于 ResNet 的 GAN 模型的主要特点为判别器使用了残差结构，生成器用上抽样替代反卷积层，判别器和生成器的深度都大幅度增加。

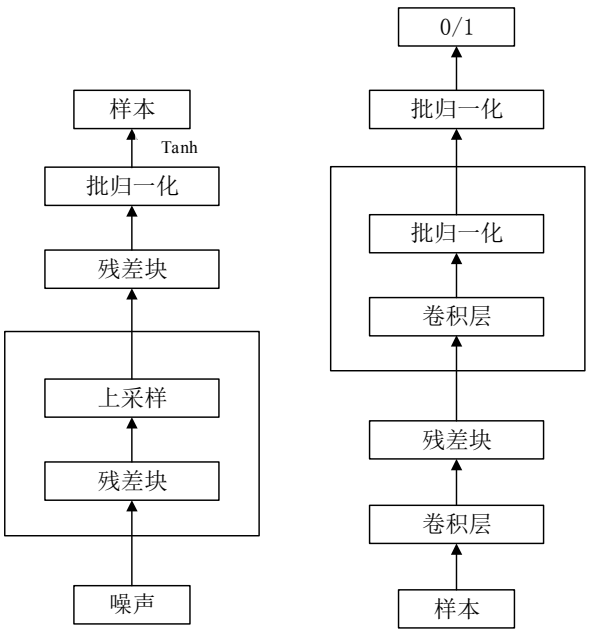

图 13 ResNet-GAN 结构

Fig.13 The structure of ResNet-GANs

基于残差结构框架的 BigGAN[57]是当前图像生成领域效果最好的模型，生成高清样本的逼真程度大幅度领先其它生成模型。BigGAN 对图像细节处理的很好，能生成非常逼真的 512×512 自然场景图像，实现了大规模和稳定性的较大提升与平衡，论文后续将模型深度增加四倍得到 BigGAN-deep 模型，改进了分层隐藏空间技术，使模型性能进一步提升。

BigGAN 模型的缺陷是需要大量的标注数据才能训练，在 BigGAN 的判别器中增加一个额外的无监督任务的模型 S^3GAN 能够为无标签样本添加标签，从而增加大量的训练数据，使 S^3GAN 用 10%的标签数据就能够匹配 BigGAN 生成的样本质量。

4.3.3 监督结构和半监督结构

为了将样本与标签信息结合，条件生成对抗网络[58]（Condition GAN， CGAN）将标签信息作为附加信息输入到生成器中，再与生成样本一起输入到判别器中：生成器同时接收噪音和标签信息，目的是让生成的样本能尽量符合标签信息；判别器输入标签信息和真伪样本，同时进行两次判断，一是判断输入样本的真伪，二是该样本与标签信息是否匹配，最后输出样本真伪和标签信息预测值。

另一种常用结构是辅助分类器生成对抗网络[59]（Auxiliary Classifier GAN, ACGAN），ACGAN 的生成器同 CGAN 相同，但判别器只输入真伪样本，输出样本真伪和标签信息预测值，因此判别器的额外输出需要设置关于标签信息的损失函数。从实验的结果来看，两种结构处理监督数据的性能相似，但 ACGAN 的结构更适合处理半监督数据。两种网络结构如图 14 所示。

为了充分利用无标签数据，半监督条件生成对抗网络（Semi-supervised CGAN, SSGAN）[60]在 CGAN 结构基础上设计了两个不同功能的判别器：第一个判别器将去除了标签信息的有标签数据和无标签数据一同作为输入，输出数据真伪概率的同时将数据的中间变量传输给第二个判别器；第二个判别器根据中间变量及其相应的标签信息判断联合概率分布的真伪。

StackGAN[61]是基于 CGAN 的改进模型，解决了无法生成高分辨率图片样本的缺点，方法是搭建两个生成器，并将训练过程分为两个阶段：第一阶段采用标准的 CGAN，输入噪声和对应的文本标注信息，然后生成一个低分辨率的样本；第二阶段的生成器将第一阶段生成的低分辨率图像与处理过的标注信息连接起来作为输入，以便生成高清样本。StackGAN++[62]在此基础上使用树状结构生成器，引入颜色正则化约束图片的色彩信息，进一步提高训练稳定性和生成样本的质量。

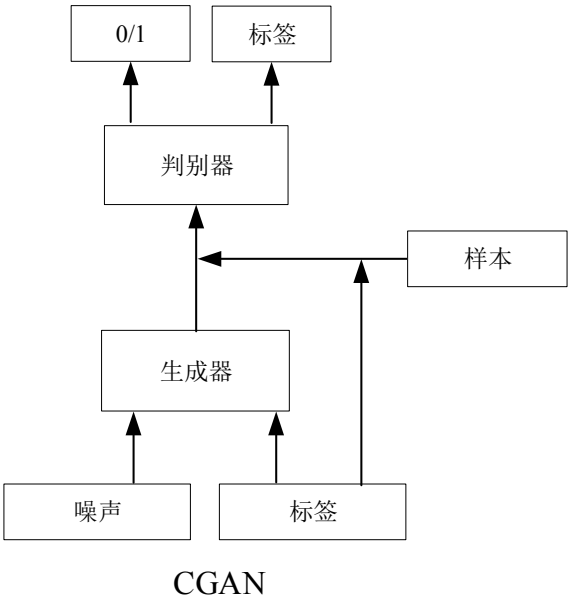

图 14 CGAN 和 ACGAN 结构

Fig.14 The structure of CGANs and ACGANs

4.4 应用、分析与总结

4.4.1 应用

GAN 模型种类繁多，应用广泛，其中最成功的应用是图像处理和计算机视觉，并以人体合成和人脸生成的发展尤为迅速：解开纠缠的隐表示生成对抗网络[114]（Disentangled representation GAN, DRGAN）可以根据任意角度的人脸样本输入，输出任意目标角度的合成人脸及身份信息；双通道生成对抗网络[115]（TPGAN）通过同时学习局部细节和全局感知实现了写实主义风格的人脸合成；姿态生成网络[116]（PG^2）根据人体图像和任意姿态合成该姿态下的人体图像，这一应用与变结构式生成对抗网络[117]的效果类似；GAN 还可以利用监督学习的方法实现风格转换，根据标签信息生成相应的风格图像样本[118-119]；相比上述需要多种类标签信息的监督模型，应用于无监督的人脸生成上的 GAN 结构更简单，相关研究很多[120-122]，目前已经可以生成1024×1024 的逼真人脸图像。

SRGAN[123]是第一个应用于图像超分辨率领域的 GAN 框架，使 GAN 能够提高自然场景图片的分辨率，ESRGSN[124]对该模型进行了改进，增强了模型性能；CycleGAN[125]和其改进模型[126-127]利用循环损失函数和两组生成器—判别器将图像转化为另一个风格；GAN 可用于目标检测，perceptual GAN[128]和 MTGAN[129]分别适用于小目标检测和多目标检测，SeGAN[130]可以用于对图像的目标检测。

文献[131]使用 GAN 框架首次实现了视频生成，但视频的像素和清晰度都非常低，连贯性也比较差，随后的相关研究逐渐的提升了生成视频的清晰度和连贯性[132-133]，文献[134]则用 GAN 模型作视频预测。GAN 在音频领域也有广泛应用，例如音乐生成[135]、语音的合成[136]和识别[137]等；GAN 在自然语言处理领域也有应用，例如信息检索[138]、文本生成[139]以及用文本生成图像[140]等；另外 GAN 在医学[141-143]和数据科学[144-145]等领域也在最近两年内得到关注和大量应用。

4.4.2 分析和小结

因为减少计算量等数学因素,GAN 以外的其他几种主流深度生成模型都假设隐藏变量服从高斯分布或平均分布等基础分布，这种不符合实际的假设，限制了模型的适用范围。这一局限性正是 GAN 最大的优势，因为 GAN 模型不需要假设隐藏变量服从的概率分布和数据的概率分布就能够生成清晰度很高的样本，使其在图像处理和计算机视觉等领域占有绝对优势。在图像生成领域，模型的优劣主要以能生成的最高分辨率为准，如今的 GAN 从最早的 28×28 灰度手写体图像发展到 512×512 的自然场景图像生成，生成样本清晰度很高且富有多样性，用于评价生成样本逼真度的常用指标 Inception Score[146]已经可以接近真实图片。

GAN 模型是深度生成模型中最大的分支，目前已经有数千篇的研究论文，表 3 列举了部分改进模型，其中最有影响力的模型包括：InfoGAN[88]将生成器的输入噪声分解为一个噪声和表示真实数据

分布的结构化语义特征，用于表示生成数据的不同特征维度，并使用基于互信息的正则化项用于约束生成器，使 GAN 能够学习解开纠缠的引表示；PGGAN[56]可以随着训练的进行逐渐增加层数，在训练好的浅层网络权重的基础上逐渐加深网络的深度，使模型能够学习更高分辨率的图像；自注意力生成对抗网络[63]（Self-Attention Generative Adversarial, SAGAN）将注意力机制引入到 GAN 结构中，使模型能够更好的处理图像中大范围、多层次的依赖关系，并在生成器中应用谱归一化方法作为正则项约束生成器参数。

在可预见的未来，GAN 模型必将是图像生成领域内最具代表性、最成功的深度生成模型。很多与 GAN 结合的生成式模型应用在自然语言处理中，例如 VAE 与 GAN 的结合、利用强化学习的 SeqGAN[135]等。这些模型的提出不仅扩展了 GAN 的应用领域，而且提高了 GAN 理论上的完备程度。

表 3 重要的 GAN 模型

Table 3 Important GANs

模型名称	核心方法	生成图片类型	生成最高分辨率
CGAN	将标签信息作为附加信息输入到生成器中 再与生成样本一起输入到判别器中	MNIST	28×28
DCGAN	在多种结构中筛选出最优的一组生成器和判别器 生成器和判别器均使用深度卷积网络	LSUN FACES ImageNet-1k	32×32
VAE-GAN	在 VAE 结构外嵌套 GAN 的框架，用 GAN 中的判别器学习 VAE 的两个分布间的相似程度	CelebA LFW	64×64
BiGAN	生成器是输入输出不相关的编码器和解码器 判别器同时输入样本和隐变量判断两者来自编码器还是解码器	MNIST ImageNet	64×64
CoGAN	在实现风格转换学习时，为了让两个编码器的输出尽量接近，共享两者的最后几层参数	MNIST CelebA	64×64
Info-GAN	将噪声 z 拆分成子向量 c 和 z' 子向量 c 用于调节输出的类别和形状等条件信息 用额外的判别器判定生成样本的子向量 c	MNIST SVHN	64×64
LSGAN	使用最小二乘损失函数 最小二乘可以将图像的分布尽可能接近决策边界	LSUN HWDB	64×64
WGAN	从理论上分析 GAN 训练不稳定的原因 通过使用 Wasserstein 距离等方法提高了训练稳定性	LSUN	64×64
f-GAN	证明了任意散度都适用于 GAN 框架	MNIST LSUN	96×96
LAPGAN	基于拉普拉斯金字塔结构逐层增加样本分辨率 上层高分图像的生成以下层低分图像为条件	CIFAR10 LSUN STL	96×96
WGAN-GP	将判别器的梯度作为正则项 加入到判别器的损失函数中	ImageNet CIFAR10 LSUN	128×128
SNGAN	使用谱归一化代替梯度惩罚	CIFAR10 STL10 ImageNet	128×128
Improved-DCGAN	使用多种方法对 DCGAN 的稳定性和生成效果进一步加强	MNIST CIFAR10	128×128

		SVHN ImageNet	
EBGAN	将判别器的功能改为鉴别输入图像重构性的高低， 生成器可以在刚开始训练时获得较大的能力驱动 （energy based）并在短期内获得效果不错的生成器	MNIST LSUN CelebA ImageNet	128×128
BEGAN	判别器为自编码结构，用于估计分布之间的误差分布 提出使用权衡样本多样性和质量的超参数	CelebA	128×128
ACGAN	每个样本都有类标签 类标签同时输入到生成器和判别器中	ImageNet CIFAR10	128×128
SAGAN	用自注意力机制代替卷积层进行特征提取	ImageNet	128×128
SRGAN	生成器用低分图像生成高分图像 判别器判断图像是生成器生成的还是真实图像		
StackGAN	第一阶段使用 CGAN 生成 64×64 的低分图像 第二阶段以低分图像和文本为输入，用另一个 GAN 生成高分图像	CUB Oxford-102 COCO	256×256
StackGAN++	在 StackGAN 的基础上用多个生成器生成不同尺度 的图像，每个尺度有相应的判别器 引入非条件损失和色彩正则化项	CUB Oxford-102 COCO	256×256
Cycle-GAN	由两个对称的 GAN 构成的环形网络 两个 GAN 共享两个生成器，各自使用单独的判别器	Cityscapes label	256×256
Star-GAN	为了实现多个领域的转换引入域的控制信息 判别器需要额外判断真实样本来自哪个域	CelebA RaFD	256×256
BigGAN	训练时增加批次数量和通道数 让权重矩阵为正交矩阵，降低权重系数的相互干扰	ImageNet JFT-300M	512×512
PGGAN	网络结构可以随着训练进行逐渐加深 使用浅层网络训练好低分图像后加深网络深度训练 分辨率更高的图像	CelebA LSUN	1024×1024
Style-GAN	在 PGGAN 的基础上增加映射网络、样式模块 增加随机变换、样式混合等功能块 使用新的权重截断技巧	FHHQ	1024×1024

5 流模型

主流深度生成模型中，VAE 推导出了似然函数的变分下界，但用容易求解的变分下界代替真实的数据分布属于近似方法，得到的近似模型无法得到最好的生成效果；N 虽然用模型对抗和交替训练的方法避免了优化似然函数，保留了模型的精确性，但在训练过程会出现各种问题，因此研究一种既能保证模型精度又容易训练的深度生成模型是有意义的。

流模型的基本思想是：真实数据分布一定可以由转换函数映射到人为给定的简单分布，如果该转换函数是可逆的且可求出该转换函数的形式，则这个简单分布和转换函数的逆函数就能够构成一个深度生成模型。可逆函数的性质说明 Flow 模型是一个精确模型，有希望生成质量足够好的样本。

Flow 模型的相关论文较少，重要的论文中存在很多必须了解的基本结构，因此本节首先介绍 Flow 的基础框架，然后详细说明 NICE、Real NVP 和 Glow 等常规流、i-ResNet 以及变分流等模型的结构。

5.1 流模型框架

数据分布 $P(x)$ 通过转换函数 $G(x)$ 将该分布映射为指定的简单分布，假设该分布是各分量独立的高斯分布，则 $P(x)$ 可以表示成带有转换函数和雅可比行列式的如下形式：

$$P(x)=\frac{1}{(2\pi)^{D/2}}\exp\left(-\frac{1}{2}\|G(x)\|^2\right)\left|\det\left[\frac{\partial G}{\partial x}\right]\right| \quad (60)$$

其中 det(·) 表示雅可比行列式。根据该目标函数优化能得到 $G(x)$ 中的参数，进而得知逆函数 $F(z)$ 的具体形式，这样就能得到一个生成模型。但雅可比行列式的计算量很大，转换函数的逆变换难以求解，为了保证计算上的可行性，$G(x)$ 必须满足如下条件：

①雅可比行列式容易计算；

②函数可逆，求逆过程的计算量尽量小。

雅可比行列式的维数与数据维数相关，对于高维数据而言，雅可比行列式的计算量要比函数求逆更大，因此 $G(x)$ 首先要满足第一个条件。流模型提出将雅可比行列式设计为容易计算的三角阵行列式，其值等于对角线元素乘积从而简化求解雅可比行列式的计算量:

$$\left|\det\left[\frac{dh^i}{dh^{i-1}}\right]\right|=\text{sum}\left|\text{diag}\left[\frac{dh^i}{dh^{i-1}}\right]\right| \quad (61)$$

三角阵行列式的上三角或下三角区域元素的值为 0 意味着每次转换都只有一部分元素参与了映射，另一部分元素只进行了恒等变换，这种简单变换产生的非线性较弱，需要多个简单变换的复合形式增强模型的拟合能力。根据链式法则可得：

$$\frac{\partial z}{\partial x}=\frac{\partial h^1}{\partial x}\cdot\frac{\partial h^2}{\partial h^1}\cdots\frac{\partial h^k}{\partial h^{k-1}}\cdot\frac{\partial z}{\partial h^k} \quad (62)$$

流模型的转换函数用神经网络表示，该神经网络相当于一系列转换函数作用效果的累积，这种简单变换的叠加过程如同流水一般积少成多，因此将这样的过程称为‘流’，大部分流模型都以这种模型框架为基础。此时流模型的对数似然函数可以写成：

$$\begin{aligned}&\log P(x)\\&=-\log P(z)-\sum_{i=1}^{k}\log\left|\det\left(\frac{dh_i}{dh_{i-1}}\right)\right| \quad (63)\\&=-\sum_{i=1}^{k}\left(\frac{1}{2}\|G^i(x)\|^2-\log\left|\det\left(\frac{dh_i}{dh_{i-1}}\right)\right|\right)+c\end{aligned}$$

其中 $c=-\frac{D}{2}\log(2\pi)$ 表示常数。

5.2 常规流

常规流（Normalizing Flow）是流模型中最主要的模型，包括一脉相承的 NICE、Real NVP 和 Glow 三个模型。这三个模型提出了流模型的概念、确立了模型的基本框架以及转换函数的具体形式，使模型性能逐步提高，最新的 Glow 已经可以生成大分辨率的高清人脸图像。

5.2.1 NICE

非线性独立成分估计（Nonlinear Independent Components Estimation，NICE）[5]是第一个流模型，此后出现的流模型大部分都是以 NICE 的结构和理论为基础。除了流模型的基本框架外，NICE 提出了三个重要的模型结构：加性耦合层、维数混合和维数压缩层。

加性耦合层 NICE 提出将雅可比行列式构造成三角阵形式，并将这种结构称为耦合层（coupling layer）。耦合层将 D 维输入变量分割成两部分 $x_D=[x_{1:d},x_{d+1,D}]=[x_1,x_2]$，然后取如下变换：

$$\begin{aligned}h_1&=x_1\\h_2&=x_2+M(x_1)\end{aligned} \quad (64)$$

其中 M 表示定义在空间 $\mathbf{R}^d$ 上的任意函数，下一个隐藏层变量为 $h=[h_1,h_2]$，这种只含有加性算法的耦合层被称为加性耦合层（Additive Coupling），其结构如图 15 所示。

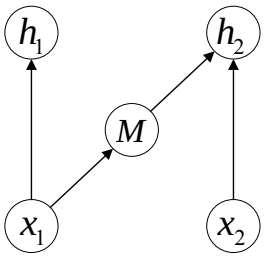

图 15 加性耦合层结构

Fig.15 The structure of aditive couping

加性耦合层的雅可比行列式是上三角行列式且对角线元素全部为 1，用分块矩阵表示该行列式为：

$$\frac{\partial h}{\partial x}=\begin{bmatrix}\partial h_1/\partial x_1 & \partial h_1/\partial x_2\\ \partial h_2/\partial x_1 & \partial h_2/\partial x_2\end{bmatrix}=\begin{bmatrix}I_d & 0\\ \partial h_2/\partial x_1 & I_{D-d}\end{bmatrix}=1 \quad (65)$$

该雅可比行列式的值为 1，根据链式法则可以得到：

$$\det\left[\frac{\partial z}{\partial x}\right]=\det\left[\frac{\partial h^1}{\partial x}\right]\cdots\det\left[\frac{\partial z}{\partial h^k}\right]=1 \quad (66)$$

这使得该项在目标函数中的值为 1，从而消除了雅可比行列式的计算量。该转换函数的逆函数也很容易得到，其逆变换的形式如下：

$$\begin{aligned} x_1 &= h_1 \\ x_2 &= h_2 - M(h_1) \end{aligned} \quad (67)$$

这种结构的转换函数即满足可逆性的要求，且逆函数和雅可比行列式都容易求解，不需要额外的计算量，后来大部分的流模型都采用了这种结构。

维度混合 转换函数不仅非线性能力较弱，而且每次转换过程都有一部分元素没有变化。为了使信息能充分混合，NICE 采用在每次耦合层后直接交换两部分元素的位置 $h_1^1=h_2^2$， $h_2^1=h_1^2$，其结构如图 16 所示。

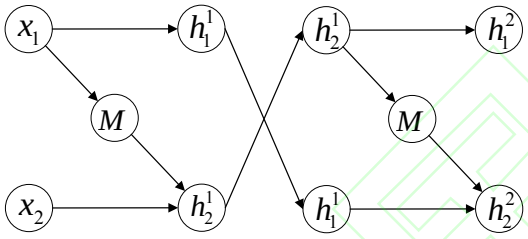

图 16 维数混合结构

Fig.16 The structure of hybrid dimensions

维数压缩层 Flow 是以可逆变换结构为基础的模型，变换可逆性使得模型中各隐藏层的维数需要与输入样本维数 D 的大小相同，这使得 Flow 模型存在严重的维数浪费问题，因此 NICE 提出在最后一层和先验分布之间引入维数压缩层，此时模型的对数似然函数变为

$$\begin{aligned} \log P(x) = &-\frac{D}{2}\log(2\pi)-\sum_{i=1}^{k}\left(\frac{1}{2}\left\|G^i(x)\right\|^2\right) \\ &-\frac{1}{2}\|s\cdot G(x)\|^2+\sum_{i=1}^{D}\log s_i \end{aligned} \quad (68)$$

其中 s 表示维数压缩层中待优化的参数。在压缩层中引入 s 等价于将先验分布的方差也作为参数进行优化。如果某个方差接近 0，说明其对应的维数所表示的流形已经塌缩为点，从而起到维数压缩的作用。

5.2.2 Real NVP

Real NVP[63]的全称为 real-valued non-volume preserving，直译为实值非体积保持，非体积保持是指该模型的雅可比行列式的值不为 1。Real NVP 在 NICE 的基本结构上，提出了比加性耦合层非线性能力更强的仿射耦合层和维数的随机打乱机制，在耦合层中引入卷积层使得 Flow 模型可以更好地处理图像问题，并设计了多尺度结构以降低 NICE 模型的计算量和存储空间。

仿射耦合层 NICE 性能较差与耦合层结构过于简单有关，因此 Real NVP 提出在原有的加性耦合层的基础上加入了乘性耦合，两者组成的混合层称为仿射耦合层（affine coupling layer），其结构如图 17 所示。

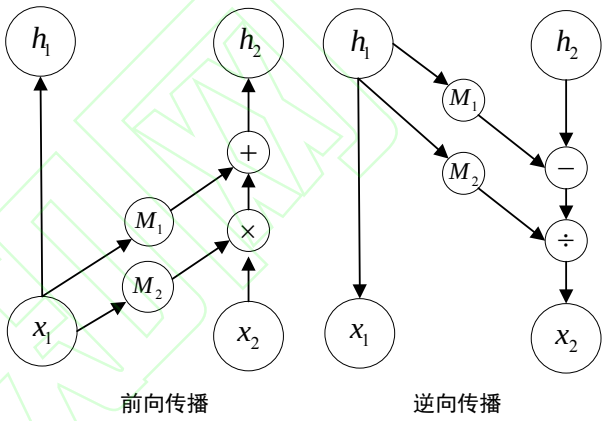

图 17 仿射耦合层结构

Fig.17 The structure of affine coupling layer

该耦合层可以表示成如下形式：

$$\begin{aligned} h_1 &= x_1 \\ h_2 &= x_2 \odot M_2(x_1) + M_1(x_1) \end{aligned} \quad (69)$$

仿射耦合层的雅可比行列式是对角线不全为 1 的下三角阵，用分块矩阵表示该行列式为：

$$\frac{\partial h}{\partial x}=\begin{bmatrix} I_d & 0 \\ \dfrac{\partial h_2}{\partial x_1} & M_2(x_1) \end{bmatrix} \quad (70)$$

该行列式的值为对角线元素乘积，为了保证可逆性需要约束雅可比行列式对角线各元素均大于 0，因此 Real NVP 直接用神经网络输出 $\log s$。该转换函数的逆函数很容易表示为：

$$\begin{aligned} x_1 &= h_1 \\ x_2 &= \frac{h_2 - M_1(x_1)}{M_2(x_1)} \end{aligned} \quad (71)$$

随机混合机制 NICE 性能较差的另一个原因是交换两个分量的位置不能充分混合变量信息，因

此 Real NVP 采用随机混合机制，对耦合层之间的分量随机打乱，再将打乱后的向量重新分割成两部分并输送到下个耦合层中，其结构如图 18 所示。

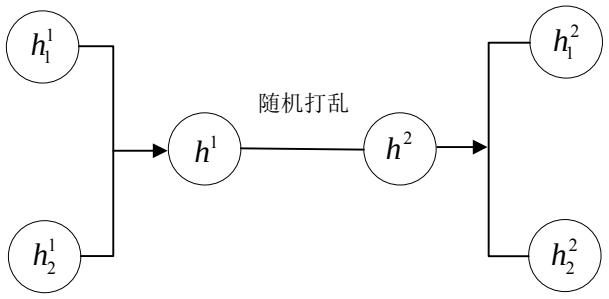

图 18 随机混合结构

Fig.18 The structure of random mixing

掩码卷积层 为了更好的处理图片样本，Real NVP 在流模型中引入了卷积层。卷积方法可以捕捉样本在空间上的局部相关性，但是随机打乱机制会使样本原有的局部相关性消失，为此 Real NVP 提出先使用掩码增加样本通道数并降低空间维数，棋盘掩码是一种固定间隔的空间轴上的交错掩码，能够有效保留样本在空间的局部相关性：

$$h \times w \times c \to \frac{1}{n} h \times \frac{1}{n} w \times 2nc \qquad (72)$$

用棋盘掩码增加样本通道数的操作称为挤压（squeeze），是流模型中使用卷积层的必须步骤，然后对样本的通道执行分割和打乱操作，这种方式保留了样本的局部相关性，以便直接使用卷积网络，大幅度提高模型的计算效率。

多尺度结构 NICE 的加性耦合层和 real NVP 的仿射耦合层在每次执行时都有部分维数的向量没有改变，因此 real NVP 提出在仿射耦合层中使用如图 19 所示的多尺度结构，是仿射耦合层交替变换的一种组合结构。

将样本分成四部分 $x=[x_1,x_2,x_3,x_4]$ 并输入到耦合层中，第一次转换将 x_1 和 x_2 转换成 h_1 和 h_2 后当作多尺度结构的结果 z_1 和 z_2，然后将没有改变的 h_3^1 和 h_4^1 输入到耦合层中继续转换，得到转换后的结果 z_3 和没有改变的 h_4^2，最后在第三次转换过程中将 h_4^2 转换成 z_4。

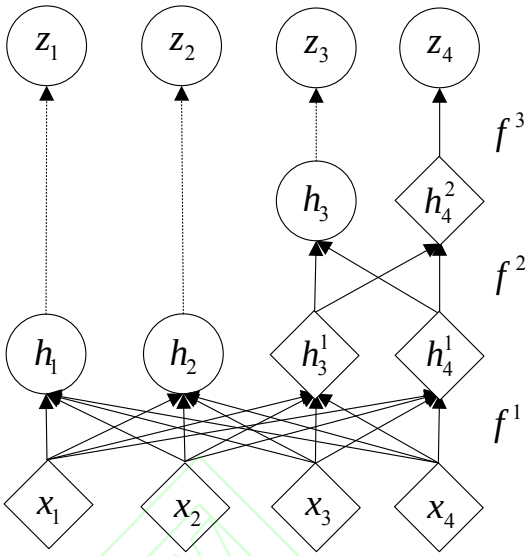

图 19 仿射耦合层的组合策略

Fig.19 Composition schemes for affine coupling layers

多尺度结构通过这种逐层转换的方式，使数据的全部元素都可以在一个复合耦合层内进行转换，保留了原有方法中雅可比行列式容易计算的特点，减少模型复杂度和计算量的同时增加模型的生成能力。

5.2.3 GLOW

GLOW[64]是以 NICE 和 real NVP 为基础结构的模型，是当前流模型中效果最好的模型。GLOW 模型主要有两个贡献：第一个贡献是修改流模型的结构，提出完整的模型结构，引入 Actnorm 层；第二个贡献是提出 1 乘 1 卷积和 LU 矩阵分解方法并将置换矩阵当作优化项。

模型结构修改 GLOW 以 real NVP 模型为基础构造了性能更好的模型框架，并针对 real NVP 的不足进行两处修改：

1.仿射耦合层内部的乘性耦合使得其计算量是加性耦合层的两倍，但经过实验证明仿射耦合层的性能提升很小，因此 GLOW 训练高维样本时为了减少计算量只保留加性耦合层。

2. GLOW 证明了棋盘掩码的复杂操作不能提升模型的生成能力，因此删除了该模块。

Actnorm 层 由于内存限制，流模型在训练较大的图像时每个批次的样本数通常选 1，因此提出了类似于批归一化处理的 Actnorm 层。Actnorm 用批次样本的均值和方差初始化参数 b 和 s，是对先验分布的平移和缩放，有助于提高模型的生成能力。

置换矩阵 相比于 NICE 中的简单交换，real

NVP 的随机打乱方法可以得到更低的损失，因此 GLOW 提出用 1 乘 1 卷积运算改变置换通道的排列，用置换矩阵替代随机打乱并放到损失函数中一并优化以进一步提升模型效果。

具体方法是通过一个随机旋转矩阵 W 置换输入轴通道的排列顺序使 $h = xW$，为了保证转换函数的可逆性，方阵 W 初始化为随机正交矩阵，因此其雅可比行列式的值为 $\det W$ 。

为了更容易计算雅可比行列式的值，GLOW 利用 LU 矩阵分解法分解正交矩阵 W 使 $W = PLU$，其中 P 是置换矩阵，L 是对角线全为 1 的下三角阵，U 是上三角阵，此时可以容易得到雅可比行列式的值为上三角阵 U 的对角线乘积：

$$\log|\det W| = \sum \log|\mathrm{diag}(U)| \qquad (73)$$

GLOW 使用 LU 分解法计算旋转矩阵 W 的雅克比行列式的值，几乎没有改变原模型的计算量，且减少了待优化参数的数量。实验证明了可逆 1×1 卷积可以得到比随机打乱机制更低的损失且具有很好的稳定性。

GLOW 的单个转换结构包括 Actnorm 层、可逆 1 乘 1 卷积和耦合层，其流程图如图 20 所示。图中的超参数 K 和 L 表示循环次数。样本 x 先进行 squeeze 操作后用单步转换结构迭代 K 次，然后将转换的结果进行维数分割，分割后的两部分变量与多尺度结构的结果意义相同，将整个过程循环 $L-1$ 次后将未转换过的部分维数再次进行 squeeze 操作和 K 次单步转换，以上构成了 GLOW 的多尺度结构。

GLOW 进一步提升了流模型的性能，各个数据集上的表现都超过了其他所有流模型，可以生成清晰度很高的人脸图像，缺点是置换矩阵导致模型的层数很多，拥有生成式模型中最大的参数量级，例如生成 256×256 的高清人脸图像需要 600 多个耦合层和 2 亿多个参数，训练成本很高，因此改进自身结构或使用非线性程度更高的转换函数以降低训练成本和模型深度是提高流模型实用性的关键。

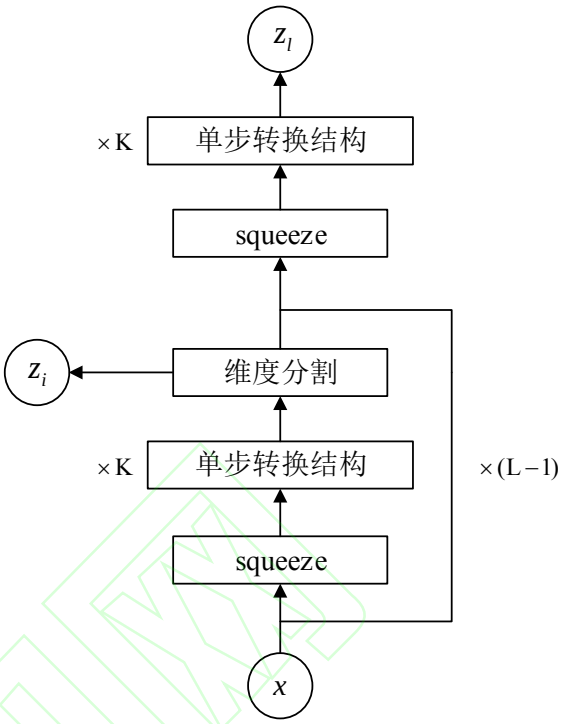

图 20 GLOW 的层结构

Fig.20 The structure of layers in GLOW

5.3 可逆残差网络

以 GLOW 为代表的常规流模型有两个严重的问题：第一个问题是流模型为了保证转换函数的雅可比行列式在计算量上的可行性，导致单层转换函数的非线性变换能力很弱，过多耦合层的累加使模型的参数个数巨大；第二个问题是为了有一个容易求解的逆函数，流模型的耦合层的存在，导致模型是不对称的。

可逆残差网络（Invertible Residual Networks, i-ResNet）[65]是以残差网络为基础的生成模型，利用约束使残差块可逆，然后用近似方法计算残差块的雅可比行列式，这使得 i-ResNet 与其它流模型有本质区别：保留了 ResNet 的基本结构和拟合能力，使残差块是对称的又有很强的非线性转换能力。

5.3.1 残差块的可逆性条件

i-ResNet 的基本模块与 ResNet 相同，可以表示成 $y = x + G(x)$，残差块用神经网络 $x + G(x)$ 拟合 y，使得残差块的梯度 $1 + \partial G(x)/\partial y$ 不会在深层网络中出现梯度消失的问题，以便训练更深层次的网络。将 i-ResNet 构造成流模型，首先要保证模型的可逆性，等同于保证单个残差块的可逆性。残差块可逆性的充分不必要条件是函数 $G(\cdot)$ 的 Lipschitz

范数小于 1 即 $\mathrm{Lip}(G)<1$。因此神经网络拟合的函数 $G(\cdot)=F(Wx+b)$ 使用普通激活函数时，其可逆性条件等价于权重矩阵 W 的谱范数小于 1：

$$\mathrm{Lip}(G)<1 \Leftrightarrow \mathrm{Lip}(W)<1 \qquad (74)$$

因此只要对 $G(\cdot)$ 内的所有权重矩阵进行谱归一化后乘一个介于 0 和 1 之间的系数即可保证残差块的可逆性：

$$W \leftarrow \frac{cW}{\|W\|_2} \qquad (75)$$

5.3.2 i-ResNet 的求解方法

流模型需要直接计算出残差块的逆函数，但残差块的形式导致很难直接求出逆函数的解析形式，为了简化计算，i-ResNet 使用迭代 $x_{n+1}=y-G(x_n)$：当 x_n 收敛到某个固定函数时表明得到了足够近似的逆函数，并给出限制 $\mathrm{Lip}(G)>0.5$ 保证 x_n 的收敛性。

i-ResNet 的关键是如何求解残差块的雅可比行列式的值，雅可比行列式可以表示如下：

$$\frac{\partial(x+G(x))}{\partial x}=I+\frac{\partial G}{\partial x} \qquad (76)$$

为了求解该式，i-ResNet 先后使用级数展开、截断和随机近似三种数学方法：首先用恒等式将雅可比行列式绝对值的对数转化为求迹，并在使用级数展开形式后在第 n 项截断，然后使用随机近似方法得到近似值。

i-ResNet 使用多种手段直接且较高效的求解出残差块的雅可比行列式，尽管模型的生成能力与 GLOW 相差很大，但摆脱了耦合层的弊端，是对 FLOW 模型的革新和大胆的尝试。

5.4 变分推理流

如 VAE 等使用变分推断的生成模型通常将近似后验分布假设成高斯分布或高斯混合分布，但高斯分布只是众多可能的后验分布中的很小一部分，如果真实的后验分布与假设分布相差较大，则解码器的拟合效果很差，导致生成器只能生成模糊的图像样本。

为了增加后验分布假设的丰富性，变分推理流模型（Variational Inference with Flow）在流模型中引入变分推断：将编码器输出的均值和方差用转换函数映射到更复杂的分布，再由解码器根据后验分布重构样本。这种方法使变分流映射得到的后验分布更接近真实的后验分布。

5.4.1 归一化流的变分推断

在 VAE 变分推断过程引入流模型结构的归一化流变分推断（Variational Inference with Normalizing Flow， VINF）[66]是变分流模型的一种，对于编码器得到的分布 $P(z)$，VINF 用归一化流将该分布映射为 $P(z_K)$，映射函数为：$G(z)=z+uF(z)$，其中 $F(z)=F(w^Tz+b)$ 表示神经网络，u,w,b 为模型参数。令 $\psi(z)=F'(w^Tz+b)$，则转换函数的雅可比行列式为：

$$\det\left|\frac{\partial F}{\partial z}\right|=\det\left|I+u\psi(z)^T\right|=\left|1+u^T\psi(z)\right| \qquad (77)$$

由此可以推导出的变分下界为：

$$\begin{aligned} &L(x)\\ &=\mathrm{E}_{Q(z|x)}(-\log Q(z|x)+\log P(z,x))\\ &=\mathrm{E}_{P(z)}[\ln P(z)]-\mathrm{E}_{P(z)}[\log P(x,z_K)]\\ &-\mathrm{E}_{P(z)}\left[\sum_{n=1}^{K}\ln\left|1+u_n^T\psi z_{n-1}\right|\right] \end{aligned} \qquad (78)$$

VINF 认为该转换函数相当于对初始密度 $P(z)$ 在垂直于超平面 $w^Tz+b=0$ 方向上进行的一系列收缩和扩展，因此称之为平面流（planar flow），这与 real NVP 的命名方法相似，此外还提出了沿固定点径向收放的径向流（radial flow）。

5.4.2 可逆自回归流的变分推断

自回归结构是用当前时刻以前的观测序列预测当前时刻的函数值，将自回归结构的流模型应用在 VAE 变分推断中的模型称为可逆自回归流（Inverse Autogressive Flow, IAF）[67]。IAF 的可逆自回归流可以分为两个部分，第一个部分以数据 x 作为编码器的输入，输出的三个变量分别为高斯分布的均值 μ、方差 σ^2 和辅助变量 h'，然后重参数化得到后验变量 z，其结构如图 21。

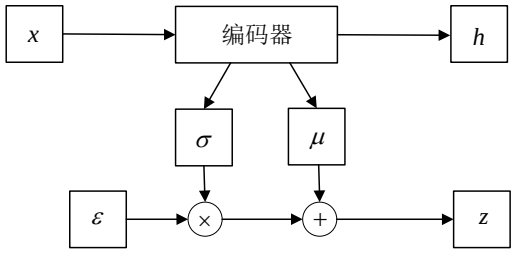

图 21 IAF 第一层结构

Fig.21 The structure of the first layer in IAF

第二部分的输入为后验变量 z 和辅助变量 h'，

通过自回归网络输出 μ_t 和 σ_t^2，其结构如图22所示。

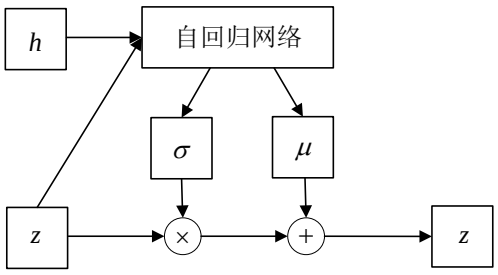

图 22 IAF 其余层结构

Fig.22 The structure of other layers in IAF

因此 IAF 的转换结构可以表示为：

$$h_t = \mu_t + \sigma_t \odot h_{t-1} \tag{79}$$

该转换函数的雅可比行列式容易计算：正向传播的雅可比行列式是对角线为 0 的下三角矩阵，逆向传播的雅可比行列式是对角线为 $dz_t / dz_{t-1} = \sigma_t$ 的下三角矩阵，由此可以得到自回归流的近似后验分布为：

$$\begin{aligned} &\log Q(z \mid x) \\ &= -\frac{1}{2}\sum_{i=1}^{D}\left(\varepsilon_i^2 + \log(2\pi) + \sum_{t=0}^{T}\log\sigma_{t,i}\right) \end{aligned} \tag{80}$$

IAF 在第二部分的自回归网络中采用长短时记忆网络（Long Short Term Memory， LSTM），此时 LSTM 的输出为 $[m_t, s_t]$，然后根据如下公式计算后验分布的均值和方差：

$$\begin{aligned} \sigma_t &= \mathrm{sigm}(s_t) \\ z_t &= \sigma_t z_{t-1} + (1-\sigma_t)m_t \end{aligned} \tag{81}$$

掩码自回归流（Masked autoregressive flow，MAF）[68]是 IAF 的衍生模型，MAF 将 real NVP 中的掩码卷积层引入到 IAF 中，使 IAF 能够更好的处理图像样本，然后提出了条件掩码自回归流 CMAF 将 MAF 应用到监督模型中。基于计算考虑，IAF 和 MAF 有互补性：MAF 训练快生成样本慢，IFA 训练慢生成样本快。对于生成模型来说，训练过程的计算量通常很大，因此在生成任务中 MAF 更为适用。

5.5 总结

Flow 是一个非常精巧的模型，也是在理论上没有误差的模型。Flow 设计了一个可逆的编码器，只要训练出编码器的参数就能直接得到完整的解码器，完成生成模型的构造。为了保证编码器的可逆性和计算上的可行性，目前 Flow 类模型只能使用多个耦合层的堆叠来增加模型的拟合能力，但耦合层的拟合能力有限，这种方法很大程度上限制了模型的性能。Flow 目前的应用范围集中在图像生成领域中的人脸生成，最优秀的模型为 GLOW。

相比于以 GAN 为首的其他深度生成模型，Flow 参数量更多、运算量更大，且应用领域只局限于图像生成，这些弊端限制了 Flow 的进一步发展，作为无误差的生成模型，潜力巨大的 Flow 模型应该在未来的研究中寻找更高效的可逆编码器结构或者拟合能力更强的耦合层，并扩展模型的应用范围。

6 自回归网络

自回归是统计学中处理时间序列的方法，用同一变量之前各个时刻的观测值预测该变量当前时刻的观测值。用条件概率表示可见层数据相邻元素的关系，以条件概率乘积表示联合概率分布的模型都可以称为自回归网络。

自回归网络中最有影响力的模型是神经自回归分布估计，该模型起源于受限玻尔兹曼机 RBM，将其中的权重共享和概率乘积准则与自回归方法结合，该模型的前向传播等同于假设隐藏变量服从平均场分布的 RBM，且更灵活、更容易推理，模型性能也更好。

6.1 自回归网络的基本形式

自回归网络的基本形式有三种：线性自回归网络、神经自回归网络和 NADE。线性自回归网络[69]是自回归网络中最简单的形式，没有隐藏单元、参数和特征共享；神经自回归网络[70]的提出是为了用条件概率分解似然函数，避免如 DBN 等传统概率图模型中高维数据引发的维数灾难。

神经自回归网络是具有与线性自回归相同结构的有向图模型，该模型采用不同的条件分布参数，能够根据实际需求增加容量，并允许近似任意联合分布。另外神经自回归网络可以使用深度学习中常见的参数共享和特征共享等方法增强泛化能力。

6.2 NADE

6.2.1 NADE 模型结构

观测数据的有序排列起源于完全可见贝叶斯网络（Fully Visible Bayes Nets, FVBN）[71]，该算法

最早定义了将高维数据的概率通过链式法则分解为条件概率乘积的方法。神经自回归分布估计模型（Neural Autoregressive Distribution Estimation，NADE）[6]根据这种方法进行建模：

$$P(x)=\prod_{d=1}^{D}P(x_{o_d}\mid x_{o_{<d}}) \tag{82}$$

其中 $x_{o_{<d}}$ 表示观测 D 维观测数据中位于 x_{o_d} 左侧的所有维数，表明该定义中第 i 个维数的数值只与其之前的维数有关，与之后的维数无关。

RBM 中输出层到隐藏层的权重是隐藏层到输入层权重的转置，而 NADE 可以利用上述公式独立参数化各层之间的权重。另外，模型中引入了附加的参数共享，将条件分布进行参数化并写成如下形式

$$\begin{aligned}P(x_d=1\mid x_{<d})&=\mathrm{sigm}(V_{d,\cdot}\cdot h_d+b_d)\\ h_d&=\mathrm{sigm}(W_{\cdot,d}\cdot x_d+c)\end{aligned} \tag{83}$$

其中 $V\in R^{D\times H}$，$b\in R^{D}$，$W\in R^{H\times D}$，$c\in R^{H}$ 均为模型参数。$V_{d,\cdot},W_{\cdot,d}$ 分别表示两个矩阵的 d 行和 d 列，说明两个矩阵和偏置 c 是共享参数，使 NADE 算法只需要 O(HD) 个数的参数，且可以降低过拟合的风险。此外该算法容易递归计算：

$$\begin{aligned}&h_1=\mathrm{sigm}(a_1),a_1=c\\ &h_d=\mathrm{sigm}(a_d)\\ &a_d=W_{\cdot,<d}x_d+c=W_{\cdot,d-1}x_{d-1}+a_{d-1}\end{aligned} \tag{84}$$

从公式可以看出每次计算隐藏变量 h 和条件概率需要的计算量均为 $O(h)$，因此计算 $P(x)$ 概率的计算量为 $O(hd)$，共享参数的引入使得 NADE 在正向传播和均匀场推断中执行的计算大致相同。

6.2.2 模型扩展

单元修正 Bengio 等人[72]指出 h_d 的多次累加会使隐藏层单元越来越饱和，因此添加权重衰减参数以降低隐藏层单元的饱和现象：

$$h_d=\mathrm{sigm}(\rho_d a_d),\ \rho_d=1/i \tag{85}$$

实验中发现使用修正线性单元作为激活函数可以得到更好的生成效果。

NADE-*k* 为了使 NADE 模型能够更好的推断数据中的缺失值，Raiko 等人[73]根据 CD-k 算法的思想对可见层和隐藏层之间进行反复迭代，替代原始 NADE 的单次迭代，实验显示这种方法能有效提升 NADE 模型推断缺失值的能力，该模型可以称为 NADE-k。

并行 NADE 尽管 NADE 的训练速度很快，但条件概率的有序性使得模型无法并行处理，生成样本的速度很慢。Reed 等人[74]为了打破像素之间的弱相关性，提出允许对某些像素组建模使之条件独立，只保留高度相关的临近像素，从而使 NADE 可以并行地生成多个像素，大大加快抽样速度，使隐藏变量和条件概率需要的计算量由 $O(h)$ 锐减到 $O(\log h)$，但是这个舍弃像素之间弱相关性的方式必然会一定程度的影响模型的性能。

6.2.3 深度 NADE

有多个隐藏层的深度 NADE 模型在第一个隐藏层和输出层的计算量和上述的单层模型相同，文中推导出了无序时损失函数的无偏估计，并在输入层引入掩码，使用卷积神经网络处理高维数据。此外，NADE 能够通过随机抽样处理任意顺序的数据，将信息提供给指定用于观测某些输入的隐藏单元并预测缺失信息，这种形式可以使 NADE 模型能高效推断[75]。

深度 NADE 模型有两个问题。第一个问题是条件概率的值与数据的排列顺序是相关，但模型接收到的数据排列是随机的，想找到最合适的排列顺序需要 $O(D!)$ 的计算量和存储空间，而直接将单个 NADE 堆叠，非线性的计算将导致计算量 $O(dh^2l)$ 过大，其中 l 表示隐藏变量层数。另一个缺点是随着隐藏层层数增加，其他隐藏层的计算量会大大增加达到 $O(h^2d^2)$，巨大的计算量使得 NADE 模型很难扩展到多层。实验证明 NADE 模型的效果比 DBM 好，但模型参数数量还是计算量都远大于 DBM。

6.2.4 卷积 NADE

使用条件概率乘积表示联合概率的 NADE 只适用于低维数据，Uria[76]提出将 CNN 与 NADE 结合的 ConvNADE 模型，该模型首先将图片输入到 CNN 网络中进行特征提取，然后将特征输入到 NADE 中，此时 NADE 输入层的条件概率公式可以表示成：

$$P(x_{o_d}\mid x_{o_{<d}})=\mathrm{sigm}(\mathrm{vec}(h^l)_{o_d}+b) \tag{86}$$

其中 h^l 表示卷积网络的第 l 层输出，vec(·) 表示行之间关系的函数，在 CNN 和 NADE 之间的层使用掩

码作为辅助通道可以有效提高模型性能。

6.3 像素循环神经网络

像素循环神经网络（Pixel Recurrent Neural Network, PixelRNN）[77]将图片的像素作为循环神经网络的输入，本质上是自回归神经网络在图片处理上的应用，国外很多文献把 PixelRNN 模型称为自回归网络。该模型利用深度自回归网络预测图片的像素值，并提出三种不同结构的深度生成模型。

PixelCNN 该模型直接利用卷积神经网络（Convolutional Neural Network, CNN）处理像素，然后用特殊结构的掩码避免生成样本时出现缺少像素的问题。这种方法结构简单、训练速度快且稳定，而且能够直接以似然函数作为目标，使 PixelCNN 的似然指标远超过其他的深度生成式模型，但缺点是生成的样本不理想，原因可能是卷积核不够大。

Row LSTM 这种模型结构能捕捉到更多邻近像素的信息，该模型对 LSTM 的输出进行行卷积，且三个门也由卷积产生，这种方法可以捕捉到更大范围的像素，但问题是该模型的像素依赖区域是个漏斗形状，明显会遗漏很多重要的相关像素。

Diagonal BiLSTM 为了捕捉到更多相关像素信息，作者提出了第三种模型。该模型通过重新构造像素位置的方法使 LSTM 的输入不存在遗漏像素，即双向长短时记忆网络 BiLSTM。BiLSTM 利用特征映射的翻转构造双向的 LSTM 网络，消除映射时的像素盲点，比 Row LSTM 更好的捕捉像素信息。

这几种模型的本质都是捕捉当前元素周围的像素信息，用残差结构优化深度模型，序列化的产生像素样本，但逐个像素生成的样本生成方式导致模型生成速度很慢。

6.4 掩码自编码器

掩码自编码器（Masked Autoencoder for Distribution Estimation， MADE）[78]是将自回归的方法应用到自编码器中，提高自编码器估计密度的能力，实现方法主要是利用掩码修改权重矩阵使自编码器的输出成为自回归形式的条件概率。

自编码器通常表示能力较差，因此适合与表示能力强的自回归模型结合。根据自回归估计概率密度的方法，MADE 的输出应为条件概率，当输入数据为二值时，模型的目标函数是交叉熵损失函数。自编码器的权重矩阵部分连接行的值为 0，构造这种权重矩阵最容易的方法就是对权重矩阵进行掩码处理，阻断无关变量之间的连接通道，实现自编码器和自回归网络的结合。

MADE 的另一个优势是很容易扩展到深层网络，只需要增加隐藏层的层数并添加对应的掩码。作者给出了其他隐藏层掩码的设计方法和针对掩码的不可知连接方法的训练算法。从实验结果可以看出 MADE 的生成能力与 NADE 基本持平并在部分数据集上超过了 NADE。

6.5 应用和总结

自回归结构最大的优势是可以对序列数据进行很好的密度估计，可以与其他生成模型结合，如将自回归结构和自编码器结合用于生成段落和文本等序列样本的自回归自编码器[79-80]，和卷积网络结合生成人类语音的生成模型 WaveNet[81]等。

自回归网络的缺点是表示目标函数的条件概率乘积无法并行运算，导致训练和生成样本所需要的计算量远大于 VAE 和 GAN 等其他通用模型，这极大限制了自回归生成模型的发展和应用，寻找合理的并行运算方式是破解这个难题的关键。

7 其他深度生成模型

7.1 矩阵匹配网络

最大平均差异（Maximum Mean Discrepancy, MMD）最早用于双样本检测[82]，从概率统计的角度比较两个数据集的差异程度，并根据两个数据集的可观测数据判断这两个数据集的分布是否相同。基于两个分布 $P(x)$ 和 $Q(x)$ 的样本，通过寻找再生核希尔伯特空间上的连续函数 F 使得两个分布的样本在该函数上的函数值的均值差最小：

$$D_{\mathrm{MMD}}[F,P,Q]=\sup(\mathrm{E}_{x\sim P}[F(x)]-\mathrm{E}_{x\sim Q}[F(x)]) \quad (87)$$

当且仅当两个函数有相同分布时函数值等于 0。

在生成式模型的损失函数中用 MMD 作为衡量数据分布和模型分布差异的模型称为生成式矩阵匹配网络（Generative Moment Matching Networks，GMMN）[83]。两个分布的最大平均差异的平方为：

$$
\begin{aligned}
&L_{\mathrm{MMD}^2} \\
&=\left\|\frac{1}{N}\sum_{i=1}^{N}F(x_i)-\frac{1}{M}\sum_{j=1}^{M}F(x_j)\right\|^2 \\
&=\frac{1}{N^2}\sum_{i,i'=1}^{N}K(x_i,x_{i'})+\frac{1}{M^2}\sum_{j,j'=1}^{M}K(x_j,x_{j'}) \\
&\quad-\frac{2}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M}K(x_i,x_j)
\end{aligned}
\tag{88}
$$

其中 $K(\cdot)$ 表示核函数。MMD 可以使模型无需 MCMC 抽样直接用批次随机梯度下降法训练，GMMN 模型原来简单，在理论上有可行的解释，但在生成模型中单独使用双样本检验方法的效果不如 GAN 和 VAE 等模型，且在训练过程中的批次需要较大样本数量，使该模型的计算量较大，所以通常将 MMD 方法应用到其他生成模型中。

将 MMD 引入到 VAE 模型中：用贪婪算法预训练自编码器，然后固定自编码的参数并在隐藏层加入先验为均匀分布的 GMMN，最小化数据样本和通过 GMMN 与 VAE 解码器共同生成的样本的最大平均差异，该模型称为 GMMN+AE，实验表明附加 GMMN 的模型性能得到有效提升。

Li 等人[84]将 MMD 引入到 GAN 模型中，把 GAN 中的判别器替换成基于核的 MMD 双样本检验，基本思想是通过引入基于对抗核的学习方法替代原来的高斯核，提高了 GAN 的生成能力和计算效率。

Ren 等人[85]提出了基于条件最大平均差异 CMMD 的生成模型，在 MMD 中引入条件分布来提高该模型在某些任务上的性能。该模型在保持原模型训练过程简单性的同时扩展了模型性能，提高模型的生产能力和判别能力。

7.2 生成式随机网络

生成随机网络（Generative Stochastic Network，GSN）[86]是降噪自编码器的一般化形式，可以使用反向传播快速训练，主要用于缺省值预测和结构化输出。GSN 可以看成是自编码器和 DBN 的结合[87]，但该模型直接将生成过程参数化，利用参数化马尔科夫过程代替直接参数化似然函数，将无监督密度估计问题转化成监督问题的近似。GSN 在马尔科夫链中加入隐藏变量，然后利用指定的一步马尔科夫链重复迭代预测可见变量。

对于来自分布 $P(x)$ 的样本，降噪自编码器构造分布 $P(\hat{x}\mid x)$ 以得到带有噪声的样本，利用自编码器学习重构分布。GSN 继承了这一思想，根据上述得到的分布 $P(\hat{x}\mid x)$ 和重构分布 $P(x\mid\hat{x})$，利用贝叶斯公式可以求得 $P(x)$：

$$
P(x\mid\hat{x})=\frac{1}{Z}P(\hat{x}\mid x)P(x) \tag{89}
$$

其中 z 表示与 x 不相关的归一化常数。重构分布比数据分布更容易学习，因此利用降噪自编码器的方法可以更容易的求出 $P(x)$ 的分布。

GSN 的计算过程类似于 DBN 中的 Gibbs 抽样，但不需要进行预训练算法。如果将降噪自编码器的马尔科夫链定义为两个条件概率分布的参数化形式：$\hat{x}_t\sim P(\hat{x}\mid x_t)$ 和 $x_{t+1}\sim P(x\mid\hat{x}_t)$，则其一般形式的 GSN 指定马尔科夫链的一步：

1. $h_{t+1}\sim P(h\mid h_t,x_t)$ 表示在给定先前的隐藏变量和可见变量时如何更新当前时刻的隐藏变量，相当于降噪自编码器中的编码器。

2. $x_{t+1}\sim P(x\mid h_{t+1})$ 表示在给定潜在状态的当前值后如何产生下一个可见变量，相当于降噪自编码器中的解码器。

GSN 的目标函数是 $\log P(x_k\mid h_k)$，并使用重参数化技巧和回退训练过程。回退训练过程原本是加速降噪自编码器训练过程收敛速度的方法，该过程用多个随机编码解码步骤组成以初始化训练样本，虽然从马尔科夫平稳分布的角度看多个步骤和单步骤是等价的，但实际应用中可以有效去除数据中的伪信息。回退训练过程同样可以改善 GSN 的收敛性。

用于无监督聚类的 GSN 可以在监督学习中使用[88]，通过在重构概率上添加监督数据的标签信息，提出了混合目标函数，将原本的生成目标函数和带有标签信息的判别目标函数混合在一起并用系数权衡两者的权重，这种方法可以有效的在监督数据中使用 GSN 算法。

8 未来趋势及发展方向

深度生成式模型试图把概率论与数理统计的知识与强有力的深度神经网络的表示学习能力相结合，在最近几年取得了显著进步，是当前主流的深度学习方向。本文对深度生成式模型的主要类型进行了梳理，给出了模型的构造过程、优缺点以及

模型存在的问题。深度生成式模型虽然大有潜力，但也存在很多挑战：

1）**评估指标与评估系统**：和判别式模型、基于矩阵与线性代数的模型、基于几何的模型相比，深度生成模型存在训练过程复杂、结构不易理解和使用、训练速度慢等问题，在大规模数据上学习模型很困难，在不同的应用领域应该有相应的有效评估指标和实用的评估系统是急需研究的问题。

2）**不确定性**：深度生成模型的动机和构造过程通常有严格的数学推导，但在实际过程往往限于求解的难度不得不进行近似和简化，使模型偏离原来的目标。训练好的模型难以在理论上分析透彻，只能借助实验结果反向判断调整方法，对生成模型的训练造成很大困扰，是限制模型进一步发展的重要因素。因此了解模型的近似和简化对模型性能、误差和实际应用的影响是发展生产模型的重要方向

3）**样本多样性**：如何使深度生成模型生成的图像、文本和语音等样本具有多样性是一个值得研究的问题。度量多样性最基本的标准是熵，因而把生成模型与最大互信息结合的 Info-VAE 和 Info-GAN[89]等模型既能限制生成模型的灵活性又能提升样本的多样性；把训练样本看作多个概率分布的噪声混合后的随机变量，提取不同噪声的特征表示，得到不同层次的特征表示，在训练目标函数里显式地引入不同的归纳偏置。

4）**泛化能力**：机器学习理论认为好的模型要具有更好的泛化能力。重新思考深度学习的泛化能力，从模型复杂性、偏差-方差权衡等观点，理论上讨论各种深度生成模型的学习机制，丰富模型的理论基础，从而真正确立深度生成模型在深度学习中的显著地位是值得思考的问题。

5）**更高效的模型结构和训练方法**：代表着最先进的一批生成模型如 BigGAN、Glow 和 VQ-VAE[90-91]等已经可以生成足够清晰的图片样本，但这样的大型模型背后是远超常规的计算量，是所有大型生成模型的弊端：高昂的计算机硬件设备以及长时间的训练让很多人难以进入该领域的前沿研究，所以更加高效的模型结构和训练方法是未来发展方向之一。

6）**应用领域扩展**：深度生产模型的应用范围相对较小，如何将其他深度生成模型的思想以及成果运用在常见场景中、如何加速与这些领域的融合，是未来进一步发展深度生成模型的关键方向，如智能家居物联网和自动驾驶等领域都有待深度生成模型的使用。目前生成模型通常用于传统机器学习和人工智能专属领域，对于工业生产等其他领域的应用也有待进一步开发。

7）**生成离散数据**：如 GAN 等深度生成模型的训练依赖于参数的完全可微，因此无法直接生成如独热编码等离散数据。这个问题限制了此类深度生成模型在 NLP 领域的应用，目前已经有初步的解决办法，例如使用 Gumbel-softmax[146]、用连续函数近似[147]等，但效果有待进一步提升。因此研究深度生成模型生成离散数据是提高文本生成能力的关键问题，是值得深入研究的领域。

8）**度量方法**：生成模型可以使用不同的度量方法，例如 GAN 使用的是 KL 散度和 JS 散度，WGAN 使用 Wasserstein 距离替换了原来的散度，可以提升模型的生成能力和训练稳定性。因此通过理论分析，使用新的度量方法可能会进一步提高模型性能。

9 总结

近年来深度学习在多个领域取得了巨大成就而受到人们的广泛关注，作为深度学习中的一个重要分支，深度生成模型在计算机视觉、密度估计、自然语言和语音识别、风格迁移、无监督问题和半监督学习等领域得到成功应用。本文对各类深度生成模型和相关的改进模型进行了详细阐述，重点介绍各种改进模型的结构、算法，然后根据模型的不同特点对模型进行分类、梳理和总结。本文根据深度生成式模型处理似然函数的不同方法将模型分为三类：

第一类方法是近似方法，其中包括采用抽样方法近似计算似然函数的受限玻尔兹曼机以及以该模型为基础模块的深度置信网络、深度玻尔兹曼机等，这类模型开启了深度学习的潮流。尽管因为模型结构和计算方式等原因而逐渐被淡忘，但其完备的理论体系和各种算法对深度生成模型很大的启发和影响；另一种模型直接优化似然函数变分下界的变分自编码器，通过编码、重构和解码三个过程完成了由隐变量到样本的生成过程，最重要的贡献是使用了变分下界并提出了重构方法，重要的改进模型包括重要性加权自编码和深度辅助深度模型，这类模型是当前主流的深度生成模型之一。

第二类方法的代表是生成对抗网络。生成对抗

网络本质上是将难以求解的似然函数转化成神经网络，让模型自己训练出合适的参数来拟合这个似然函数，即模型中的判别器，避开了难以求解的似然函数，因为在多个领域中占有绝对优势而成为当前机器学习领域最热门的研究方向之一。

第三类方法是对似然函数进行适当变形的流模型和自回归模型。Flow 利用可逆函数构造似然函数后直接优化模型参数，属于理论上没有误差的深度生成模型，但雅可比行列式的巨大计算量限制了生成能力的提升，可逆残差网络通过多种运算技巧消除了这种限制，却又带来了计算误差且模型的生成能力较差；自回归模型将目标函数分解为条件概率乘积，这类模型应用非常广泛，包括神经自回归密度估计、像素循环神经网络、掩码自编码器以及 WaveNet 等，主要缺点是条件概率乘积无法并行运算，导致训练和生成样本所需要的时间远大于其他深度生成模型。

从上述各类模型的总结可以看出深度生成模型的种类相当丰富且发展迅速，尽管各类模型都存在一定的问题和限制，但不能否认的是，随着理论研究的进一步深入和应用领域的进一步扩展，深度生成模型必将成为未来人工智能领域的主流技术。

符 号 说 明

符号		含义
a,b	----------	神经元偏置
c	----------	常数
D	----------	散度
$E(\cdot)$	----------	能量函数
$\mathrm{E}(\cdot)$	----------	数学期望
$F(\cdot),G(\cdot)$	----------	神经网络构成的函数
h	----------	中间层神经元
$H(\cdot)$	----------	熵函数
$K(\cdot)$	----------	核函数
$L(\cdot)$	----------	目标函数
$M(\cdot)$	----------	空间上的任意函数
$N()$	----------	高斯分布
$P(\cdot)$	----------	概率分布
$\tilde{P}(\cdot)$	----------	未归一化的分布
$Q(\cdot)$	----------	近似分布
x	----------	可见层神经元
$\hat{x}$	----------	重构样本
W	----------	权重矩阵
z	----------	隐藏层神经元
Z	----------	配分函数
∇	----------	梯度
$P_g(x)$	----------	生成分布
$P_r(x)$	----------	数据分布
α,λ	----------	超参数
θ,φ	----------	模型参数
μ	----------	均值
σ^2	----------	方差

参 考 文 献

[1] Smolensky P. *Information processing in dynamical systems: foundations of harmony theory.* Colorado Univ at Boulder Dept of Computer Science, 1986. 194-281

[2] Kingma D, Welling M. Auto-encoding variational bayes. arXiv :1312.6114, 2013

[3] Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. In: *Proceedings of Neural Information Processing Systems*. 2014: 2672-2680

[4] Bengio Y, Thibodeau-Laufer, Éric, Alain G, et al. Deep generative stochastic networks trainable by backprop. In: *Proceedings of International Conference on Machine Learning*. 2014. 226-234

[5] Dinh L, Krueger D, Bengio Y. NICE: Non-linear Independent Components Estimation. *Computer Science*, 2014

[6] Larochelle H, Murray I. The neural autoregressive distribution estimator. In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. 2011. 29-37

[7] Salakhutdinov R. Learning Deep generative models. *Annual Review of Statistics & Its Application*, 2015, 2(1): 361-385

[8] Liu Jian-Wei, Liu Yuan, Luo XiongLin. Research process of Boltzmann machine. *Journal of Computer Research and Development*, 2014, 51(1): 1-16
(刘建伟, 刘媛, 罗雄麟. 玻尔兹曼机研究进展. 计算机研究与发展, 2014, 51(1): 1-16)

[9] Hinton G E. *Training Products of Experts by Minimizing Contrastive divergence.* Neural computation: MIT Press, 2002. 1771-1800

[10] Carreira-Perpinan M A, Hinton G E. On contrastive divergence learning. In: *Proceedings of International Conference on Artificial Intelligence and Statistics*. 2005. 33-40

[11] Bengio Y, Delalleau O. *Justifying and Generalizing Contrastive*

Divergence. Neural computation: MIT Press, 2009. 1601-1621

[12] Cho K H, Raiko T, Ilin A. Parallel tempering is efficient for learning restricted Boltzmann machines. In: *Proceedings of International Joint Conference on Neural Networks*. IEEE, 2012. 1-8

[13] Tieleman T, Hinton G E. Using fast weights to improve persistent contrastive divergence. In: *Proceedings of International Conference on Machine Learning*. Montreal, Quebec, Canada: DBLP, 2009. 14-18

[14] Hyvärinen A. Some extensions of score matching. *Computational statistics & data analysis*, 2007, 51(5): 2499-2512

[15] Hyvärinen A, Dayan P. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 2005, 6(4):695-709

[16] Gutmann M, Hyvarinen A. Noise-contrastive estimation: a new estimation principle for unnormalized statistical models. In: *Proceedings of International Conference on Artificial Intelligence and Statistics*. 2010. 297-304

[17] Jarzynski C. Nonequilibrium equality for free energy differences. *Physical Review Letters*, 1997, 78(14): 2690-2693

[18] Hinton G E, Salakhutdinov R. Replicated softmax: an undirected topic model. In: *Proceedings of Neural Information Processing Systems*. 2009. 1607-1614

[19] Montufar G, Rauh J, Ay N. Expressive power and approximation errors of restricted boltzmann machines. In: *Proceedings of Neural Information Processing Systems*. 1998. 415-423

[20] Sutskever I, Hinton G, Taylor G. The recurrent temporal restricted boltzmann machine. In: *Proceedings of Neural Information Processing Systems*. 2008. 1601-1608

[21] Nair V, Hinton G E. Rectified linear units improve restricted boltzmann machines. In: *Proceedings of International Conference on Machine Learning*. 2010. 807-814

[22] Krizhevsky A, Hinton G. Factored 3-way restricted boltzmann machines for modeling natural images. In: *Proceedings of International Conference on Artificial Intelligence and Statistics*. 2010. 621-628

[23] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets. *Neural Compuper*, 2006, 18(7): 1527-1554

[24] Taylor G W, Hinton G E, Roweis S. Modeling human motion using binary latent variables. In: *Proceedings of Neural Information Processing Systems*. 2006. 1345-1352

[25] Dayan P, Hinton G E, Neal R M, Zemel R. The Helmholtz machine. *Neural computation*, 1995, 7(5): 889-904

[26] Hinton G E, Dayan P, Frey B J, Neal R M. The "wake-sleep" algorithm for unsupervised neural networks. *Science*, 1995, 268(5214): 1158-1161

[27] Mohamed A R, Yu D, Deng L. Investigation of full-sequence training of deep belief networks for speech recognition. In: *Proceedings of International Speech Communication Association*. 2010. 2846-2849

[28] Dahl G E, Yu D, Deng L, Acero A. Large vocabulary continuous speech recognition with context-dependent DBN-HMMS. In: *Proceedings of International Conference on Acoustics, Speech and Signal Processing*. 2011. 4688-4691

[29] Hinton G E, Salakhutdinov R. Using deep belief nets to learn covariance kernels for Gaussian processes. In: *Proceedings of Neural Information Processing Systems*. 2008: 1249-1256

[30] Hinton G E, Salakhutdinov R. Reducing the dimensionality of data with neural networks. *Science*, 2006, 313(5786):504-507

[31] Lee H, Grosse R, Ranganath R, Ng A Y. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: *Proceedings of Annual International Conference on Machine Learning*. 2009. 609-616

[32] Salakhutdinov R, Hinton G E. Deep Boltzmann machines. In: *Proceedings of International Conference on Artificial Intelligence and Statistics*. 2009. 448-455

[33] Montavon G, Müller K R. *Deep Boltzmann Machines and the Centering Trick*. Neural Networks: Tricks of the Trade, 2012. 621-637

[34] Melchior J, Fischer A, Wiskott L. How to center deep Boltzmann machines. *The Journal of Machine Learning Research*, 2016, 17(1): 3387-3447

[35] Goodfellow I, Mirza M, Courville A, Bengio Y. Multi-prediction deep Boltzmann machines. In: *Proceedings of Neural Information Processing Systems*. 2013. 548-556

[36] Salakhutdinov R. Learning in Markov random fields using tempered transitions. In: *Proceedings of Neural Information Processing Systems*. 2009. 1598-1606

[37] Paisley J, Blei D, Jordan M. Variational Bayesian inference with stochastic search. arXiv:1206.6430, 2012

[38] Theis L, Oord A, Bethge M. A note on the evaluation of generative models. arXiv:1511.01844, 2015

[39] Burda Y, Grosse R, Salakhutdinov R. Importance weighted autoencoders. arXiv:1509.00519, 2015

[40] Maaløe L, Sønderby C K, Sønderby S K, Winther O. Auxiliary deep generative models. arXiv:1602.05473, 2016

[41] Kingma D P, Mohamed S, Rezende D J, Welling M. Semi-supervised learning with deep generative models. In: *Proceedings of Neural Information Processing Systems*. 2014. 3581-3589

[42] Abbasnejad M E, Dick A, Hengel A. Infinite variational autoencoder

for semi-supervised learning. In: *Proceedings of Computer Vision and Pattern Recpgnition*. 2017. 781-790

[43] Kulkarni T D, Whitney W, Kohli P, Tenenbaum J. Deep convolutional inverse graphics network. In: *Proceedings of Neural Information Processing Systems*. 2015. 2539-2547

[44] Makhzani A, Shlens J, Jaitly N, Goodfellow I, Frey B. Adversarial autoencoders. arXiv:1511.05644, 2015

[45] Zhao S, Song J, Ermon S. Infovae: Information maximizing variational autoencoders. arXiv:1706.02262, 2017

[46] Higgins I, Matthey L, Pal A, et al. beta-vae: Learning basic visual concepts with a constrained variational framework. In: *Proceedings of International Conference on Learning Representations*. 2016

[47] Sønderby C K, Raiko T, Maaløe L, Sønderby S K, Winther O. Ladder variational autoencoders. In: *Proceedings of Neural Information Processing Systems*. 2016. 3738-3746

[48] Cai L, Gao H, Ji S. Multi-stage variational auto-encoders for coarse-to-fine image generation. arXiv:1705.07202, 2017

[49] Arjovsky M, Bottou L. Towards principled methods for training generative adversarial networks. arXiv:1701.04862, 2017

[50] Huszár F. How (not) to train your generative model: scheduled sampling, likelihood, adversary? arXiv:1511.05101, 2015

[51] Arjovsky M, Chintala S, Bottou L. Wasserstein GAN. arXiv:1701.07875, 2017

[52] Nowozin S, Cseke B, Tomioka R. f-GAN: Training generative neural samplers using variational divergence minimization. In: *Proceedings of Neural Information Processing Systems*. 2016. 271-279

[53] Gulrajani I, Ahmed F, Arjovsky M, Dumonlin V, Courville A C. Improved training of wasserstein GANs. In: *Proceedings of Neural Information Processing Systems*. 2017. 5767-5777

[54] Miyato T, Kataoka T, Koyama M, Yoshida Y. Spectral normalization for generative adversarial networks. In: *Proceedings of International Conference on Learning Representations*. 2018

[55] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks. In: *Proceedings of International Conference on Learning Representations*. 2015

[56] Karras T, Aila T, Laine S, Lehtinen J. Progressive growing of GANs for improved quality, Stability, and Variation. In: *Proceedings of International Conference on Learning Representations*. 2017

[57] Brock A, Donahue J, Simonyan K. Large scale GAN training for high fidelity natural image synthesis. In: *Proceedings of International Conference on Learning Representations*. 2019

[58] Mirza M, Osindero S. Conditional generative adversarial nets. arXiv:1411.1784, 2014

[59] Odena A, Olah C, Shlens J. Conditional image synthesis with auxiliary classifier GANs. In: *Proceedings of International Conference on Machine Learning*. 2017. 2642-2651

[60] Sricharan K, Bala R, Shreve M, Ding H, Saketh K, Sun J. Semi-supervised conditional GANs. arXiv:1708.05789, 2017

[61] Zhang H, Xu T, Li H. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In: *Proceedings of International Conference on Computer Vision*. 2016. 5908-5916

[62] Zhang H, Xu T, Li H, et al. StackGAN++: Realistic image synthesis with stacked generative adversarial networks. *Pattern Analysis and Machine Intelligence*, 2018, 41(8): 1947-1962

[63] Dinh L, Sohl-Dickstein J, Bengio S. Density estimation using Real NVP. In: *Proceedings of International Conference on Learning Representations*. 2016

[64] Kingma D P, Dhariwal P. Glow: Generative flow with invertible 1x1 convolutions. In: *Proceedings of Neural Information Processing Systems*. 2018. 10235-10244

[65] Behrmann J, Grathwohl W, Chen R, Duvenaud D, Jacobsen J H. Invertible residual networks. In: *Proceedings of International Conference on Machine Learning*. 2019. 573-582

[66] Rezende D J, Mohamed S. Variational inference with normalizing flows. In: *Proceedings of International Conference on Machine Learning*. 2015. 1530-1538

[67] Kingma D P, Salimans T, Jozefowicz R, Chen X, Sutskever I, Welling M. Improving variational inference with inverse autoregressive flow. In: *Proceedings of Neural Information Processing Systems*. 2016. 4743-4751

[68] Papamakarios G, Murray I, Pavlakou T. Masked autoregressive flow for density estimation. In: *Proceedings of Neural Information Processing Systems*. 2017. 2338-2347

[69] Frey B J, Brendan J F, Frey B J. *Graphical Models for Machine Learning and Digital Communication*. MIT press, 1998

[70] Bengio S, Bengio Y. Taking on the curse of dimensionality in joint distributions using neural networks. *IEEE Transactions on Neural Networks*, 2000, 11(3): 550-557

[71] Neal R M. Connectionist learning of belief networks. *Artificial intelligence*. 1992, 56(1): 71-113

[72] Bengio Y. Discussion of "the neural autoregressive distribution estimator". In: *Proceedings of International Conference on Artificial Intelligence and Statistics*. 2011. 431-439

[73] Raiko T, Li Y, Cho K, Bengio Y. Iterative neural autoregressive distribution estimator nade-k. In: *Proceedings of Neural Information Processing Systems*. 2014. 325-333

[74] Reed S, Oord A, Kalchbrenner N, et al. Parallel multiscale autoregressive density estimation. arXiv:1703.03664, 2017

[75] Uria B, Murray I, Larochelle H. A deep and tractable density estimator. In: *Proceedings of International Conference on Machine Learning*. 2014. 467-475

[76] Uria B, Côté M A, Gregor K, Murray I, Larochelle H. Neural autoregressive distribution estimation. *The Journal of Machine Learning Research*, 2016, 17(1): 7184-7220

[77] Oord A, Kalchbrenner N, Kavukcuoglu K. Pixel recurrent neural networks. arXiv:1601.06759, 2016

[78] Germain M, Gregor K, Murray I, et al. Made: Masked autoencoder for distribution estimation. In: *Proceedings of International Conference on Machine Learning*. 2015. 881-889

[79] Papamakarios G, Murray I, Pavlakou T. Masked autoregressive flow for density estimation. In: *Proceedings of Neural Information Processing Systems*. 2017. 2338-2347

[80] Socher R, Huang E H, Pennin J, Andrew Ng, Manning C. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In: *Proceedings of Neural Information Processing Systems*. 2011. 801-809

[81] Socher R, Pennington J, Huang E H, Andrew Ng, Manning C. Semi-supervised recursive autoencoders for predicting sentiment distributions. In: *Proceedings of the 2011 conference on empirical methods in natural language processing*. 2011. 151-161

[82] Gretton A, Borgwardt K M, Rasch M J, Schölkopf B, Smola A. A kernel two-sample test. *Journal of Machine Learning Research*, 2012, 13(5): 723-773

[83] Dziugaite G K, Roy D M, Ghahramani Z. Training generative neural networks via maximum mean discrepancy optimization. arXiv: 1505.03906, 2015

[84] Li C L, Chang W C, Cheng Y, Yang Y, Poczos B. MMD-GAN: Towards deeper understanding of moment matching network. In: *Proceedings of Neural Information Processing Systems*. 2017. 2203-2213

[85] Ren Y, Zhu J, Li J, Zhu J. Conditional generative moment-matching networks. In: *Proceedings of Neural Information Processing Systems*. 2016. 2928-2936

[86] Bengio Y, Yao L, Alain G, Vincent P. Generalized denoising auto-encoders as generative models. In: *Proceedings of Neural Information Processing Systems*. 2013. 899-907

[87] Rezende D J, Mohamed S, Wierstra D. Stochastic backpropagation and approximate inference in deep generative models. In: *Proceedings of International Conference on Machine Learning*. 2014

[88] Zöhrer M, Pernkopf F. General stochastic networks for classification. In: *Proceedings of Neural Information Processing Systems*. 2014. 2015-2023

[89] Chen X, Duan Y, Houthooft R, Schulman J, Sutskever I, Abbeel P. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In: *Proceedings of Neural Information Processing Systems*. 2016. 2172-2180

[90] Oord A, Vinyals O, Kavukcuoglu K. Neural discrete representation learning. In: *Proceedings of Neural Information Processing Systems*. 2017. 6306-6315

[91] Razavi A, Oord A, Vinyals O. Generating diverse high-fidelity images with VQ-VAE-2. In: *Proceedings of Neural Information Processing Systems*. 2019. 14866-14876

[92] Shaham T R, Dekel T, Michaeli T. SinGAN: Learning a generative model from a single natural image. In: *Proceedings of International Conference on Computer Vision*. 2019. 4570-4580

[93] Hinton G E. To recognize shapes, first learn to generate images. *Progress in brain research*, 2007, 165(6):535-547

[94] Taylor G W, Hinton G E, Roweis S T. Modeling human motion using binary latent variables. In: *Proceedings of International Conference on Computer Vision*. 2007. 1345-1352

[95] Mohamed A, Sainath T N, Dahl G, Ramabhadran B, Hinton G H, Picheny M. Deep belief networks using discriminative features for phone recognition. In: *Proceedings of International Conference on Acoustics, Speech and Signal Processing*. 2011. 22-27

[96] Ghahabi E O. Deep belief networks for i-vector based speaker recognition. In: *Proceedings of International Conference on Acoustics, Speech and Signal Processing*. 2014. 1700-1704

[97] Thomas D，Oliver B，Hermann N. A deep learning approach to machine transliteration. In: *Proceedings of the Fourth Workshop on Statistical Machine Translation*. 2009. 233-241

[98] Abdollahi B, Nasraoui O. Explainable restricted boltzmann Machines for collaborative filtering. In: *Proceedings of International Conference on Machine Learning*. 2016

[99] Xing L, Demertzis K, Yang J. Identifying data streams anomalies by evolving spiking restricted Boltzmann machines. *Neural Computing and Applications*, 2019, 1-15

[100] Zheng J, Fu X, Zhang G. Research on exchange rate forecasting based on deep belief network. *Neural Computing and Applications*, 2019, 31(1):573-582

[101] Lee H, Grosse R, Ranganath R, YNg A. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: *Proceedings of the 26th Annual International Conference on Machine Learning*. 2009. 609-616

[102] Salimans T, Kingma D, Welling M. Markov chain Monte Marlo and

variational inference: bridging the gap. In: *Proceedings of International Conference on Machine Learning*. 2015. 1218-1226

[103] Gregor K, Danihelka I, Graves A, Rezende D J, Wierstra D. Draw: A recurrent neural network for image generation. arXiv:1502.04623, 2015

[104] Chen R, Li X, Grosse R B, Duvenaud D K. Isolating sources of disentanglement in variational autoencoders. In: *Proceedings of Neural Information Processing Systems*. 2018. 2610-2620

[105] Walker J, Doersch C, Gupta A, Hebert M. An uncertain future: Forecasting from static images using variational autoencoders. In: *Proceedings of European Conference on Computer Vision*. 2016. 835-851

[106] Gregor K, Besse F, Rezende D J, Danihelka L, Wierstra D. Towards conceptual compression. In: *Proceedings of Neural Information Processing Systems*. 2016. 3549-3557

[107] Bowman S R, Vilnis L, Vinyals O, Dai A, Jozefowicz R, Bengio S. Generating sentences from a continuous space. arXiv:1511.06349, 2015

[108] Kusner M J, Paige B, Hernández-Lobato J M. Grammar variational autoencoder. In: *Proceedings of International Conference on Machine Learning*. 2017. 1945-1954

[109] Jang M, Seo S, Kang P. Recurrent neural network-based semantic variational autoencoder for sequence-to-sequence learning. *Information Sciences*, 2019, 490: 59-73

[110] Ravanbakhsh S, Lanusse F, Mandelbaum R, Schneider J, Poczos B. Enabling dark energy science with deep generative models of galaxy images. In: *Proceedings of Thirty-First AAAI Conference on Artificial Intelligence.* 2017

[111] Li X, She J. Collaborative variational autoencoder for recommender systems. In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* 2017. 305-314

[112] White T. Sampling generative networks. arXiv:1609.04468, 2016

[113] Gómez-Bombarelli R, Wei J N, Duvenaud D, et al. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 2018, 4(2): 268-276

[114] Tran L, Yin X, Liu X. Disentangled representation learning gan for pose-invariant face recognition. In: *Proceedings of Computer Vision and Pattern Recognition.* 2017. 1415-1424

[115] Huang R, Zhang S, Li T, He R. Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis. In: *Proceedings of International Conference on Computer Vision*. 2017. 2439-2448

[116] Ma L, Jia X, Sun Q, Schiele B, Tuytelaars T, Gool L V. Pose guided person image generation. In: *Proceedings of Neural Information Processing Systems*. 2017. 406-416

[117] Siarohin A, Sangineto E, Lathuilière S, Sebe N. Deformable GANs for pose-based human image generation. In: *Proceedings of Computer Vision and Pattern Recognition.* 2018. 3408-3416

[118] Chang H, Lu J, Yu F, Finkelstein A. PairedCycleGAN: Asymmetric style transfer for applying and removing makeup. In: *Proceedings of Computer Vision and Pattern Recognition.* 2018. 40-48

[119] Pumarola A, Agudo A, Martinez A M, Sanfeliu A, Moreno-Noguer F. Ganimation: Anatomically-aware facial animation from a single image. In: *Proceedings of European Conference on Computer Vision*. 2018. 818-833

[120] Donahue C, Lipton Z C, Balsubramani A, McAuley J. Semantically decomposing the latent spaces of generative adversarial networks. arXiv:1705.07904, 2017

[121] Shu Z, Sahasrabudhe M, Guler A R, Samaras D, Paragios N, Kokkinos I. Deforming autoencoders: Unsupervised disentangling of shape and appearance. In: *Proceedings of European Conference on Computer Vision*. 2018. 650-665

[122] Lu Y, Tai Y W, Tang C K. Attribute-guided face generation using conditional cycleGAN. In: *Proceedings of European Conference on Computer Vision*. 2018. 282-297

[123] Ledig C, Theis L, Huszár F, et al. Photo-realistic single image super-resolution using a generative adversarial network. In: *Proceedings of Computer Vision and Pattern Recognition.* 2017. 4681-4690

[124] Wang X, Yu K, Wu S, et al. EsrGAN: Enhanced super-resolution generative adversarial networks. In: *Proceedings of European Conference on Computer Vision*. 2018

[125] Zhu J Y, Park T, Isola P, AEfros A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: *Proceedings of International Conference on Computer Vision.* 2017. 2223-2232

[126] Bansal A, Ma S, Ramanan D, Sheikh Y. Recycle-GAN: Unsupervised video retargeting. In: *Proceedings of European Conference on Computer Vision*. 2018. 119-135

[127] Yuan Y, Liu S, Zhang J, Zhang Y, Dong C, Lin L. Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. In: *Proceedings of Computer Vision and Pattern Recognition.* 2018. 701-710

[128] Li J, Liang X, Wei Y, Xu T, Feng J, Yan S. Perceptual generative adversarial networks for small object detection. In: *Proceedings of Computer Vision and Pattern Recognition.* 2017. 1222-1230

[129] Bai Y, Zhang Y, Ding M, Ghanem B. Sod-mtGAN: Small object

detection via multi-task generative adversarial network. In: *Proceedings of European Conference on Computer Vision*. 2018. 206-221

[130] Ehsani K, Mottaghi R, Farhadi A. SeGAN: Segmenting and generating the invisible. In: *Proceedings of Computer Vision and Pattern Recognition*. 2018. 6144-6153

[131] Vondrick C, Pirsiavash H, Torralba A. Generating videos with scene dynamics. In: *Proceedings of Neural Information Processing Systems*. 2016. 613-621

[132] Villegas R, Yang J, Hong S, Lin X, Lee H. Decomposing motion and content for natural video sequence prediction. arXiv: 1706.08033, 2017

[133] Chan C, Ginosar S, Zhou T, A Efros A. Everybody dance now. In: *Proceedings of European Conference on Computer Vision*. 2019. 5933-5942

[134] Mathieu M, Couprie C, LeCun Y. Deep multi-scale video prediction beyond mean square error. arXiv:1511.05440, 2015

[135] Yu L, Zhang W, Wang J, Yu Y. SeqGAN: Sequence generative adversarial nets with policy gradient. In: *Proceedings of Thirty-First AAAI Conference on Artificial Intelligence*. 2017

[136] Saito Y, Takamichi S, Saruwatari H. Statistical parametric speech synthesis incorporating generative adversarial networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2017, 26(1): 84-96

[137] Pascual S, Bonafonte A, Serra J. SEGAN: Speech enhancement generative adversarial network. arXiv:1703.09452, 2017

[138] Wang J, Yu L, Zhang W, et al. IrGAN: A minimax game for unifying generative and discriminative information retrieval models. In: *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*. 2017. 515-524

[139] Lin K, Li D, He X, Sun MT. Adversarial ranking for language generation. In: *Proceedings of Neural Information Processing Systems*. 2017. 3155-3165

[140] Qiao T, Zhang J, Xu D, Tao D. MirrorGAN: Learning text-to-image generation by redescription. In: *Proceedings of Computer Vision and Pattern Recognition*. 2019. 1505-1514

[141] Schlegl T, Seeböck P, Waldstein S M, Schmidt-Erfurth U, Langs G. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: *Proceedings of International conference on information processing in medical imaging*. Springer, Cham, 2017. 146-157

[142] Xue Y, Xu T, Zhang H, Long LR, Huang X. SegAN: Adversarial network with multi-scale L1 loss for medical image segmentation. *Neuroinformatics*, 2018, 16(3-4): 383-392

[143] Yang Q, Yan P, Zhang Y, et al. Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss. *IEEE transactions on medical imaging*, 2018, 37(6): 1348-1357

[144] Zheng Z, Zheng L, Yang Y. Unlabeled samples generated by GAN improve the person re-identification baseline in vitro. In: *Proceedings of International Conference on Computer Vision*. 2017. 3754-3762

[145] Gupta A, Johnson J, Li F, Savarese S, Alahi A. Social GAN: Socially acceptable trajectories with generative adversarial networks. In: *Proceedings of Computer Vision and Pattern Recognition*. 2018. 2255-2264

[146] Jang E, Gu S, Poole B. Categorical reparameterization with gumbel-softmax. arXiv:1611.01144, 2016

[147] J Song, T He, L Gao, X Xu, Hanjalic A, Shen H T. Binary generative adversarial networks for image retrieval. In: *Proceedings of AAAI Conference on Artificial Intelligence*. 2018. 394–401

[148] Sohn K, Lee H, Yan X. Learning structured output representation using deep conditional generative models. In: *Proceedings of* Neural Information Processing Systems. 2015. 3483-3491

[149] Walker J, Doersch C, Gupta A, Hebert M. An uncertain future: Forecasting from static images using variational autoencoders. In: *Proceedings of European Conference on Computer Vision*. 2016. 835-851

[150] Xu W, Tan Y. Semisupervised text classification by variational autoencoder. *IEEE Transactions on Neural Networks and Learning Systems*, 2019, 31(1): 295-308

胡铭菲 中国石油大学 (北京) 自动化系博士研究生.主要研究方向为模式识别.
E-mail: hmfzsy@gmail.com
(**HU Ming-Fei** Ph. D. candidate at the Department of Automation, China University of Petroleum (Beijing). His research interest covers pattern recognition and intelligent system.)

刘建伟 中国石油大学 (北京) 自动化系副研究员. 主要研究方向为模式识别与智能系统, 先进控制.
E-mail: liujw@cup.edu.cn
(**LIU Jian-Wei** Associate researcher at the Department of Automation, China University of Petroleum (B-

eijing). His research interest covers pattern recognition and intelligent system, advanced control.）

左信 中国石油大学 (北京) 自动化系教授. 主要研究领域为智能控制.
E-mail: zuox@cup.edu.cn
(**Xin Zuo** Ph. D. Professor in the Department of Automation, College of Geophysics and Information Engineering, China University of Petroleum, Beijing Campus (CUP). His research interest is intelligent control.)