计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2014.20121044
51(1):1-16, 2014

玻尔兹曼机研究进展

刘建伟 刘 媛 罗雄麟
(中国石油大学自动化研究所 北京 102249)
(liujw@cup.edu.cn)

Research and Development on Boltzmann Machine

Liu Jianwei, Liu Yuan, and Luo Xionglin
(*Research Institute of Automation, China University of Petroleum, Beijing* 102249)

Abstract Being a new research area of machine learning, deep learning is good at solving some complex problems. As a representative of deep learning, Boltzmann machine is being widely studied. In view of the theoretical significance and practical value of Boltzmann machine, the research and development on Boltzmann machine are reviewed systematically. Firstly, some concepts about Boltzmann machine are summarized, which include configuration of Boltzmann machine as a single layer feedback network and classification of Boltzmann machine according to the topological structure, including general Boltzmann machine, semi-restricted Boltzmann machine and restricted Boltzmann machine. Secondly, the learning procedure of Boltzmann machine is reviewed in detail. Thirdly, several typical algorithms of Boltzmann machine are introduced, such as Gibbs sampling, parallel tempering, variational approach, stochastic approximation procedure, and contrastive divergence. Fourthly, the learning procedure of deep Boltzmann machine is described. New research and development on aspects of algorithms, models and practical application of Boltzmann machine in recent years are expounded then. Finally, the problems to be solved are pointed out.

Key words Boltzmann machine; visible unit; hidden unit; probability distribution; expected value; simulated annealing; Gibbs sampling; Markov chain

摘 要 深度学习是机器学习中的新兴研究领域,能够很好地用于解决目标识别、语言理解等复杂问题.玻尔兹曼机作为深度学习的典型代表近年来受到了广泛研究.鉴于玻尔兹曼机的理论意义和实际应用价值,系统综述了玻尔兹曼机的研究进展,首先概述了玻尔兹曼机的相关概念,包括单层反馈网络的结构和拓扑结构分类,然后详细描述了玻尔兹曼机的学习过程和几种典型学习算法,接着对近几年玻尔兹曼机研究的新进展进行了阐述,最后提出了玻尔兹曼机中有待进一步研究解决的问题.

关键词 玻尔兹曼机;可见单元;隐单元;概率分布;期望值;模拟退火;吉布斯采样;马尔可夫链

中图法分类号 TP181

最近深度学习(deep learning)作为一种学习复杂层次概率模型的方法广泛流行.深神经网络,例如深信任网络(deep belief network, DBN)和深玻尔兹曼机(deep Boltzmann machine, DBM)由多层神经元组成,已经应用于许多机器学习任务中,能够很好地解决一些复杂问题,在一定程度上提高了学习

收稿日期:2012-11-18;**修回日期**:2013-05-06
基金项目:国家"九七三"重点基础研究计划基金项目(2012CB720500);国家自然科学基金项目(21006127);中国石油大学(北京)基础学科研究基金项目(JCXK-2011-07)

性能.深神经网络由许多受限玻尔兹曼机(restricted Boltzmann machine, RBM)堆栈构成,RBM 的可见层神经元之间和隐层神经元之间假定无连接.深神经网络用层次无监督贪婪预训练方法分层预训练 RBM,将得到的结果作为监督学习训练概率模型的初始值,学习性能得到很大改善.无监督特征学习就是在 RBM 的复杂层次结构与海量数据集之间实现统计建模.通过无监督预训练使网络获得高阶抽象特征,并且提供较好的初始权值,将权值限定在对全局训练有利的范围内,使用层与层之间的局部信息进行逐层训练,注重训练数据自身的特性,能够减小对学习目标过拟合的风险,并避免深神经网络中误差累积传递过长的问题.RBM 由于表示力强、易于推理等优点被成功用作深神经网络的结构单元使用,在近些年受到广泛关注,作为实际应用,RBM 的学习算法已经在 MNIST 和 NORB 等数据集上显示出优越的学习性能.但是,训练 RBM 非常困难,研究表明如果选择的学习参数不适合数据集或 RBM 结构,传统的学习算法将无法正确建立数据分布模型.由此可见,RBM 的学习在深神经网络的学习中占据核心的地位.

RBM 是玻尔兹曼机(Boltzmann machine, BM)的一种特殊拓扑结构.BM 的原理起源于统计物理学,是一种基于能量函数的建模方法,能够描述变量之间的高阶相互作用,BM 的学习算法较复杂,但所建模型和学习算法有比较完备的物理解释和严格的数理统计理论作基础.BM 是一种对称耦合的随机反馈型二值单元神经网络,由可见层和多个隐层组成,网络节点分为可见单元(visible unit)和隐单元(hidden unit),用可见单元和隐单元来表达随机网络与随机环境的学习模型,通过权值表达单元之间的相关性.

以 Hinton 和 Ackley 两位学者为代表的研究人员从不同领域以不同动机同时提出 BM 学习机.

Hopfield 的开创性论文首次将退火伊辛模型(annealed Ising model)与物理学和统计力学联系起来[1];之后 Hinton 和 Sejnowski 第 1 次对用退火伊辛模型思想推理的 BM 的算法进行了描述[2-3],并将其应用于模式分类、预测和组合优化等方面;Hofstadter 也提出了将伊辛模型与退火吉布斯采样结合的想法[4-5],并且在 Smolensky[6] 的调和论中也发现类似的想法.

一般的神经网络学习都只允许网络的误差或能量函数梯度下降,是一种确定型的贪婪搜索法,易于陷入局部最小.Ackley 等人[7] 以模拟退火思想为基础,将随机机制引入到 Hopfield 网络模型中,提出了 BM 及其学习规则.BM 是 Hopfield 网络的 Monte Carlo 版本,是一种具有随机神经元动力学(称为 Glauber 动力学)的二值神经网络,神经元状态的概率分布是固定的并且由玻尔兹曼-吉布斯分布给出.BM 的神经元状态变化引入了统计概率和模拟退火算法(simulated annealing, SA),是一种全局最优搜索算法.

第 1 个用于大规模 BM 的有效学习过程由 Smolensky[6] 提出,他用非常受限的简化网络拓扑结构使推理变得容易.Smolensky 提出的 RBM 由 1 个可见神经元层和 1 个隐神经元层组成,由于隐层神经元之间没有相互连接并且隐层神经元独立于给定的训练样本,这使直接计算依赖数据的期望值变得容易,可见层神经元之间也没有相互连接,通过从训练样本得到的隐层神经元状态上执行马尔可夫链抽样过程,来估计独立于数据的期望值,并行交替更新所有可见层神经元和隐层神经元的值[8].

鉴于 BM 的理论意义和实际应用价值,本文系统综述 BM 的研究进展,为进一步深入研究 BM 理论和拓展其应用领域奠定了一定的基础.

1 玻尔兹曼机概述

BM 是由 Hinton 和 Sejnowski[2-3,9] 提出的一种随机递归神经网络,可以看作是一种随机生成的 Hopfield 网络,是能够通过学习数据的固有内在表示解决困难学习问题的最早的人工神经网络之一,因样本分布遵循玻尔兹曼分布而命名为 BM.BM 由二值神经元构成,每个神经元只取 1 或 0 这两种状态,状态 1 代表该神经元处于接通状态,状态 0 代表该神经元处于断开状态.在下面的讨论中单元和节点的意思相同,均表示神经元.

1.1 玻尔兹曼机作为单层反馈网络

BM 作为单层反馈网络时,形式上与离散型 Hopfield 网络[1] 一样,具有对称的连接权值,并且每个单元与自己之间无连接.BM 的结构用向量 $\boldsymbol{s}=(s_1,\cdots,s_i,\cdots,s_N)$ 表示,其中 s_i 是神经元 i 的状态,N 是神经元总数.

与 Hopfield 网络相同,BM 是一个基于能量函数的网络,不同的是 BM 的单元状态是随机的,每个神经元有 2 个状态:$s_i\in\{0,1\}$.网络的能量函数为

$$E=-\sum_{i=1}^{N}\sum_{j=1,i\neq j}^{N}w_{ij}s_is_j+\sum_{i=1}^{N}\theta_is_i, \qquad (1)$$

其中 w_{ij} 是神经元 i 和 j 之间的连接权值,θ_i 是神经元 i 的阈值.神经元 i 的状态为 0 与 1 所产生的能量差值为

$$\Delta E_i = E_{s_i=0} - E_{s_i=1} = \sum_{j=1}^{N} w_{ij} s_j - \theta_i. \quad (2)$$

$s_i=1$ 的概率为

$$p_{s_i=1} = \frac{1}{1+e^{-\Delta E_i/T}}, \quad (3)$$

其中 T 是系统的温度.相应地,$s_i=0$ 的概率为

$$p_{s_i=0} = 1 - p_{s_i=1} = \frac{e^{-\Delta E_i/T}}{1+e^{-\Delta E_i/T}}. \quad (4)$$

式(3)和式(4)相除得到:

$$\frac{p_{s_i=0}}{p_{s_i=1}} = e^{-\Delta E_i/T} = e^{-(E_{s_i=0}-E_{s_i=1})/T}. \quad (5)$$

将式(5)推广到网络中任意的 2 个全局状态 α 和 β,全局状态 α 和 β 出现的概率与相应的网络能量之间也满足:

$$\frac{p_\alpha}{p_\beta} = e^{-(E_\alpha - E_\beta)/T}. \quad (6)$$

式(6)为玻尔兹曼分布的表达式,因此将网络命名为 BM.

1.2 玻尔兹曼机按拓扑结构分类

多层网络的 BM 的神经元 $\boldsymbol{s}=[\boldsymbol{v}^{\mathrm{T}},\boldsymbol{h}^{\mathrm{T}}]^{\mathrm{T}}$ 通常分为可见单元和隐单元,可见单元向量 $\boldsymbol{v}=\{0,1\}^D$ 由输入节点和输出节点组成,是网络和环境之间的接触面,表示可观察的数据;隐单元向量 $\boldsymbol{h}=\{0,1\}^K$ 由隐节点组成,不与外界环境直接接触,主要作用是表示从数据中提取特征.

定义可见层节点与隐层节点$\{\boldsymbol{v},\boldsymbol{h}\}$之间的能量函数为

$$E(\boldsymbol{v},\boldsymbol{h},\Psi) = -\boldsymbol{v}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h} - \frac{1}{2}\boldsymbol{v}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{v} - \frac{1}{2}\boldsymbol{h}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{h} - \boldsymbol{v}^{\mathrm{T}}\boldsymbol{B} - \boldsymbol{h}^{\mathrm{T}}\boldsymbol{A}, \quad (7)$$

其中 $\Psi=\{\boldsymbol{W},\boldsymbol{L},\boldsymbol{R},\boldsymbol{B},\boldsymbol{A}\}$ 是模型参数;$\boldsymbol{W},\boldsymbol{L},\boldsymbol{R}$ 分别表示可见层节点到隐层节点、可见层节点到可见层节点和隐层节点到隐层节点的对称连接权,并且 $\boldsymbol{L}$ 和 $\boldsymbol{R}$ 的对角线元素为 0;$\boldsymbol{B},\boldsymbol{A}$ 分别为可见层节点和隐层节点的阈值.通过改变 $\boldsymbol{L}$ 和 $\boldsymbol{R}$ 的不同配置,可以得到一般 BM、半 RBM 和 RBM 这 3 种不同拓扑结构的 BM.

1.2.1 一般玻尔兹曼机

一般 BM 的可见层节点与隐层节点、可见层节点与可见层节点以及隐层节点与隐层节点之间都有连接权,如图 1 所示:

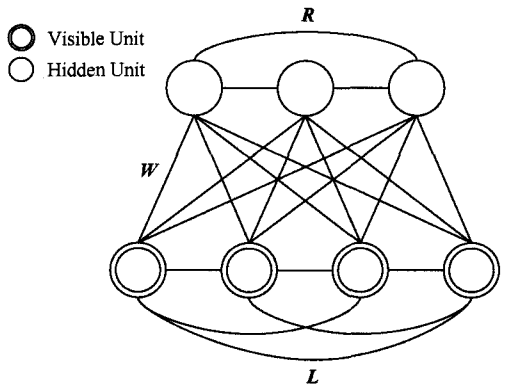

Fig. 1 Diagram of general Boltzmann machine.

图 1 一般玻尔兹曼机示意图

可见层节点与隐层节点$\{\boldsymbol{v},\boldsymbol{h}\}$之间的能量函数为

$$E(\boldsymbol{v},\boldsymbol{h},\Psi) = -\boldsymbol{v}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h} - \frac{1}{2}\boldsymbol{v}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{v} - \frac{1}{2}\boldsymbol{h}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{h} - \boldsymbol{v}^{\mathrm{T}}\boldsymbol{B} - \boldsymbol{h}^{\mathrm{T}}\boldsymbol{A} = -\sum_{i=1}^{D}\sum_{j=1}^{K} v_i w_{ij} h_j - \frac{1}{2}\sum_{i,k=1}^{D} v_i l_{ik} v_k - \frac{1}{2}\sum_{j,m=1}^{K} h_j r_{jm} h_m - \sum_{i=1}^{D} b_i v_i - \sum_{j=1}^{K} a_j h_j, \quad (8)$$

其中 v_i,h_j 是可见单元 i 和隐单元 j 的二值状态.

由于一般 BM 的结构比较复杂,可见层节点与可见层节点以及隐层节点与隐层节点之间均存在连接权,难以求解 BM 的能量最小值,算法复杂性高,需要耗费大量的网络训练与学习时间,因此很难对一般 BM 进行训练学习,在语音和图像识别上通常采用半 RBM 或 RBM 结构.

1.2.2 半受限玻尔兹曼机

半 RBM 的网络拓扑结构只有可见层节点与隐层节点及可见层节点与可见层节点之间的连接,而隐层节点与隐层节点之间没有连接,即 $\boldsymbol{R}=0$,如图 2 所示:

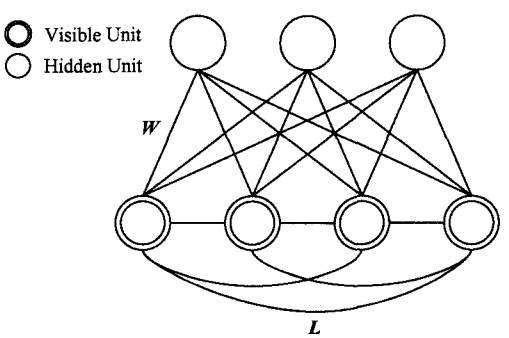

Fig. 2 Diagram of semi-restricted Boltzmann machine.

图 2 半受限玻尔兹曼机示意图

可见层节点与隐层节点$\{\boldsymbol{v},\boldsymbol{h}\}$之间的能量函数为

$$E(\boldsymbol{v},\boldsymbol{h},\Psi) = -\boldsymbol{v}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h} - \frac{1}{2}\boldsymbol{v}^{\mathrm{T}}\boldsymbol{L}\boldsymbol{v} - \boldsymbol{v}^{\mathrm{T}}\boldsymbol{B} - \boldsymbol{h}^{\mathrm{T}}\boldsymbol{A} = -\sum_{i=1}^{D}\sum_{i=1}^{K} v_i w_{ij} h_j - \frac{1}{2}\sum_{i,k=1}^{D} v_i l_{ik} v_k - \sum_{i=1}^{D} b_i v_i - \sum_{j=1}^{K} a_j h_j. \quad (9)$$

半 RBM 的结构相对于上面描述的一般 BM 简单，这意味着需要的网络训练与学习时间比一般 BM 少得多，但是在进行大型数据训练时仍需耗费大量的训练与学习时间.

1.2.3 受限玻尔兹曼机

RBM 是一个双向概率图模型，只有可见层节点与隐层节点之间有连接权，而可见层节点与可见层节点及隐层节点与隐层节点之间没有连接权，即 $\boldsymbol{L}=0$ 且 $\boldsymbol{R}=0$，如图 3 所示：

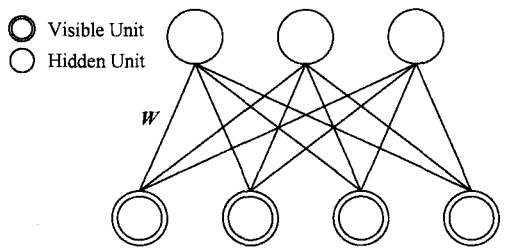

Fig. 3 Diagram of restricted Boltzmann machine.

图 3 受限玻尔兹曼机示意图

可见层节点与隐层节点$\{\boldsymbol{v},\boldsymbol{h}\}$之间的能量函数为

$$E(\boldsymbol{v},\boldsymbol{h},\Psi) = -\boldsymbol{v}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h} - \boldsymbol{v}^{\mathrm{T}}\boldsymbol{B} - \boldsymbol{h}^{\mathrm{T}}\boldsymbol{A} = -\sum_{i=1}^{D}\sum_{i=1}^{K} v_i w_{ij} h_j - \sum_{i=1}^{D} b_i v_i - \sum_{j=1}^{K} a_j h_j. \quad (10)$$

RBM 的结构相对于前面介绍的 2 种 BM 的网络拓扑结构更简单，因为各层内部神经元之间没有连接，这在很大程度上提高了网络训练与学习的效率. RBM 的一个应用实例是用于改进语音识别软件的性能. 由于 RBM 的结构简单，因此在构造 DBN 时，一般都是先构造出 RBM，然后将得到的 RBM 堆栈起来得到想要的 DBN.

2 玻尔兹曼机学习过程

BM 学习的目的是得到各神经元之间的连接权值，找到系统的最小全局能量结构. BM 可见层节点与隐层节点$\{\boldsymbol{v},\boldsymbol{h}\}$之间的能量函数见式(8)～(10)，将能量函数指数化并且正则化，得到可见单元向量 $\boldsymbol{v}$ 和隐单元向量 $\boldsymbol{h}$ 状态均为 1 的联合概率分布为

$$p(\boldsymbol{v},\boldsymbol{h},\Psi) = \frac{p^*(\boldsymbol{v},\boldsymbol{h},\Psi)}{Z(\Psi)} = \frac{1}{Z(\Psi)} \mathrm{e}^{-E(\boldsymbol{v},\boldsymbol{h},\Psi)}, \quad (11)$$

其中 p^* 表示非归一化概率，配分函数 $Z(\Psi) = \sum_{i=1}^{D}\sum_{j=1}^{K} \mathrm{e}^{-E(v_i,h_j,\Psi)}$ 是归一化项. 可见单元向量 $\boldsymbol{v}$ 状态为 1 的概率分布为

$$p(\boldsymbol{v},\Psi) = \frac{p^*(\boldsymbol{v},\Psi)}{Z(\Psi)} = \frac{1}{Z(\Psi)} \sum_{j=1}^{K} \mathrm{e}^{-E(\boldsymbol{v},h_j,\Psi)}. \quad (12)$$

令可见单元服从某种概率分布，隐单元向量 $\boldsymbol{h}$ 状态为 1 的条件概率分布为

$$p(\boldsymbol{h}|\boldsymbol{v},\Psi) = \prod_{j=1}^{K} p(h_j|\boldsymbol{v},\Psi). \quad (13)$$

令隐单元服从某种概率分布，可见单元向量 $\boldsymbol{v}$ 状态为 1 的条件概率分布为

$$p(\boldsymbol{v}|\boldsymbol{h},\Psi) = \prod_{i=1}^{D} p(v_i|\boldsymbol{h},\Psi). \quad (14)$$

隐单元 j 和可见单元 i 状态为 1 的条件概率分布为

$$p(h_j = 1|\boldsymbol{v},h_{-j}) = \sigma\Big(\sum_{i=1}^{D} w_{ij} v_i + \sum_{m=1,m\neq j}^{K} r_{jm} h_m + a_j\Big), \quad (15)$$

$$p(v_i = 1|\boldsymbol{h},v_{-i}) = \sigma\Big(\sum_{j=1}^{K} w_{ij} h_j + \sum_{k=1,k\neq i}^{D} l_{ik} v_k + b_i\Big), \quad (16)$$

其中 $\sigma(x) = \frac{1}{1+\mathrm{e}^{-x}}$ 是逻辑斯蒂函数，$\boldsymbol{x}_{-i}$ 表示向量 $\boldsymbol{x}$ 不包含 x_i 的部分.

最初的参数更新算法由 Hinton 和 Sejnowski[2] 提出，用极大似然估计从训练样本中学习 BM 的参数. 式(12)取对数，并分别对 $\boldsymbol{W},\boldsymbol{L},\boldsymbol{R}$ 求偏微分，得到参数更新规则：

$$\frac{\partial \ln p(\boldsymbol{v},\Psi)}{\partial \boldsymbol{W}} = \Delta\boldsymbol{W} = \varepsilon(E_{p_{\mathrm{data}}}[\boldsymbol{v}\boldsymbol{h}^{\mathrm{T}}] - E_{p_{\mathrm{model}}}[\boldsymbol{v}\boldsymbol{h}^{\mathrm{T}}]), \quad (17)$$

$$\frac{\partial \ln p(\boldsymbol{v},\Psi)}{\partial \boldsymbol{L}} = \Delta\boldsymbol{L} = \varepsilon(E_{p_{\mathrm{data}}}[\boldsymbol{v}\boldsymbol{v}^{\mathrm{T}}] - E_{p_{\mathrm{model}}}[\boldsymbol{v}\boldsymbol{v}^{\mathrm{T}}]), \quad (18)$$

$$\frac{\partial \ln p(\boldsymbol{v},\Psi)}{\partial \boldsymbol{R}} = \Delta\boldsymbol{R} = \varepsilon(E_{p_{\mathrm{data}}}[\boldsymbol{h}\boldsymbol{h}^{\mathrm{T}}] - E_{p_{\mathrm{model}}}[\boldsymbol{h}\boldsymbol{h}^{\mathrm{T}}]), \quad (19)$$

其中 ε 是学习率；$E_{p_{\mathrm{data}}}[\cdot]$是将可见单元状态值取为训练样本值时得到的依赖数据的期望值，是整个网络单元的联合概率分布 $p_{\mathrm{data}}(\boldsymbol{v},\boldsymbol{h},\Psi) = p(\boldsymbol{h}|\boldsymbol{v},\Psi) \times p_{\mathrm{data}}(\boldsymbol{v})$的期望值；$p_{\mathrm{data}}(\boldsymbol{v}) = \frac{1}{N}\sum_{i=1}^{N} \delta(\boldsymbol{v}-v_i)$表示数

据的经验概率分布;$E_{p_{\text{model}}}[\cdot]$是未将训练样本值输入到可见单元(即可见单元为随机二值状态)时得到的独立于数据的期望值,是式(11)定义的模型联合概率分布的期望值.

下面以计算$\boldsymbol{W}$为例,介绍BM的具体学习过程.

BM有同步和异步两种类型的工作方式,通常所研究的是异步工作方式.从上面的BM过程可以看出,BM处于某一状态的概率取决于在该状态下的能量,能量越低概率越大;同时该概率还取决于温度参数,温度越大不同状态下的概率差异越小,越容易跳出能量的局部极小值进入全局最小值附近.通常采用SA算法来有效搜索网络能量函数的全局最小值,一开始采用较高的温度进行粗调,使网络快速接近热平衡(thermal equilibrium),粗略找到全局状态空间的整体结构框架,然后逐渐降低温度进行微调,寻找整体结构框架极小值中的最小值.

BM通过下面的训练过程完成学习:在训练过程中,环境将所有可见单元约束到特定状态;在测试过程中,环境可以约束可见单元的任意子集的状态.在上面所有过程中,环境始终不约束隐单元的状态.

令$\boldsymbol{v}_\alpha$表示D维可见单元向量,$\boldsymbol{h}_\beta$表示K维隐单元向量,$\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta$表示整个网络的状态向量.BM的学习过程分为两个阶段:

1) 约束学习阶段,也称为正阶段.将可见单元状态值取为训练样本值,采样得到隐单元.可见单元向量$\boldsymbol{v}_\alpha$出现第α个状态的概率$p^+(\boldsymbol{v}_\alpha)$由训练样本集决定,可以用式(3)计算隐单元向量$\boldsymbol{h}_\beta$出现第$\beta$个状态的概率$p(\boldsymbol{h}_\beta|\boldsymbol{v}_\alpha,\boldsymbol{\Psi})=\prod_{j=1}^{K} p(h_j|v_\alpha,\boldsymbol{\Psi})$.$v_\alpha$是可见单元向量的第$\alpha$个状态,最多有$2^D$个,因此$p^+(\boldsymbol{v}_\alpha)$也有$2^D$个,当训练样本数少于$2^D$个时,可取其余未给定样本出现的概率为0.

2) 自由学习阶段,也称为负阶段.网络自由运行,不约束任何可见单元的状态,从当前模型采样得到可见单元和隐单元状态值,即单元的状态不是由训练样本决定的,系统稳定时可见单元向量$\boldsymbol{v}_\alpha$出现第α个状态的概率$p^-(\boldsymbol{v}_\alpha)$由模型决定.

BM的学习是通过调整连接权矩阵,使模型定义的概率分布$p^-(\boldsymbol{v}_\alpha)$尽可能地与训练样本集定义的概率分布$p^+(\boldsymbol{v}_\alpha)$相一致.用K-L离差(Kullback-Leibler divergence)度量这两个概率分布的接近程度:

$$G=\sum_{\alpha=1}^{2^D} p^+(\boldsymbol{v}_\alpha)\ln\left(\frac{p^+(\boldsymbol{v}_\alpha)}{p^-(\boldsymbol{v}_\alpha)}\right), \tag{20}$$

其中G取非负值,仅当$p^-(\boldsymbol{v}_\alpha)=p^+(\boldsymbol{v}_\alpha)$时$G=0$.

学习BM的目的是学习连接权矩阵,使出现概率最高的全局状态得到最低的能量,因此问题变为最小化似然函数的过程,寻找连接权矩阵,使G取得最小值.

设$p^+(\boldsymbol{v}_\alpha)$均为已知,并且:

$$\sum_{\alpha=1}^{2^D} p^+(\boldsymbol{v}_\alpha)=1, \tag{21}$$

$p^+(\boldsymbol{v}_\alpha)$是由训练样本集决定的概率分布,与模型无关,因此独立于连接权值w_{ij}.网络在平衡态自由运行时,可见单元向量$\boldsymbol{v}_\alpha$出现第α个状态的概率为

$$p^-(\boldsymbol{v}_\alpha)=\sum_{\beta=1}^{2^K} p^-(\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta)=\frac{\sum_{\beta=1}^{2^K} \mathrm{e}^{-E_{\alpha\beta}/T}}{\sum_{\lambda=1}^{2^D}\sum_{\mu=1}^{2^K} \mathrm{e}^{-E_{\lambda\mu}/T}}, \tag{22}$$

其中$E_{\alpha\beta}$表示整个网络的状态为$\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta$时系统的能量,$E_{\lambda\mu}$表示整个网络的状态为$\boldsymbol{v}_\lambda \wedge \boldsymbol{h}_\mu$时系统的能量.为了简化问题,设神经元阈值$\theta=0$,有:

$$E_{\alpha\beta}=-\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij} s_i^{\alpha\beta} s_j^{\alpha\beta}, \tag{23}$$

$$E_{\lambda\mu}=-\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij} s_i^{\lambda\mu} s_j^{\lambda\mu}, \tag{24}$$

其中$s_i^{\alpha\beta}$表示整个网络的状态为$\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta$时$N$维状态向量的第$i$个分量,$s_i^{\lambda\mu}$表示整个网络的状态为$\boldsymbol{v}_\lambda \wedge \boldsymbol{h}_\mu$时$N$维状态向量的第$i$个分量.对式(23)取指数并求导,得到:

$$\frac{\partial \mathrm{e}^{-E_{\alpha\beta}/T}}{\partial w_{ij}}=\frac{1}{T} s_i^{\alpha\beta} s_j^{\alpha\beta} \mathrm{e}^{-E_{\alpha\beta}/T}. \tag{25}$$

对式(22)求导,得到:

$$\frac{\partial p^-(\boldsymbol{v}_\alpha)}{\partial w_{ij}}=\left(\frac{1}{T}\sum_{\beta=1}^{2^K} \mathrm{e}^{-E_{\alpha\beta}/T} s_i^{\alpha\beta} s_j^{\alpha\beta} \sum_{\lambda=1}^{2^D}\sum_{\mu=1}^{2^K} \mathrm{e}^{-E_{\lambda\mu}/T} - \frac{1}{T}\sum_{\lambda=1}^{2^D}\sum_{\mu=1}^{2^K} \mathrm{e}^{-E_{\lambda\mu}/T} s_i^{\lambda\mu} s_j^{\lambda\mu} \sum_{\beta=1}^{2^K} \mathrm{e}^{-E_{\alpha\beta}/T}\right) \Big/ \left(\sum_{\lambda=1}^{2^D}\sum_{\mu=1}^{2^K} \mathrm{e}^{-E_{\lambda\mu}/T}\right)^2 =$$

$$\frac{1}{T}\left(\sum_{\beta=1}^{2^K} p^-(\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta) s_i^{\alpha\beta} s_j^{\alpha\beta} - p^-(\boldsymbol{v}_\alpha) \sum_{\lambda=1}^{2^D}\sum_{\mu=1}^{2^K} p^-(\boldsymbol{v}_\lambda \wedge \boldsymbol{h}_\mu) s_i^{\lambda\mu} s_j^{\lambda\mu}\right). \tag{26}$$

对式(20)求导,得到:

$$\frac{\partial G}{\partial w_{ij}}=\sum_{\alpha=1}^{2^D} \frac{p^+(\boldsymbol{v}_\alpha)}{p^-(\boldsymbol{v}_\alpha)} \frac{\partial p^-(\boldsymbol{v}_\alpha)}{\partial w_{ij}} =$$

$$-\frac{1}{T}\sum_{\alpha=1}^{2^D} \frac{p^+(\boldsymbol{v}_\alpha)}{p^-(\boldsymbol{v}_\alpha)}\left(\sum_{\beta=1}^{2^K} p^-(\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta) s_i^{\alpha\beta} s_j^{\alpha\beta} - p^-(\boldsymbol{v}_\alpha)\sum_{\lambda=1}^{2^D}\sum_{\mu=1}^{2^K} p^-(\boldsymbol{v}_\lambda \wedge \boldsymbol{h}_\mu) s_i^{\lambda\mu} s_j^{\lambda\mu}\right). \tag{27}$$

因为：

$$p^+(\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta)=p^+(\boldsymbol{h}_\beta|\boldsymbol{v}_\alpha)p^+(\boldsymbol{v}_\alpha), \tag{28}$$

$$p^-(\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta)=p^-(\boldsymbol{h}_\beta|\boldsymbol{v}_\alpha)p^-(\boldsymbol{v}_\alpha), \tag{29}$$

$$p^+(\boldsymbol{h}_\beta|\boldsymbol{v}_\alpha)=p^-(\boldsymbol{h}_\beta|\boldsymbol{v}_\alpha), \tag{30}$$

所以有：

$$p^-(\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta)\frac{p^+(\boldsymbol{v}_\alpha)}{p^-(\boldsymbol{v}_\alpha)}=p^+(\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta). \tag{31}$$

将式(21)和式(31)带入式(27)，得到：

$$\frac{\partial G}{\partial w_{ij}}=-\frac{1}{T}\left(\sum_{\alpha=1}^{2^D}\sum_{\beta=1}^{2^K}p^+(\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta)s_i^{\alpha\beta}s_j^{\alpha\beta}-\sum_{\lambda=1}^{2^D}\sum_{\mu=1}^{2^K}p^-(\boldsymbol{v}_\lambda \wedge \boldsymbol{h}_\mu)s_i^{\lambda\mu}s_j^{\lambda\mu}\right)=-\frac{1}{T}(p_{ij}^+-p_{ij}^-), \tag{32}$$

其中 $p_{ij}^+=\sum_{\alpha=1}^{2^D}\sum_{\beta=1}^{2^K}p^+(\boldsymbol{v}_\alpha \wedge \boldsymbol{h}_\beta)s_i^{\alpha\beta}s_j^{\alpha\beta}$ 是环境约束可见单元的状态时，两个单元 i 和 j 都处于接通状态的平均概率；$p_{ij}^-=\sum_{\lambda=1}^{2^D}\sum_{\mu=1}^{2^K}p^-(\boldsymbol{v}_\lambda \wedge \boldsymbol{h}_\mu)s_i^{\lambda\mu}s_j^{\lambda\mu}$ 是网络自由运行时，两个单元 i 和 j 都处于接通状态的平均概率.

最小化 G 的步骤是，当网络处于热平衡状态时观察 p_{ij}^+ 和 p_{ij}^-，并且按式(33)改变每个连接权值：

$$\Delta w_{ij}=\varepsilon(p_{ij}^+-p_{ij}^-), \tag{33}$$

其中 $\varepsilon>0$ 用于衡量每个连接权值改变的大小.

BM 的学习步骤如下：

算法 1. 玻尔兹曼机学习步骤.

1) 随机设定网络的初始连接权值 $w_{ij}(0)$ 及初始高温；

2) 按照已知概率 $p(\boldsymbol{v}_\alpha)$ 依次给定训练样本，在训练样本的约束下按照 SA 算法运行网络直到平衡状态，统计出各个 p_{ij}^+，在无约束条件下按同样的步骤运行网络相同的次数，统计出各个 p_{ij}^-；

3) 按式(34)修改每个权值 w_{ij}：

$$w_{ij}(k+1)=w_{ij}(k)+\Delta w_{ij}. \tag{34}$$

重复上面的步骤，直到 $p_{ij}^+-p_{ij}^-$ 小于某个预设的容限.

BM 的学习规则只用局部可用的信息，权值的改变仅与相互连接的 2 个单元有关，并且每个权值的最优值依赖其他所有权值. 学习算法中有许多自由参数和变量，ε 决定梯度下降中每一步的大小，估计 p_{ij}^+ 和 p_{ij}^- 所用的时间对学习过程有重要的影响. 实际系统在估计 p_{ij}^+ 和 p_{ij}^- 的过程中必然存在一些噪声，导致了 G 值的偶然上升. 用一个小的 ε 值，或者用更长的时间计算期望值，都能够减小噪声对估计的影响，使 SA 算法搜索 G 的最小值变得相对容易.

BM 的学习目的是使模型的概率分布等于训练样本集的概率分布，来得到网络的连接权值. 采用上面介绍的学习过程需要计算许多梯度值，但是由于模型太复杂，包含大量的变量，难以计算出这些梯度值，因此常用采样方法来近似计算式(17)～(19)中的期望值，避免了梯度的计算. 训练样本中有些样例的值已知，有些样例的值未知，通常用训练样本的子集(称为微批次)的均值近似代替期望值 $E_{p_{\text{data}}}[\cdot]$. 采用一些算法近似学习期望值 $E_{p_{\text{model}}}[\cdot]$：随机选择微批次初始化可见单元的状态，从条件概率分布式(13)中采样隐单元状态，然后从条件概率分布式(14)中重新采样可见单元状态来重新构造可见单元，重复上述过程，用重新采样的可见单元状态和隐单元状态近似代替期望值，如图 4 所示：

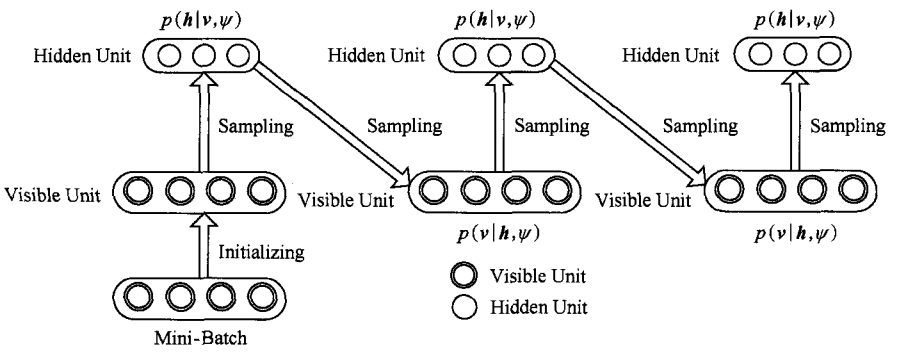

Fig. 4 Diagram of Boltzmann machine learning process.

图 4 玻尔兹曼机学习过程示意图

3 玻尔兹曼机学习算法

典型 BM 学习算法主要有吉布斯采样法(Gibbs sampling)[2,10-11]、平行回火法(parallel tem-pering, PT)[12]、变分近似法(variational approach)[13]、随机近似法(stochastic approximation procedure, SAP)[14-15]、对比离差算法(contrastive divergence, CD)[8,16]、持续

对比离差算法(persistent contrastive divergence, PCD)[15]和快速持续对比离差算法(fast persistent contrastive divergence, FPCD)[17]等,下面对这几种算法进行讨论.

3.1 吉布斯采样法

Hinton 和 Sejnowski[2,10-11]提出了一种概率推理的方法,用吉布斯采样近似期望值.吉布斯采样法是马尔可夫链算法的一种,也称为马尔可夫链蒙特卡罗法(Markov chain Monte Carlo, MCMC).给定一个 N 维的随机向量 $\boldsymbol{x}=(x_1,\cdots,x_i,\cdots,x_N)$,无法求得关于 $\boldsymbol{x}$ 的联合概率分布 $p(\boldsymbol{x})$,但是知道给定 $\boldsymbol{x}$ 的其他分量时 x_i 的条件概率分布 $p(x_i|\boldsymbol{x}_{-i})$,其中 $\boldsymbol{x}_{-i}$表示向量 $\boldsymbol{x}$ 不包含 x_i 的部分.可以从 $\boldsymbol{x}$ 的任意状态$[x_1(0),\cdots,x_i(0),\cdots,x_N(0)]$开始,利用条件概率分布 $p(x_i|x_{-i})$,迭代对其分量依次进行采样.随着采样次数的增加,随机变量$[x_1(k),\cdots,x_i(k),\cdots,x_N(k)]$的概率分布将以 k 的几何级数的速度收敛于 $\boldsymbol{x}$ 的联合概率分布$p(\boldsymbol{x})$,可以在联合概率分布 $p(\boldsymbol{x})$ 未知的情况下对其进行采样.在训练 BM 的每个迭代过程中,设置一个收敛到模型分布的马尔可夫链并将其运行到平衡状态,用马尔可夫链近似期望值 $E_{P_{\text{model}}}[\cdot]$.

这种算法的优点是通用性比较好,缺点是计算代价较高,运行缓慢,在每次迭代过程中都要等到每个马尔可夫链达到平稳分布.

3.2 平行回火法

PT 法是最近提出的用于代替吉布斯采样法估计期望值 $E_{p_{model}}[\cdot]$的方法.用有索引温度参数的目标分布 $p_{t_i}(\boldsymbol{x})=\dfrac{\mathrm{e}^{-E(\boldsymbol{x})/t_i}}{Z(t_i)}$进行抽样,PT 法的基本思想是模型以不同的温度平行运行多个 MCMC 链,每个 MCMC 链在一个有序序列温度 t_i 上,$t_0=1<t_1<\cdots<t_i<\cdots<t_{T-1}<t_T=\tau$,其中 $t_0=1$ 是目标分布采样温度,$t_T=\tau$ 是高温.引入交叉温度状态互换,每个运行步骤中,以交换概率 $\gamma=\dfrac{p_{t_i}(x_{i+1})p_{t_{i+1}}(x_i)}{p_{t_i}(x_i)p_{t_{i+1}}(x_{i+1})}$交换温度 t_i 和 t_{i+1}上运行的 2 个近邻链之间的样本.对于吉布斯分布类,有 $\gamma=\mathrm{e}^{(\beta_i-\beta_{i+1})\cdot(E(x_i)-E(x_{i+1}))}$,其中 β_i 是逆温度参数.在每个梯度更新步骤中,在随机选择的近邻链之间执行状态互换之后,所有 MCMC 链执行一步吉布斯采样.在更高的温度下 MCMC 链对应的模型分布更加扩散,因此可以产生各种各样的样本,有利于更好地搜索状态空间.

3.3 变分近似法

在变分学习中,对每个训练样本可见单元向量 $\boldsymbol{v}$,用近似后验分布 $q(\boldsymbol{h}|\boldsymbol{v},\boldsymbol{\mu})$替换隐单元向量上的真实后验分布 $p(\boldsymbol{h}|\boldsymbol{v},\boldsymbol{\Psi})$,BM 模型的对数似然函数有下面形式的变分下界:

$$\ln p(\boldsymbol{v},\boldsymbol{\Psi})\geqslant\sum_{j=1}^{K}q(h_j|\boldsymbol{v},\boldsymbol{\mu})\ln p(\boldsymbol{v},\boldsymbol{h},\boldsymbol{\Psi})+H(q)=\ln p(\boldsymbol{v},\boldsymbol{\Psi})-KL[q(\boldsymbol{h}|\boldsymbol{v},\boldsymbol{\mu})\|p(\boldsymbol{v}|\boldsymbol{h},\boldsymbol{\Psi})],\quad(35)$$

其中 $H(\cdot)$是熵函数.

变分学习具有很好的特性,在最大化训练样本的对数似然函数的同时,还能找到最小化近似后验分布和真实后验分布之间 K-L 离差的参数.用朴素平均场方法,选择完全因式分解的分布来近似真实后验分布:$q(\boldsymbol{h},\boldsymbol{\mu})=\prod_{j=1}^{K}q(h_j)$,其中 $q(h_j=1)=\mu_j$.训练样本的对数似然函数的下界有如下形式:

$$\ln p(\boldsymbol{v},\boldsymbol{\Psi})\geqslant\frac{1}{2}\sum_{i,k=1}^{D}v_i l_{ik}v_k+\frac{1}{2}\sum_{j,m=1}^{K}\mu_j r_{jm}\mu_m+\sum_{i=1}^{D}\sum_{j=1}^{K}v_i w_{ij}\mu_j-\ln Z(\boldsymbol{\Psi})+\sum_{j=1}^{K}[\mu_j\ln\mu_j+(1-\mu_j)\ln(1-\mu_j)].\quad(36)$$

固定 $\boldsymbol{\Psi}$,最大化式(36)来学习变分参数 $\boldsymbol{\mu}$,得到平均场不动点方程:

$$\mu_j\leftarrow\sigma\Big(\sum_{i=1}^{D}w_{ij}v_i+\sum_{m=1,m\neq j}^{K}r_{jm}\mu_m\Big).\quad(37)$$

接着给定变分参数 $\boldsymbol{\mu}$,用吉布斯采样法、PT 法和 SAP 法等其他算法更新模型参数 $\boldsymbol{\Psi}$.

变分近似法能很好地估计依赖数据的期望值 $E_{p_{\text{data}}}[\cdot]$,不能用于近似模型期望值 $E_{P_{\text{model}}}[\cdot]$,因为式(17)~(19)中的负号改变变分参数,使近似后验分布和真实后验分布之间的离差最大.

3.4 随机近似法

SAP 法属于广义的 Robbins-Monro 式随机近似法,用于近似期望值 $E_{p_{\text{model}}}[\cdot]$.令 $\boldsymbol{\Psi}_t$ 和 $\boldsymbol{s}^t$ 为当前的参数和状态,按下面步骤依次更新 $\boldsymbol{s}^t$ 和 $\boldsymbol{\Psi}^t$:1)给定 $\boldsymbol{s}^t$,从转换因子 $T_{\Psi_t}(\boldsymbol{s}^{t+1}\leftarrow\boldsymbol{s}^t)$采样得到 1 个新的状态 $\boldsymbol{s}^{t+1}$,并且保持概率 p_{Ψ_t} 不变;2)用 $\boldsymbol{s}^{t+1}$的估计值替换求解困难的模型期望值 $E_{P_{\text{model}}}[\cdot]$,然后执行梯度下降,得到新的参数 $\boldsymbol{\Psi}_{t+1}$.

SAP 法学习过程可行的主要原因是:当学习率相对于马尔可夫链的混合速率变得足够小时,持续马尔可夫链将会一直接近平稳分布.对于成功的参数更新,从持续马尔可夫链采集的数据将会高度关联,但是当学习率足够小时,在参数足以改变估计值之前马尔可夫链将会到达混合时间.

3.5 对比离差算法

Hinton[8]提出的CD算法用于有效实现BM的学习,有效避免了计算对数似然函数梯度的麻烦,尤其适用于RBM[16]. CD算法用估计概率分布与真实概率分布之间的距离度量函数作为度量准则,在近似的概率分布差异度量函数上求解最小化. 用从对应微批次的每个训练样本开始运行 n 步吉布斯采样,得到的样本计算 $E_{p_{\text{model}}}[\cdot]$. 权值的CD梯度近似为

$$\Delta W \approx \varepsilon(E_{p_{\text{data}}}[\boldsymbol{v}\boldsymbol{h}^{\mathrm{T}}]-E_{p_n}[\boldsymbol{v}\boldsymbol{h}^{\mathrm{T}}]), \quad (38)$$

其中 p_n 为 n 步吉布斯采样后得到的概率分布. 从样本数据概率分布 $p_0=p_{\text{data}}$ 上的马尔可夫链开始,运行少量步骤的马尔可夫链,用吉布斯采样或混合蒙特卡罗作为马尔可夫链的转换因子,在很大程度上减少了每个梯度下降步骤的计算量,能得到好的参数估计. 通常取 $n=1$ 能够得到有效的学习.

尽管CD算法应用广泛,但是不能产生最好的对数似然梯度近似. 通过实验发现在训练开始时CD算法表现得很好,随着训练过程的进行及参数值的增加,马尔可夫链的遍历性下降,因此对梯度的近似质量下降.

3.6 持续对比离差算法

Tieleman[15]提出的里程碑式算法——PCD算法弥补了CD算法无法极大化似然函数的缺陷. 大量实验表明,与CD算法相比,经PCD算法训练的RBM具有更好的学习模型的能力.

PCD算法从持续马尔可夫链得到负阶段样本来近似梯度. 令 t 步的持续马尔可夫链状态为 $\boldsymbol{v}_t$,梯度更新规则为

$$\Delta W \approx \varepsilon(E[\boldsymbol{v}\tilde{\boldsymbol{h}}_0^{\mathrm{T}}]-E[\tilde{\boldsymbol{v}}_{t+k}\tilde{\boldsymbol{h}}_{t+k}^{\mathrm{T}}]), \quad (39)$$

其中 $(\tilde{\boldsymbol{v}}_{t+k},\tilde{\boldsymbol{h}}_{t+k})$ 为从状态 $\boldsymbol{v}_t$ 开始经过 k 个持续马尔可夫链步骤得到的样本. 似然函数梯度的估计可以看作损失函数 $KL(p\|p_\Psi)-KL(p_{\Psi,t}\|p_\Psi)$ 的梯度,其中 p 是训练数据分布,p_Ψ 是模型分布,$p_{\Psi,t}$ 是 t 步的持续马尔可夫链形成的概率分布,后一项本质上是由于仅运行 k 步持续马尔可夫链代替运行整个马尔可夫链到平衡态而引起的误差项. 当马尔可夫链有更快的混合时间时,这个误差项几乎可以忽略,然而由于学习率增大或训练时间变长,使得马尔可夫链的遍历性下降,产生不稳定的误差.

3.7 快速持续对比离差算法

Tieleman和Hinton[17]在PCD算法的基础上引入单独的混合机制,提出FPCD算法,对PCD算法的混合性质进行改进,使其性能随着训练过程的进行不会恶化. FPCD算法用2组权值,除了一般的模型标准参数 $\boldsymbol{W}$(也称为慢速权值)用于正阶段和负阶段外,负阶段还用另一组快速权值 $\boldsymbol{W}_{\mathrm{f}}$. 慢速权值用来估计数据的期望值,快速权值的作用是提高样本的混合速率. 用参数为 $\boldsymbol{W}+\boldsymbol{W}_{\mathrm{f}}$ 的持续马尔可夫链更新样本. 为了提高混合速率,快速权值有一个固定的比较大的学习率,与慢速权值用的学习率不同. 快速权值在模型定义的势能面顶端产生动态叠加,通过减小最新采样单元的产生概率提高混合速率. 对快速权值用大学习率可以得到更快的混合时间. 根据PCD算法的梯度更新快速权值和慢速权值,快速权值中的改变以指数衰减到0来保证它们的作用仅仅是临时的.

4 深玻尔兹曼机

DBM是BM类模型的一种特殊子类,是有对称耦合随机二值单元的网络,是有无向层连接的马尔可夫随机场. 与仅有1个隐层的RBM不同,DBM包含多个隐层,如图5所示:

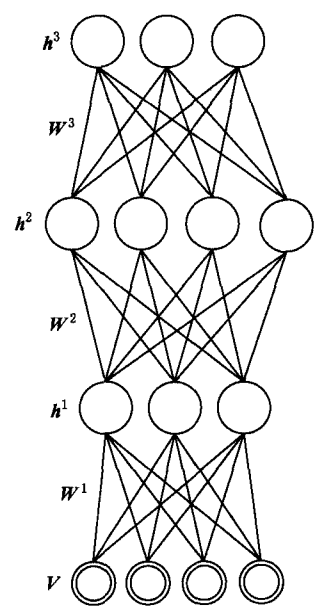

Fig. 5 Boltzmann machine with three hidden layers.

图5 具有3层隐层的玻尔兹曼机

下面以没有层内连接的具有3个隐层的DBM为例,介绍DBM的学习过程. 定义可见层节点与隐层节点 $\{\boldsymbol{v},\boldsymbol{h}^1,\boldsymbol{h}^2,\boldsymbol{h}^3\}$ 之间的能量函数为:

$$E(\boldsymbol{v},\boldsymbol{h}^1,\boldsymbol{h}^2,\boldsymbol{h}^3,\Psi)=-\boldsymbol{v}^{\mathrm{T}}\boldsymbol{W}^1\boldsymbol{h}^1-\boldsymbol{h}^{1\mathrm{T}}\boldsymbol{W}^2\boldsymbol{h}^2-\boldsymbol{h}^{2\mathrm{T}}\boldsymbol{W}^3\boldsymbol{h}^3-\boldsymbol{v}^{\mathrm{T}}\boldsymbol{B}-\boldsymbol{h}^1\boldsymbol{A}^{1\mathrm{T}}-\boldsymbol{h}^2\boldsymbol{A}^{2\mathrm{T}}-\boldsymbol{h}^{3\mathrm{T}}\boldsymbol{A}^3, \quad (40)$$

其中 $\Psi=\{\boldsymbol{W}^1,\boldsymbol{W}^2,\boldsymbol{W}^3,\boldsymbol{B},\boldsymbol{A}^1,\boldsymbol{A}^2,\boldsymbol{A}^3\}$ 是模型参数,$\boldsymbol{W}^1,\boldsymbol{W}^2,\boldsymbol{W}^3$ 分别表示可见层节点到隐层节点和隐层节点到隐层节点的对称连接,$\boldsymbol{B},\boldsymbol{A}^1,\boldsymbol{A}^2,\boldsymbol{A}^3$ 分别为

可见层节点和隐层节点的阈值. 可见单元向量 $\boldsymbol{v}$ 状态为 1 的概率分布为

$$p(\boldsymbol{v},\Psi)=\frac{1}{Z(\Psi)}\sum_{\boldsymbol{h}^1,\boldsymbol{h}^2,\boldsymbol{h}^3}\mathrm{e}^{-E(\boldsymbol{v},\boldsymbol{h}^1,\boldsymbol{h}^2,\boldsymbol{h}^3,\Psi)}. \quad (41)$$

隐单元 j,m,l 和可见单元 i 状态为 1 的条件概率分布为

$$p(h_j^1=1|\boldsymbol{v},\boldsymbol{h}^2)=\sigma\Big(\sum_{i=1}^{D}w_{ij}^1v_i+\sum_{m=1}^{K^2}w_{jm}^2h_m^2+a_j^1\Big), \quad (42)$$

$$p(h_m^2=1|\boldsymbol{h}^1,\boldsymbol{h}^3)=\sigma\Big(\sum_{j=1}^{K^1}w_{jm}^2h_j^1+\sum_{l=1}^{K^3}w_{ml}^3h_l^3+a_m^2\Big), \quad (43)$$

$$p(h_l^3=1|\boldsymbol{h}^2)=\sigma\Big(\sum_{m=1}^{K^2}w_{ml}^3h_m^2+a_l^3\Big), \quad (44)$$

$$p(v_i=1|\boldsymbol{h}^1)=\sigma\Big(\sum_{j=1}^{K^1}w_{ij}h_j^1+b_i\Big), \quad (45)$$

其中 $\sigma(x)=\frac{1}{1+\mathrm{e}^{-x}}$ 是逻辑斯蒂函数.

仍然可以用前面描述的 BM 的学习过程对 DBM 进行极大似然学习,将式(41)取对数,对 $\boldsymbol{W}^1$ 求偏微分,得到参数更新规则为 $\frac{\partial\ln p(\boldsymbol{v},\Psi)}{\partial\boldsymbol{W}^1}=\Delta W=\varepsilon(E_{p_{\text{data}}}[\boldsymbol{v}\boldsymbol{h}^{1T}]-E_{p_{\text{model}}}[\boldsymbol{v}\boldsymbol{h}^{1T}])$,然后用上面介绍的算法近似得到期望值. 但是由于 DBM 具有多个隐层,增加了其不确定性,使学习过程变得相当慢,尤其对远离可见单元的隐层来说. 因此,通常对模型进行预训练将模型参数初始化到合适的值来加速学习过程[18].

预训练学习算法一次学习堆栈 RBM 中的一层,学习完这些 RBM 之后,将堆栈 RBM 组合形成 DBM. 学习堆栈 RBM 最下面一层产生的模型为

$$p(\boldsymbol{v},\Psi)=\sum_{\boldsymbol{h}^1}p(\boldsymbol{h}^1,\boldsymbol{W}^1)p(\boldsymbol{v}|\boldsymbol{h}^1,\boldsymbol{W}^1), \quad (46)$$

其中 $p(\boldsymbol{h}^1,\boldsymbol{W}^1)=\sum_{\boldsymbol{v}}p(\boldsymbol{h}^1,\boldsymbol{v},\boldsymbol{W}^1)$ 是参数 $\boldsymbol{W}^1$ 定义的 $\boldsymbol{h}^1$ 上的隐式先验概率分布. 在堆栈 RBM 中的第 2 层,用参数 $\boldsymbol{W}^2$ 定义的更好的先验概率分布 $p(\boldsymbol{h}^1,\boldsymbol{W}^2)=\sum_{\boldsymbol{h}^2}p(\boldsymbol{h}^1,\boldsymbol{h}^2,\boldsymbol{W}^2)$ 替换 $p(\boldsymbol{h}^1,\boldsymbol{W}^1)$. 当第 2 层 RBM 被正确初始化时,$p(\boldsymbol{h}^1,\boldsymbol{W}^2)$将会成为 $\boldsymbol{h}^1$ 上更好的整体后验分布模型. 然后取 $\boldsymbol{h}^1$ 的两个模型的平均值,即在用 $1/2\boldsymbol{W}^1$ 自底向上近似的同时用 $1/2\boldsymbol{W}^2$ 自顶向下近似,推断出 $p(\boldsymbol{h}^1,\boldsymbol{W}^1,\boldsymbol{W}^2)$. 因为 $\boldsymbol{h}^2$ 是从 $\boldsymbol{v}$ 推断得到的,若将所有的自底向上近似和自顶向下近似相加,会产生对 $\boldsymbol{v}$ 计算 2 次的问题,得到急剧变化的概率分布模型.

Hinton 等人[19]提出的贪心逐层预训练学习算法同时训练多个 RBM,然后再将这些 RBM 组合形成 DBM,解决了将自底向上近似和自顶向下近似组合产生的 2 次计算问题. 如图 6 所示,对底层 RBM,

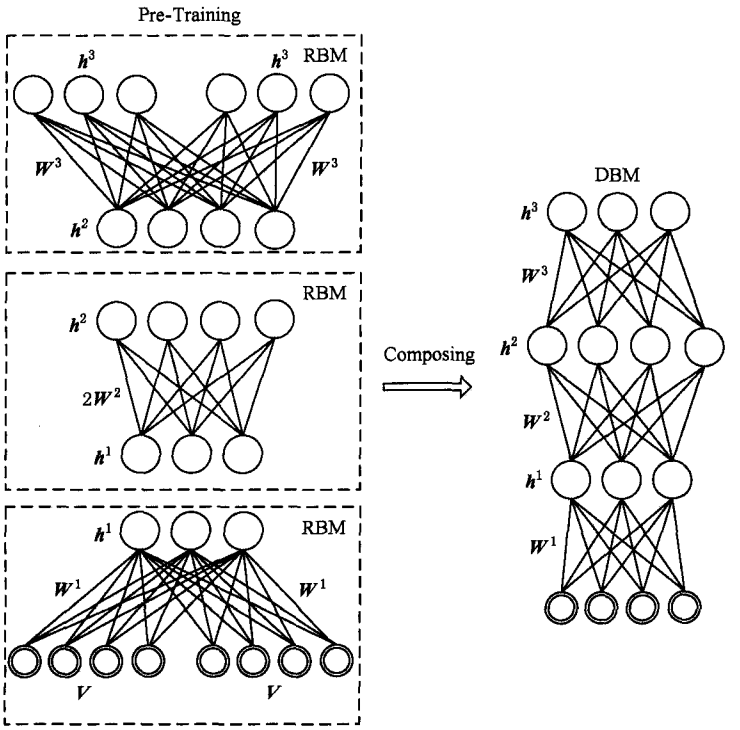

Fig. 6 diagram of greedy layer-wise pre-training method.

图 6 贪心逐层预训练法示意图

分别将可见单元 $\boldsymbol{v}$ 和可见单元 $\boldsymbol{v}$ 到隐单元 $\boldsymbol{h}^1$ 的权值 $\boldsymbol{W}^1$ 的数量加倍，定义修改后的 RBM 的第 1 层隐单元 j 和可见单元 i 状态为 1 的条件概率分布为：

$$p(h_j^1=1|\boldsymbol{v})=\sigma\Big(\sum_{i=1}^{D}w_{ij}^1v_i+\sum_{i=1}^{D}w_{ij}^1v_i\Big)\ ,\quad(47)$$

$$p(v_i=1|\boldsymbol{h}^1)=\sigma\Big(\sum_{j=1}^{K^1}w_{ij}h_j^1\Big).\quad(48)$$

分别将隐单元 $\boldsymbol{h}^3$ 和隐单元 $\boldsymbol{h}^2$ 到隐单元 $\boldsymbol{h}^3$ 的权值 $\boldsymbol{W}^3$ 的数量加倍，得到第 2 层隐单元 $\boldsymbol{m}$ 和第 3 层隐单元 l 状态为 1 的条件概率分布为：

$$p(h_m^2=1|\boldsymbol{h}^3)=\sigma\Big(\sum_{l=1}^{K^3}w_{ml}^3h_l^3+\sum_{l=1}^{K^3}w_{ml}^3h_l^3\Big),\quad(49)$$

$$p(h_l^3=1|\boldsymbol{h}^2)=\sigma\Big(\sum_{m=1}^{K^2}w_{ml}^3h_m^2\Big).\quad(50)$$

对中间层 RBM，将隐单元 $\boldsymbol{h}^1$ 到隐单元 $\boldsymbol{h}^2$ 的权值 $\boldsymbol{W}^2$ 加倍，得到第 1 层隐单元 j 和第 2 层隐单元 m 状态为 1 的条件概率分布为：

$$p(h_j^1=1|\boldsymbol{h}^2)=\sigma\Big(\sum_{m=1}^{K^2}2w_{jm}^2h_m^2\Big),\quad(51)$$

$$p(h_m^2=1|\boldsymbol{h}^1)=\sigma\Big(\sum_{j=1}^{K^1}2w_{jm}^2h_j^1\Big).\quad(52)$$

将这 3 个 RBM 模块组成一个单独的系统，将进入第 1 个隐层和第 2 个隐层的总输入减半，得到下面的第 1 层隐单元 j 和第 2 层隐单元 m 状态为 1 的条件概率分布为：

$$p(h_j^1=1|\boldsymbol{v},\boldsymbol{h}^2)=\sigma\Big(\sum_{i=1}^{D}w_{ij}^1v_i+\sum_{m=1}^{K^2}w_{jm}^2h_m^2\Big),\quad(53)$$

$$p(h_m^2=1|\boldsymbol{h}^1,\boldsymbol{h}^3)=\sigma\Big(\sum_{j=1}^{K^1}w_{jm}^2h_j^1+\sum_{l=1}^{K^3}w_{ml}^3h_l^3\Big).\quad(54)$$

可见单元 i 和第 3 层隐单元 l 状态为 1 的条件概率分布保持不变，如式(48)和式(50).

注意到组合模型定义的条件概率分布与 DBM 定义的条件概率分布完全一样. 因此，贪心预训练 3 个改进的 RBM 得到有对称权值的无向图模型——DBM. 当用贪心逐层预训练法训练更多的 RBM 时，只用修改堆栈 RBM 中的第 1 个和最后 1 个，对所有中间的 RBM，将它们组合形成 DBM 时只需将其 2 个方向的权值减半.

用这种方法贪心预训练 DBM 的权值在将权值初始化为合适值的同时也保证了存在自底向上通过堆栈 RBM 来进行近似推断的快速方法. 给定可见单元上的一个训练样本，在自底向上近似的过程中激活每个隐层的单元，通过加倍自底向上的输入来弥补自顶向下的反馈缺失.

DBM 作为多层网络，中间包含多个隐层，存在大量的变量和参数的不确定性，在计算上不可行. 因此，在实际应用中，一般采用 3 层 BM，即输入层和输出层之间只有 1 个隐层.

5 玻尔兹曼机研究的新进展

由于 BM 能够用于很好地解决一些复杂问题，近几年许多研究人员对其进行了深入研究，使得 BM 研究领域出现了许多新进展，下面分别从学习算法、模型结构和实际应用这 3 个方面对近几年 BM 的研究新进展进行介绍.

5.1 学习算法

Montufar 等人[20]从理论上说明{0,1}上的任意概率分布与 RBM 模型的概率分布之间的极大 K-L 离差的上界是$(D-1)-\ln(K+1)$.

许多研究人员对 BM 的学习算法进行改进，并不断提出一些新的算法用于有效训练 BM.

Sutskever 等人[21]提出循环回火 RBM，对回火 RBM 稍加改进，使其能够很容易进行准确推断和精确梯度学习，说明对动作捕捉数据和弹力球视频像素，将循环回火 RBM 产生的样本和回火 RBM 产生的样本用于实验，前者能够得到更好的学习性能，其主要原因是循环回火 RBM 通过隐单元之间的连接权能够传递更多的信息.

将堆栈 RBM 组合得到 DBN，Mohamed 等人[22-23]用梅尔频率倒谱系数和梅尔刻度滤波器训练 DBN 产生高层 RBM 特征，用这个特征能够预测 HMM 状态上的后验分布，用反向传播微调后，在 TIMIT 数据集上识别与说话者无关的音素得到的性能优于其他方法. 相似地，Pham 等人[24]也用频谱图训练 DBN，并将 DBN 应用到一些音频分类任务中.

Salakhutdinov 和 Hinton[18]提出一种用于训练 DBM 的学习算法，用变分近似法估计依赖数据的期望值，用持续马尔可夫链估计模型的期望值，说明用逐层预训练法可以更有效地进行学习以及如何用退火重要性抽样估计 DBM 的对数似然函数的下界，在 MNIST 手写数字数据集和视觉目标识别任务 NORB 数据集上进行实验说明 DBM 学习得到性能很好的产生式模型.

Salakhutdinov 和 Hinton[25]提出用于训练一般 BM 的相当有效的学习方法，通过学习仿真实后验

分布和平均场变分推理假定因子分布接近,来估计依赖数据的期望值,通过学习和马尔可夫链之间的相互作用,允许少量缓慢混合链从多模态能量图中快速采样,来估计独立于数据的期望值,并说明可以通过训练堆栈 RBM 对 DBM 的权值进行初始化.

Salakhutdinov[26]基于自适应 MCMC 算法,提出耦合自适应模拟回火算法(coupled adaptive simulated tempering, CAST)用于训练 DBM 来得到更好的多模态能量图,并说明 FPCD 算法和自适应 MCMC 算法在概念上的关联关系,在 MNIST 和 NORB 数据集上进行实验证明 CAST 能够有效改进参数估计.

Salakhutdinov[27]提出基于回火变换 MCMC 因子的 Trans-SAP 算法,与模拟回火不同,这种算法用不同温度的多个链采样,系统地将样本在原始目标分布与高温分布之间移动,基于 Metropolis-Hasting 规则接受新的状态.为了应用到 DBM 等复杂的模型,通常引入许多中间分布来保持合理的接受概率并允许样本移出局部模式.

CD 算法是一种学习 RBM 的有效方法,但是由于学习梯度中包含有偏差近似而存在缺点.Cho 等人[28]提出用一种先进的蒙特卡罗方法——PT 算法,代替 CD 算法训练 RBM,改进了采样质量,减小了计算损失,并通过实验说明其有效性.

Desjardins 等人[29]提出用 PT 算法学习 RBM,给出了 PT 算法表现优于 CD 算法的原因,并用实验结果说明用 PT 算法学习的优越性.但是存在的问题是运行多条链的速度比朴素随机近似慢许多倍,并且存储量大,在每个时间步骤都需要存储每个耦合链的状态.

Desjardins 等人[12]提出一种交替策略——回火 MCMC 采样技术,用于改善 PCD 算法负阶段马尔可夫链的混合时间,用一系列 PT 链替换 PCD 算法中的单个马尔可夫链,通过在实际数据集和人工生成数据集上定性定量观察,得到这种方法能产生好的产生式模型,说明回火技术的运用为学习率和无监督训练阶段提供了高可靠性和强鲁棒性.

Cho 等人[30]提出改进自适应学习率和增强梯度估计,用于提高 PT 算法和 CD 算法的性能,在 MNIST 手写数字数据集和加州理工学院 101 轮廓数据集上进行实验,结果表明新的训练算法能够避免用传统梯度下降法训练 RBM 的许多困难,使 RBM 的训练更稳定.

5.2 模型结构

除了改进学习算法,研究人员也考虑提出新的 BM 模型拓扑结构来提高其学习性能.

Nair 和 Hinton[31]用无限数量的阶梯 S 型单元近似替换每个二值单元产生 RBM,所有阶梯 S 型单元的权值相同,负偏差渐进减小,并且学习和推断规则不变,也可以用有噪声的、可调整的线性单元有效近似,在人脸识别和 NORB 数据集的目标识别任务上进行实验,说明这些单元比二值单元能更好地学习特征,能够比二值单元更加自然地学习光强大范围变化情况.

Ranzato 等人[32]提出一种新的模型——因式分解的三值 RBM,修改有实值可见单元和二值隐单元的 RBM 使其包含三元交互作用,将实值图像映射到因式分解的输出,用隐单元的状态表示图像的局部协方差结构中的异常,允许隐单元控制可见单元的协方差及阈值,在 CIFAR-10 数据集上进行实验得到非常高的准确性,对小图像数据集上的目标识别非常有效.

Courville 等人[33]提出一种长且宽的 RBM,由与隐层每个单元关联的二值长变量和实值宽变量构成,宽变量允许模型得到协方差信息,同时通过吉布斯采样得到简单有效的推断,说明长且宽的 RBM 如何在 CIFAR-10 目标识别任务中表现出优越的学习性能.

Eslami 等人[34]将形状 BM 用于二值形状图像的建模任务中,说明形状 BM 能够很好地得到形状模型的特征,将从模型采样得到的样本用于实际应用,并且将模型泛化产生不同于训练样本的样本,说明形状 BM 如何在目标形状上学习高质量的概率分布,形状 BM 学习的分布在质量和数量上都表现出优于这个任务的现有模型的性能.

Cai 等人[35]用 RBM 从维数高达 324 的原始数据中提取可辨识的低维特征,然后用提取的特征作为支持向量机的输入用于回归,实验结果表明这种方法用于股票价格预测能够得到很大的改进,与用原始数据的支持向量机相比有更低的预测误差.

5.3 实际应用

BM 及其模型已经成功应用于协同滤波、分类、降维、图像检索、信息检索、语言处理、自动语音识别、时间序列建模、文档分类、非线性嵌入学习、暂态数据模型学习和信号与信息处理等任务中.

学习自然图像的产生式模型是提取特征的有效

方法,之前学习这种模型的方法是用隐特征分别确定每个像素的均值和方差,或者用隐单元确定0均值高斯分布的协方差阵. Ranzato 和 Hinton[36]提出一种概率模型,将上面2种方法组合成一个单一结构,用一组二值隐特征表示每个图像,用隐特征建立特定图像协方差的模型,用一个单独的集合建立均值的模型,并说明这种方法提供一种广泛适用于简单单元和复杂单元结构的概率框架,在难以处理的CIFAR10数据集上进行实验得到优越的识别准确性.

Yu 和 Deng[37]提出一组新的批处理模式算法,该算法是他们提出的基于语音识别的可伸缩深神经网络的一个关键部分,该算法的实质是构造单隐层神经网络,上面层的权值为下面层权值的确定性函数,使权值沿着误差减小最多的方向移动,并在MNIST手写数字识别及TIMIT框架等级音素和音素状态分类任务上进行实验,验证该算法能够用于解决训练深神经网络的可伸缩结构问题.

DBN被用于音素识别,并且发现能够得到很好的学习性能. Mohamed 等人[38]提出用连续判别训练准则来优化DBN权值、状态变换参数及语言模型分数,描述并分析了提出的训练算法和策略,并讨论该算法如何影响学习结果,说明基于序列训练准则学习的DBN的性能优于基于框架准则的DBN,但是与此同时产生的优化过程更加困难.

最近独立于语境的DBN隐马尔可夫模型(hidden Markov model, HMM)混合结构在音素识别上得到好的结果. Dahl 等人[39]提出一种依赖语境的DBN-HMM系统,在来自Bing移动语音搜索任务的具有大型词汇的任意语音识别数据集上进行实验,得到的结果性能明显优于高斯混合-隐马尔可夫模型(Gaussian mixture model-hidden Markov model, GMM-HMM),并且用最小音素误差率和极大似然准则可以明显改进精度.

近期深度学习结构被成功应用于自动语音识别系统,先是出现了独立于语境的DBN-HMM进行语音识别,后来又提出了依赖于语境的DBN-HMM进行有大规模词汇的连续语音识别. Yu 等人[40]设计DBN-HMM识别性能实验,用来说明预训练和微调这2个阶段的作用,说明预训练可以将权值初始化到空间中的一个点,在该点微调能够进行有效调节,其中该实验基于大规模词汇语音识别器,并且依赖语境. 当原始数据集的大小足够初始化DBN权值时,适当增加未标记预训练数据的数量对最终的识别结果有重要的影响,另外用额外的有标记训练数据,DBN训练的微调阶段可以有效提高识别准确性.

Deng 等人[41]报道了对训练语音谱图的多层产生式模型的逐层学习策略的研究,产生式模型的顶层进行二值编码学习,可以用于有效的语音压缩、可扩展的语音识别及快速语音语境检索,产生式模型的每一层与下一层完全连接,用CD算法可以有效预训练这些连接权值. 在逐层预训练之后,展开产生式模型形成深自动编码器,然后用反向传播微调其参数. 用相应的二值编码预测每个谱图片段然后叠加组合,重建完整长度的语音谱图.

Stafylakis 等人[42]用BM提出一种新的用于说话者识别的非高斯概率结构,说明提取的语言表达向量表示如何将一些BM结构用于说话者识别任务中.

下面对BM在自动语音识别、目标识别和文档分类等重要应用中的情况进行说明.

5.3.1 自动语音识别

目前的语音识别系统依赖梅尔倒谱系数和线性预测编码系数等,将高维语音声波转化为低维编码的预处理语音特征,但是低维编码可能会丢失一些相关信息,并用难以判别的方式表达一些其他信息. Jaitly 和 Hinton[43]用RBM建立语音信号模型,这个模型可以用CD算法进行有效训练,学习与识别任务关联性更高的特征来更好地得到信号的期望值,在TIMIT数据集上进行音素识别,为了防止隐单元饱和而产生小的学习信号,将TIMIT训练数据集归一化到标准偏差为10,使用包含100个高斯可见单元和120个阶梯S型隐单元的RBM,用100个随机选择的语音微批次执行随机梯度下降,对TIMIT数据集每个音素生成61个有3个状态的单声道音素HMMs. 实验结果表明该方法的性能优于当前基于梅尔滤波器组或梅尔频率倒谱系数的方法性能.

5.3.2 目标识别

Salakhutdinov 和 Larochelle[44]提出一种用于DBM的近似推断算法,学习一个单独的识别模型,用该模型快速初始化所有隐单元的值,说明采用该识别模型结合自顶向下和自底向上,可以有效学习一个好的有高维结构感知输入的产生式模型,并说明算法由于包含自顶向下反馈所产生的额外计算对DBM的性能产生重要的影响. 用这种算法在MNIST

手写数字识别、OCR 英文字母识别、NORB 视觉目标识别任务中进行实验. MNIST 数字数据集包含 5 万个训练图像、1 万个校验图像和 1 万个测试图像,每个图像中有像素为 28×28 的 0～9 这 10 个手写数字;OCR 字母数据集包含 32 152 个训练样本、1 万个校验样本和 1 万个测试样本,将 16×8 个二值像素图像划分到 26 个英文字母类中;最难处理的 NORB 目标识别数据集包含 24 300 个训练立体图像对、4 300 个校验立体图像对和 24 300 个测试立体图像对,将 50 个不同的 3D 玩具目标分为 5 类. 对于相对简单的 MNIST 数字识别和 OCR 字母识别问题,用有 2 个隐层的 DBM,用于 MNIST 数据集的 DBM 的第 1 隐层和第 2 隐层分别有 500 和 1 000 个隐单元,用于 OCR 数据集的 DBM 的 2 个隐层均有 2 000 个隐单元;而对于较难处理的 NORB 目标识别问题,使用有 3 个隐层的 DBM,每个隐层包含 4 000 个隐单元. 为了加速学习过程,将数据划分为微批次,采用 5 步吉布斯更新随机近似法. 实验结果显示训练 DBM 能够得到优越的性能.

5.3.3 文档分类

在文档分类问题中,直接将不规范的文档内容作为输入,会产生过高的输入数据维数而无法对其进行处理,因此有必要对文档进行预处理,选择词组出现的频率作为特征项,提取能够表示其本质特征的数据. 杨莹等人[45]采用 RBM 从原始的高维输入特征中提取可高度区分的低维特征,然后将其作为支持向量机的输入,进行回归分析,从而实现对文档进行分类. 目前国内还没有标准的中文文档分类测试数据库,杨莹等人用来自腾讯网的文档建立数据库对其构造的文档分类器进行测试,这些文档分为 40 类,取其中包含文档数最多的 20 个类进行测试,训练集包括 10 033 个文档,测试集包括 8 032 个文档. 实验结果显示,采用 RBM 提取可高度区分的低维特征对提高支持向量机的回归性能起到改进作用,从而可有效提高文档分类的准确性.

6 总结与展望

BM 作为深神经网络的一个重要代表受到了广泛的关注. BM 是对称耦合的随机二值单元网络,通过学习建立单元之间的高阶相关模型,用基于模型的能量函数中的隐单元和可见单元来得到具有更高表示能力的模型,能够对复杂层次结构数据进行建模. BM 的原理比较完备,在 MNIST 和 NORB 等数据集上显示出优越的学习性能. 但是,BM 的推理学习过程算法复杂性过高,无法有效地应用于大规模学习问题,因而研究人员提出对网络拓扑结构简化,改进学习算法,对非线性寻优过程合理近似,减少学习时间,许多 BM 理论和方法得以发展.

本文详细概述了 BM 的基本概念、单层反馈网络的模型及拓扑结构分类,对 BM 和 DBM 的学习过程和典型学习算法进行了探讨,从学习算法、模型结构和实际应用 3 方面介绍了近几年 BM 研究的相关进展. 随着 BM 理论与方法研究的深入,BM 将更加广泛地应用于各个领域. 未来 BM 的研究需要解决以下问题:

1) BM 包含大量神经元,并且其学习过程中包含许多参数和计算项,因此需要耗费大量的学习时间,如何有效减少计算复杂性是一个重要的问题. 除了上面介绍的一些随机全局算法之外,现在仍存在许多全局最优搜索算法,如交叉熵方法、模型参考自适应算法等,能用于有效学习 BM,因此需要把现有的其他有效全局最优化算法应用于 BM 学习过程中;除了用预训练结合微调之外,还有很大的空间来改进当前的 DBM 优化技术,这是值得继续研究的方向. 除了上面介绍的用 K-L 离差作为距离度量的算法之外,可以考虑用 Kolmogorov 距离、Bhattacharyya 距离和 Patrick-Fisher 距离等概率距离度量[46]替换 K-L 离差,得到各种新的算法,并理论分析和实验比较与 K-L 离差算法之间的异同优劣.

2) 除了使用有效的最优搜索算法外,运用简化的网络拓扑结构也能够减少计算复杂性,这是简化模型结构问题. 除了上面介绍的一些网络拓扑结构之外,未来需要进一步开发新的能有效学习的网络拓扑结构,对现有模型的拓扑结构进行简化,并研究拓扑结构简化后对最终学习性能的影响,给出理论上的权衡准则、拓扑结构的简化与预测性能之间有效的平衡,给出预测误差界担保的简化拓扑结构设计.

3) 目前存在的 DBN-HMM 等方法在利用 DBN 的能力方面只是最简单的模型,还没有充分发掘出 DBN 的优势. 因此,需要研究 DBN 结构的特点和规律,找到更好的方法用深结构建立数据的模型,充分利用 DBN 内在的优势. 这是模型选择问题,可以研究将现有的社会网络、基因调控网络、结构化建模理论、稀疏化建模理论以及压缩传感等理论运用于 BM 中.

4) BM 在协同滤波、图像检索、语言处理等问题中表现出色,性能大大超过了简单神经网络,但是还需要进一步开发其在解决实际问题中的应用,拓展其应用场景.

5) BM 算法的可调参数,如学习率、温度值和样本的数量,对学习性能的影响很大.未来还需要更深入地探讨参数的改变如何对学习性能产生影响,并且研究如何选择合适的参数来保证学习性能的提高,需要特定有效的参数调整规则,这方面信息论、编码理论、最小描述长度理论和贝叶斯理论等理论的引入可以提供相应的指导意见.除了学习模型的参数之外,还要提出有效学习模型结构的理论和方法.

6) 目前的神经元层模型对于从特征中提取信息来说不够强大,需要提出理论来指导在每层搜索合适的特征提取模型,确定在何种条件下,能将更好的产生式模型用于训练来提高学习性能,如何提高模型泛化能力,这些都是值得继续研究的问题,需要新的理论的引入.在提高模型泛化能力方面,可以考虑引入经验过程理论和统计学习理论.

7) 在实际应用中,当测试数据集的分布与训练数据集的分布不同时,深神经网络模型很难得到好的学习效果.因此,有必要提出深神经网络模型的自适应技术,可以考虑将交叉熵、自适应抽样等技术应用到 BM 中.

8) 目前基于微批次处理随机梯度的优化算法很难在计算机上并行处理,最好的解决方法是用图处理单元(graphical processing unit, GPU)加速学习过程,但是单个机器 GPU 无法用于大规模数据集.因此需要提出有效的可扩展的并行学习算法来训练深神经网络模型,可以考虑引入并行吉布斯采样法、网络结构因式分解法、并行马尔可夫随机场法和并行坐标下降法等算法.

参 考 文 献

[1] Hopfield J J. Neural networks and physical systems with emergent collective computational abilities [J]. Proceedings of the National Academy of Sciences of the United States of America, 1982, 79(8): 2554-2558

[2] Hinton G E, Sejnowski T J. Optimal perceptual inference [C] //Proc of the 1983 IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 1983: 448-453

[3] Hinton G E, Sejnowski T J. Analyzing cooperative computation [C] //Proc of the 5th Annual Congress of the Cognitive Science Society. New York: ACM, 1983: 2554-2558

[4] Hofstadter D R. The copycat project: An experiment in nondeterminism and creative analogies [DB/OL]. MIT Artificial Intelligence Laboratory Memo 755. (1984-01-01) [2004-10-01]. http://hdl.handle.net/1721.1/5648

[5] Hofstadter D R. A Non-Deterministic Approach to Analogy, Involving the Ising Model of Ferromagnetism [M] //The Physics of Cognitive Processes. Hackensack: World Scientific, 1987

[6] Smolensky P. Information Processing in Dynamical Systems: Foundations of Harmony Theory [M] //Parallel Distributed Processing, Vol 1: Foundations. Cambridge: MIT Press, 1986: 194-281

[7] Ackley D H, Hinton G E, Sejnowski T J. A learning algorithm for Boltzmann machines [J]. Cognitive Science, 1985, 9(1): 147-169

[8] Hinton G E. Training products of experts by minimizing contrastive divergence [J]. Neural Computation, 2002, 14(8): 1771-1800

[9] Kirkpatrick S, Gelatt C D, Vecchi M P. Optimization by simulated annealing [J]. Science, 1983, 220(4598): 671-680

[10] Hinton G E. To recognize shapes, first learn to generate images [J]. Computational Neuroscience: Theoretical Insights into Brain Function, 2007, 165(1): 535-547

[11] Hinton G E. A practical guide to training restricted Boltzmann machines [G] //LNCS 7700: Neural Networks: Tricks of the Trade. Berlin: Springer, 2012: 599-619

[12] Desjardins G, Courville A, Bengio Y, et al. Parallel tempering for training of restricted Boltzmann machines [C] //Proc of the 13th Int Conf on Artificial Intelligence and Statistics. New York: ACM, 2010: 145-152

[13] Jordan M I, Ghahramani Z, Jaakkola T S, et al. An introduction to variational methods for graphical models [J]. Machine Learning, 1999, 37(2): 183-233

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

[15] Tieleman T. Training restricted Boltzmann machines using approximations to the likelihood gradient [C] //Proc of the 25th Int Conf on Machine Learning. New York: ACM, 2008: 1064-1071

[16] Hinton G E. Training products of experts by minimizing contrastive divergence [J]. Neural Computation, 2002, 14(8): 1711-1800

[17] Tieleman T, Hinton G E. Using fast weights to improve persistent contrastive divergence [C] //Proc of the 26th Annual Int Conf on Machine Learning. New York: ACM, 2009: 1033-1040

[18] Salakhutdinov R, Hinton G E. Deep Boltzmann machines [J]. Journal of Machine Learning Research-Proceedings Track, 2009, 9(1): 448-455

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

[20] Montufar G F, Rauh J, Ay N. Expressive power and approximation errors of restricted Boltzmann machines [C] // Proc of the 2011 Neural Information Processing Systems. New York: ACM, 2011: 415-423

[21] Sutskever I, Hinton G E, Taylor G. The recurrent temporal restricted Boltzmann machine [C] //Proc of the 2008 Neural Information Processing Systems. New York: ACM, 2008: 1601-1608

[22] Mohamed A, Sainath T N, Dahl G E. Deep belief networks using discriminative features for phone recognition [C] //Proc of the 2011 IEEE Int Conf on Acoustics, Speech and Signal Processing. Piscataway, NJ: IEEE, 2011: 5060-5063

[23] Dahl G, Ranzato M A, Mohamed A, et al. Phone recognition with the mean-covariance restricted Boltzmann machine [C] //Proc of the 2010 Neural Information Processing Systems. New York: ACM, 2010: 469-477

[24] Pham P, Lee H, Largman Y, et al. Unsupervised feature learning for audio classification using convolutional deep belief networks [C] //Proc of the 2009 Neural Information Processing Systems. New York: ACM, 2009: 1096-1104

[25] Salakhutdinov R, Hinton G E. An efficient learning procedure for deep Boltzmann machines [J]. Neural Computation, 2012, 24(8): 1967-2006

[26] Salakhutdinov R. Learning deep Boltzmann machines using adaptive MCMC [C] //Proc of the 27th Int Conf on Machine Learning. New York: ACM, 2010: 943-950

[27] Salakhutdinov R. Learning in Markov random fields using tempered transitions [C] //Proc of the 2009 Neural Information Processing Systems. New York: ACM, 2009: 1598-1606

[28] Cho K, Raiko T, Ilin A. Parallel tempering is efficient for learning restricted Boltzmann machines [C] //Proc of the 2010 Int Joint Conf on Neural Networks. New York: ACM, 2010: 1-8

[29] Desjardins G, Courville A, Bengio Y, et al. Tempered Markov chain Monte Carlo for training of restricted Boltzmann machines [J]. Journal of Machine Learning Research-Proceedings Track, 2010, 9(1): 145-152

[30] Cho K, Raiko T, Ilin A. Enhanced gradient and adaptive learning rate for training restricted Boltzmann machines [C] //Proc of the 28th Int Conf on Machine Learning. New York: ACM, 2011: 105-112

[31] Nair V, Hinton G E. Restified linear units improve restricted Boltzmann machines [C] //Proc of the 27th Int Conf on Machine Learning. New York: ACM, 2010: 807-814

[32] Ranzato M A, Krizhevsky A, Hinton G E. Factored 3-way restricted Boltzmann machines for modeling natural images [J]. Journal of Machine Learning Research-Proceedings Track, 2010, 9(1): 621-628

[33] Courville A, Bergstra J, Bengio Y. A spike and slab restricted Boltzmann machine [J]. Journal of Machine Learning Research : Proceedings Track, 2011, 15(1): 233-241

[34] Eslami S M A, Heess N, Winn J. The shape Boltzmann machine: a strong model of object shape [C] //Proc of the 2012 IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 2012: 406-413

[35] Cai X, Hu S, Lin X. Feature extraction using restricted Boltzmann machine for stock price prediction [C] //Proc of the 2012 IEEE Int Conf on Computer Science and Automation Engineering. Los Alamitos, CA: IEEE Computer Society, 2012, 3: 80-83

[36] Ranzato M A, Hinton G E. Modeling pixel means and convariances using factorized third-order Boltzmann machines [C] //Proc of the 2010 IEEE Conf on Computer Vision and Pattern Recognition. Los Alamitos, CA: IEEE Computer Society, 2010: 2551-2558

[37] Yu D, Deng L. Accelerated parallelizable neural network learning algorithm for speech recognition [C] //Proc of Interspeech 2011. Piscataway, NJ: IEEE, 2011: 2281-2284

[38] Mohamed A, Yu D, Deng L. Investigation of full-sequence training of deep belief networks for speech recognition [C] // Proc of Interspeech 2010. Piscataway, NJ: IEEE, 2010: 2846-2849

[39] Dahl G E, Yu D, Deng L, et al. Large vocabulary continuous speech recognition with context-dependent DBN-HMMS [C] //Proc of the 2011 IEEE Int Conf on Acoustics, Speech and Signal Processing. Piscataway, NJ: IEEE, 2011: 4688-4691

[40] Yu D, Deng L, Dahl G E. Roles of pre-training and fine-tuning in context-dependent DBN-HMMs for real-world speech recognition [C] //Proc of NIPS 2010 Workshop on Deep Learning and Unsupervised Feature Learning. New York: Academy Press, 2010

[41] Deng L, Seltzer M, Yu D, et al. Binary coding of speech spectrograms using a deep auto-encoder [C] //Proc of Interspeech 2010. Piscataway, NJ: IEEE, 2010: 1692-1695

[42] Stafylakis T, Kenny P, Senoussaoui M, et al. Preliminary investigation of Boltzmann machine classifiers for speaker recognition [C] //Proc of Odyssey 2012: The Speaker and Language Recognition Workshop. 2012: 109-116

[43] Jaitly N, Hinton G E. Learning a better representation of speech sound waves using restricted Boltzmann machines [C] //Proc of the 2011 IEEE Int Conf on Acoustics, Speech and Signal Processing. Piscataway, NJ: IEEE, 2011: 5884-5887

[44] Salakhutdinov R, Larochelle H. Efficient learning of deep Boltzmann machines [J]. Journal of Machine Learning Research : Proceedings Track, 2010, 9(1): 693-700

[45] Yang Ying, Wu Chengwei, Hu Su. Chinese documents classification based on restricted Boltzmann machines [J]. Science and Technology Innovation Herald, 2012, 26(16): 35-36 (in Chinese)
(杨莹, 吴诚炜, 胡苏. 基于受限玻尔兹曼机的中文文档分类[J]. 科技创新导报, 2012, 26(16): 35-36)

[46] Zhou S K, Chellappa R. From sample similarity to ensemble similarity: Probabilistic distance measures in reproducing kernel hilbert space [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2006, 28(6): 917-929

Liu Jianwei, born in 1966. PhD, associate professor in the Department of Automation, College of Geophysics and Information Engineering, China University of Petroleum, Beijing Campus (CUP). His main research interests include intelligent information processing, machine learning, analysis, prediction, controlling of complicated nonlinear system, and analysis of the algorithm and the designing, corresponding author.

Liu Yuan, born in 1989. MSc candidate in the Department of Automation, College of Geophysics and Information Engineering, China University of Petroleum, Beijing Campus (CUP). Her main research interests include machine learning and digital image processing.

Luo Xionglin, born in 1963. PhD, professor in the Department of Automation, College of Geophysics and Information Engineering, China University of Petroleum, Beijing Campus (CUP). His main research interests include intelligent control, and analysis, prediction, controlling of complicated nonlinear system.

2014 年《计算机研究与发展》专题(正刊)征文通知

——“深度学习”

2006 年以来,深度学习开始受到学术界广泛关注,到今天已经成为互联网、大数据和人工智能的一个热潮.深度学习通过建立类似于人脑的分层模型结构,对输入数据逐级提取从底层到高层的特征,从而能很好地建立从底层信号到高层语义的映射关系.近年来,谷歌、微软、IBM、百度等拥有大数据的高科技公司相继投入大量资源进行深度学习技术研发,在语音、图像、自然语言、在线广告等领域取得显著进展.

为推动我国深度学习的科学研究和工程应用,及时报道深度学习领域国内外科技工作者所取得的研究成果,同时展望深度学习的研究方向,《计算机研究与发展》将于 2014 年 9 月或 10 月出版深度学习专辑,欢迎相关领域的专家学者和科研人员踊跃投稿.现将专题论文征集的有关事项通知如下.

征文范围(但不限于)

本专辑的征文范围包括(但不限于)下列主题:

- 深度学习的模型与算法
- 自动编码器
- 限制波尔兹曼机
- 深信度网络
- 卷积神经网络
- 稀疏编码
- 深度学习理论
- 大数据分析

征文要求

1) 论文应属于作者的科研成果,数据真实可靠,具有重要的学术价值与推广应用价值,未在国内外公开发行的刊物或会议上发表或宣读过,不存在一稿多投问题.作者在投稿时,需向编辑部提交投稿声明.
2) 论文应包括题目、作者信息、摘要、关键词、正文和参考文献,论文一律用 word 排版,论文格式体例格式请参考《计算机研究与发展》近期文章.
3) 论文需附通讯作者的联系地址、电话或手机及 E-mail 地址.
4) 论文请通过期刊网站(http://crad.ict.ac.cn)进行投稿,并注明“深度学习 2014 专题”(否则按自由来稿处理).

重要日期

发布征文通知:2013 年 10 月 30 日　　征文截止日期: 2014 年 3 月 1 日
录用通知日期: 2014 年 4 月 1 日　　作者修改稿提交日期:2014 年 5 月 1 日
出版日期:2014 年 9 月或 10 月

特邀编委

史忠植　研究员　中国科学院计算技术研究所　shizz@ics.ict.ac.cn
张长水　教　授　清华大学　zcs@mail.tsinghua.edu.cn
邓　立　Principal Researcher, Microsoft Research, Redmond, Washington, USA　deng@microsoft.com
陈松灿　教　授　南京航空航天大学　s.chen@nuaa.edu.cn
张　军　教　授　中山大学　junzhang@ieee.org
彭宇新　教　授　北京大学　pengyuxin@pku.edu.cn

联系方式

联系人:史忠植 shizz@ics.ict.ac.cn
编辑部: crad@ict.ac.cn 010-62620696,010-62600350
通信地址: 北京 2704 信箱《计算机研究与发展》编辑部　邮政编码: 100190