文章编号:1000-5641(2014)01-0060-08

基于深层神经网络(DNN)的汉语方言种属语音识别

景亚鹏, 郑 骏, 胡文心
(华东师范大学 计算中心,上海 200062)

摘要: 将深层神经网络(Deep Neural Network)应用于汉语方言种属语音识别. 基于优化的QuickNet软件,为方言识别实现了一种有监督的DNN逐层预训练方法. 在训练时,从3层开始逐层做有监督的神经网络训练,每增长一层的初始权值包含前一层训练好的部分权值和输出端的随机权值. 在得到最大层的初始权值后,再进行传统的BP网络训练. 该方法和普通神经网络相比识别率有较大提升,可用于移动互联网标准语音识别入口、方言口音鉴识等领域.

关键词: 深层神经网络; 方言语音识别; QuickNet
中图分类号: TP391 **文献标识码**: A **DOI**:10.3969/j.issn.1000-5641.2014.01.008

Belongingness of Chinese dialect speech recognition based on deep neural network

JING Ya-peng, ZHENG Jun, HU Wen-xin
(*Computing Center, East China Normal University, Shanghai* 200062, *China*)

Abstract: Based on the modified QuickNet software, we proposed a supervised DNN layerwise pre-training method for dialect speech recognition. The pre-training will start from a 3-layer neural network till the maximum layer, during which we will do supervised training. The initial weights of a new layer are composed of the partial trained weights of lower level network and the randomized weights closed to the output layer. Then we will do traditional back-propagation training when the initial weights of the maximum layer network are obtained. This method achieved a relatively higher recognition rate compared with normal neural network training and can be used in mobile speech recognition apps, the recognition of dialects speech and so on.

Key words: deep neural network; dialects speech recognition; QuickNet

0 引 言

语音识别隶属于模式识别,是机器学习领域近年来发展较为迅速的方向之一,特别是2012年Geoffrey E. Hinton等人将新的训练方法应用于识别之后[1]. 由机器自动识别人的语言并能够理解,是语音识别期望达到的最终目标. 但由于理论和技术条件的限制,迄今尚

收稿日期:2013-03
第一作者:景亚鹏,男,硕士研究生,研究方向为机器学习、Web应用技术. E-mail:bronzesword@sina.com.

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

未完全实现.

20 世纪 70 年代，隐马尔科夫模型(HMM)被用于语音识别[2]，取得了重大成功，成为该领域的主流方法. 人工神经网络(Artificial Neural Network)也是一种模式识别的有效模型. 但由于其算法(Back-propagation Algorithm)运算量太大，难以并行计算，深层网络无有效训练方法等原因导致其在识别领域成功应用不多. 直到 2006 年以后，计算机硬件已经有了大幅提高，GPU 用于浮点运算使神经网络的运算速度很快提高了几十倍[3]. 理论方面，借鉴生物学上的新发展，Hinton 等人为深层神经网络(DNN)的训练提出了逐层 pre-training 的新方法，使得 DNN 的识别率有了前所未有的提高. 克服了运算速度和算法上的不足后，神经网络作为模式识别方法的优点再次显现，重新成为语音识别研究和应用的热点.

移动互联网兴起之后，手机平台上出现了很多应用语音识别的软件，如苹果公司的 Siri、国内的众多语音助手软件等. 以国内的语音助手软件为例，它们往往要求用户以固定的模式、标准普通话朗读一定语音后才能给出较准确的识别，如“今天上海的天气怎么样”. 对于方言较重的用户，识别率则显著下降. 因此，汉语方言自动辨识成为语音识别领域亟待解决的问题之一. 国内这方面的研究也时有报道，如 2006 年顾明亮等人[4]，使用 GMM、N 元语言模型和简单神经网络对普通话和三种地方方言(吴方言，粤方言，闽方言)进行了分类识别，取得了 83.8%的识别率. 但总体而言，受限于大规模方言语料库建设等原因，目前离“识别任一方言”的理想目标还有一段距离.

要解决汉语方言输入识别问题，一个可能的方案是先识别方言的种属，即语音属于何种方言，然后在相应的方言库语音模型中给出识别结果. 考虑到普通神经网络表达能力有限，区别于文献[4]中使用简单神经网络模型(16×10×4，即 16 个输入结点，10 个隐层结点，4 个输出结点)识别 4 种地方方言的做法，本文使用当今机器学习领域前沿的 DNN 去尝试解决 9 种汉语方言种属语音识别问题，并在小数据集(9 种短时长典型汉语方言音频)上做了分类实验，其结果好于普通的神经网络识别. 该方法可用于移动互联网标准语音识别入口、方言口音鉴识等领域.

1　神经网络训练软件准备

QuickNet 是美国加州大学伯克利分校(UC Berkeley)国际计算机科学研究院(ISCI)语音项目组(Speech Group)开发的一款工作在 Linux 系统上的高性能神经网络训练软件，主要用于语音识别. 由于开源，支持使用 GPU 硬件和包含多种高效数学库等优点，该软件在世界语音识别界广为使用，也是本文实验采用的基础软件.

QuickNet 源代码主要包含了一些实现了神经网络算法的类(以大写 QN 开头的类，如 QN_MLP_BunchCudaVar. cu)和基于这些类的命令(以小写 qn 开头的文件，如 qnmultitrn. cc). 而由于 QuickNet 的最新版本只支持 5 层(即有 3 个隐层)以下神经网络训练，并且在神经网络反向传播计算权值更改量的时候没有加入动量项因子. 因此，根据训练 DNN 的需求，我们对 QuickNet 的源代码做出了相应的优化：① 从支持最多 5 层到支持最多 9 层；② 在反向传播计算权值修改量时加入动量项因子.

动量项因子(Momentum)是在 BP 算法反向传播计算权值更改量时加入的一个常用优化策略，以时间 t 时某个权值 $w(t)$ 为例，其公式是

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

$$\Delta w(t) = \alpha \Delta w(t-1) - \varepsilon \frac{\partial E}{\partial w}(t). \tag{1}$$

其中,α 为动量项因子(Momentum),$\Delta w(t-1)$为该权值在时间 $t-1$ 的更改量,ε 为学习率,$\frac{\partial E}{\partial w}(t)$为误差函数 E 对权值 $w(t)$的偏导数,$\Delta w(t)$即为该权值在本次的修改量. 而如果没有动量项,则有 $\Delta w(t) = -\varepsilon \frac{\partial E}{\partial w}(t)$. 研究表明,动量项因子可以加快神经网络训练时的收敛速度[5],基于优化过的 QuickNet 代码. 我们的实验结果也证明了这一点.

2 神经网络及 DNN 简介

2.1 神经网络简介

人工神经网络是一种模拟人类大脑神经网络行为特征,进行分布式并行信息处理的算法模型,主要用于预测和分类问题. 该模型由很多结点组成,结点之间用权值连接,权值的大小则代表了连接的重要程度. 在实践中,经常使用的是分层前馈型神经网络. 该网络模型一般由一个输入层、一个或多个隐层、一个输出层组成,输入信号分层向前传播,到达输出层后再依据 BP 算法(Back-propagation algorithm)反向传播,通过梯度下降法调整结点之间的权值以使网络的实际输出不断接近期望输出.

令 i 和 j 代表相邻两层的结点,且 $i<j$,w_{ji} 代表两者之间的连接权值,则结点 j 的输入为

$$v_j = \sum_{i=0}^{m} w_{ji} y_i. \tag{2}$$

其中,m 代表 i 所在层的所有结点,w_{j0}代表偏置,即输出恒为 1 的结点与 j 的连接权值. 结点 j 的输出 y_j 如式(3)所示,其中 $\varphi(x)$为 sigmoid 函数.

$$y_j = \varphi_j(v_j), \varphi(x) = \frac{1}{1+e^{-x}} \tag{3}$$

输入信号按公式(2)、(3)所述的方式到达输出层后,再从输出层开始做反向传播过程. 定义神经网络实际输出与期望输出的误差函数为

$$e_j = d_j - y_j, \tag{4}$$

$$\xi = \sum_{j \in C} \xi_j = \frac{1}{2} \sum_{j \in C} e_j^2. \tag{5}$$

C 代表所有输出层结点,d_j 和 y_j 分别代表输出层结点 j 的期望输出和实际输出. 我们通常把 ξ 看作权值的函数,反向传播过程(Back-propagation)即是通过梯度下降法在权值空间中搜索局部最优解的过程.

对于输出层和隐层之间的权值,依据链式法则及公式(2)—(5),有

$$\frac{\partial \xi}{\partial w_{ji}} = \frac{\partial \xi}{\partial e_j} \frac{\partial e_j}{\partial y_j} \frac{\partial y_j}{\partial v_j} \frac{\partial v_j}{\partial w_{ji}} = -e_j \varphi_j'(v_j) y_i. \tag{6}$$

再依据梯度下降法及公式(6),得到结点 i、j 间的权值更改量 Δw_{ji} 为

$$\Delta w_{ji} = -\eta \frac{\partial \xi}{\partial w_{ji}} = \eta e_j \varphi_j'(v_j) y_i. \tag{7}$$

其中,η 为学习率,即权值更改的幅度. 再令 h 为和 i 相连的结点,且 $h<i<j$,对于隐层和隐层之间的权值,有

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

$$\frac{\partial \xi}{\partial w_{ih}} = \frac{\partial \xi}{\partial y_i}\frac{\partial y_i}{\partial x_i}\frac{\partial x_i}{\partial w_{ih}} = \frac{\partial \xi}{\partial y_i}\varphi_i'(x_i)y_h. \tag{8}$$

考虑到结点 i 的输出会作用到下一层(即结点 j 所在层)所有结点，依据链式法则及公式(2)、(3)有

$$\frac{\partial \xi}{\partial y_i} = \sum_C \frac{\partial \xi}{\partial y_j}\frac{\partial y_j}{\partial x_j}\frac{\partial x_j}{\partial y_i} = \sum_C \frac{\partial \xi}{\partial y_j}\varphi_j'(x_j)w_{ji}. \tag{9}$$

其中 C 代表结点 j 所在层的所有结点. 如果结点 j 在输出层，则$\frac{\partial \xi}{\partial y_j}$可由公式(4)、(5)得出；如果结点 j 在隐层，则在反向传播过程中，$\frac{\partial \xi}{\partial y_j}$已经求出. 由公式(8)、(9)得结点 i、h 之间的权值更改量为

$$\Delta w_{ih} = -\eta\frac{\partial \xi}{\partial w_{ih}} = -\eta\left(\sum_C \frac{\partial \xi}{\partial y_j}\varphi_j'(x_j)w_{ji}\right)\varphi_i'(x_i)y_h. \tag{10}$$

神经网络训练过程中，一般需要经历多次上述计算过程，直到网络的实际输出和期望输出接近到我们所设定的阈值时训练结束.

2.2　DNN 简介

我们一般将超过一个隐层的神经网络模型称为深层神经网络，即 DNN. 长期以来，由于 DNN 采用传统 BP 算法训练时极易陷入局部极小值，因此成功的应用不多，未能引起人们的关注. 2006 年，Hinton 等人在文献[6]中为 Deep Belief Nets 提出了一种逐层贪心预训练(Layerwise greedy pre-training)的新方法，就此开创了 Deep learning 方向. 基于生物学上的启示，Deep learning 理念认为人们观察到的数据受到很多人们并不完全知道的因子的影响，并且这些因子是以层次结构组织的，它们分别对应了不同的抽象级别：较高层的表示需要通过转换或者产生较底层的表示来获得. 具体到神经网络上，我们先开始训练含有一个隐层的网络，然后保留训练好的权值，使网络层数加 1，接着训练含两个隐层的网络……以此类推，直到含有最大隐层的网络. 这样逐层训练完之后，得到的只是最大层神经网络权值的初始分布，因此还需要一次训练. 一般认为这样经过 pre-training 得到的 DNN，其权值的初始值比纯 BP 算法采用的随机初始权值更接近权值空间中的收敛值，因此比没有经过 pre-training 的神经网络在同等参数下有较高的识别率[7].

在方言种属识别的实验中，我们采用了一种类似的、有监督的 DNN pre-training 方法，如图 1 所示.

首先采用有监督的方式训练 3 层网络(即只含有一个隐层)，然后增加一个隐层，训练 4 层网络，其中，4 层网络的初始权值包含了 3 层时训练好的 $w1$ 以及随机化的权值 $w2$、$w3$，依次类推，直到 9 层网络(为了叙述方便起见，以上所说的权值包含了偏置，即输出恒为 1 的结点与下一层结点的连接权值). 这样的 pre-training 进行完之后，得到 9 层网络的初始权值，再进行一次 9 层网络的训练过程即可得到我们的网络模型.

3　基于 DNN 的方言识别

3.1　数据准备

为了验证及探索 DNN pre-training 方法的有效性，我们选取了 9 种短时长的汉语方言音频，分别是东北方言、客家方言、广东方言、河南方言、湖南方言、山东方言、山西方言、上海

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

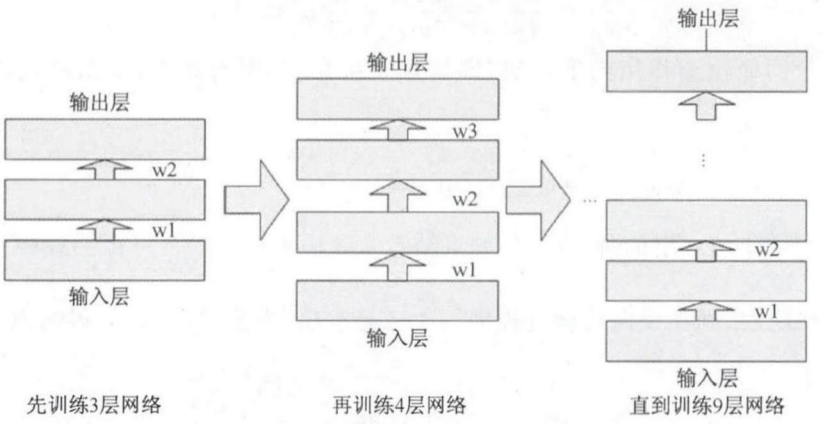

图 1 DNN pre-training 示意图

Fig. 1 The outline of DNN pre-training

方言和四川方言,总时长约为 70 min. 为了数据规整及处理方便,又将所有音频切割成 15 sec的小段,这样我们就得到了所有的原始音频素材.

3.2 特征提取及训练、测试文件生成

我们选择音频的 MFCC(Mel Frequency Cepstrum Coefficient)参数作为神经网络的输入特征. 该参数基于人的听觉特性提出,与频率(Hz)呈非线性对应关系,是语音识别领域应用较广的参数之一. 已经有一批流行的软件用于求解该参数,我们选用了最早由剑桥大学工程院(CUED)开发的 HTK 工具箱作为我们的基本工具. HTK 工具箱(Hidden Markov Model Toolkit)最早是为了建立和训练隐马尔科夫模型所用,其中包含了一些用于处理音频的实用软件,如 HCopy 等即可用于求解 MFCC 参数.

在得到所有音频的 MFCC 参数后,又使用 ISCI 开发的 feacat 等软件将所有音频提取的参数生成 PFile 格式的数据集,训练集和测试集分别生成. PFile 格式是一种经常用于存储语音参数的二进制文件格式,其特点是每一个句子(Sentence)和每一个帧(Frame)的特征与标注(label)在一个文件里,并存储在同一行. 在实验中,一个句子即为一段15 sec的音频,求解 MFCC 参数时会把每段音频划分成若干帧,然后对每帧计算参数. 为了避免神经网络存储不同方言音频的呈现顺序信息,又对训练数据按帧进行了随机化,以保证实验的科学性.

3.3 训练过程(含 pre-training 和 training)

采用 QuickNet 软件进行神经网络训练,从 3 层开始一直到 9 层,做逐层的有监督 pre-training,当最大层的 pre-training 做完后,得到 9 层网络的初始权值,再对 9 层网络进行训练即得到我们的识别模型. 部分训练细节在第 2 节中有详细介绍.

为了充分验证及探索该方法的有效性,我们做了多组对比实验,如不使用 pre-training 和加入 pre-training 的对比、不同训练层数的对比等.

采用 39 维的 MFCC 系数作为神经网络的输入层,9 种方言类别作为输出层,隐层则探索了多种组成. 动量项因子设为 0.9. 训练中为了加快训练过程则采用 mini-batch 的训练方式,其大小为 16. 默认的学习率为 0.008,之后也探索了其他值. 输出层结点采用了 softmax 函数,其公式是

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

$$p_i = \frac{\exp(q_i)}{\sum_{j=1}^{n}\exp(q_j)}. \tag{11}$$

其中，n 代表有个 n 输出层结点，p_i 代表结点 i 的输出，q_i 代表结点 i 的输入. 由该公式可知，所有输出层结点的输出之和为 1，实际上代表了输入所属模式的后验概率分布.

3.4　测试

在上述每个实验做完后，我们都使用独立的测试集对训练好的模型进行测试，以帧识别率(Frame accuracy)作为我们对比的主要依据.

4　实验结果与分析

首先对未加 pre-training 的 BP 神经网络进行了训练，如图 2 所示.

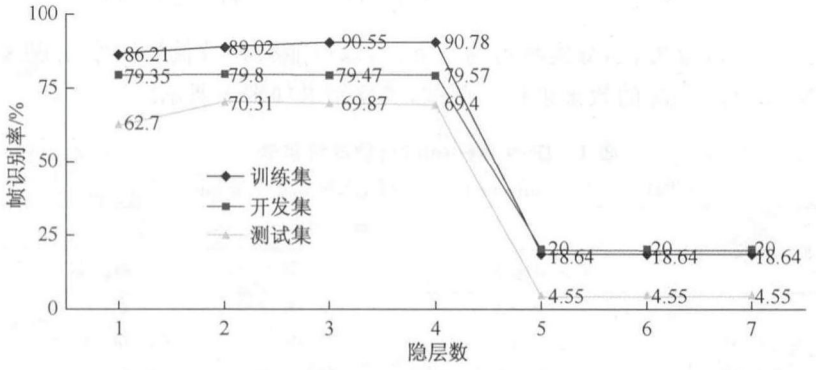

图 2　未加 pre-training 的 BP 网络帧识别率图

Fig. 2　Frame accuracy of BP network without pre-training

其中，三种折线分别代表在训练集、开发集和测试集上的帧识别率，横轴则代表从含有 1 个隐层的网络到含有 7 个隐层的网络. 从图中可以看出，当隐层数适量增加时，神经网络对数据的识别能力随之增强，但如果层数过深，网络的性能则急剧下降，如图 1 中隐层数超过 5 层时. 一般认为，此时的深层神经网络极易陷入局部极小值(Local minima)从而失去泛化能力，这也是 DNN 长期以来没有有效训练方法的原因[8]. 但对于具有高度非线性映射特性的问题，简单的 3 层神经网络往往无法解决问题，因此需要充分探索深层网络的表达能力. DNN 逐层 pre-training 方法的提出开创性地解决了这个问题，也是我们实验的主要内容.

采用 pre-training 训练方法和采用单纯 BP 算法(pure BP)在测试集上的帧识别率对比如图 3 所示.

从图 3 可以看出，采用逐层 pre-training 方法的神经网络在隐层数增加时表现出明显的优势，比采用随机权值初始化的纯 BP 网络在深层的识别率提高了几十倍，并且保持了基本稳定的识别性能. 而在层数较低时，采用 pre-training 方法的神经网络，其识别率比纯 BP 神经网络的识别率略低. 这可能和问题的规模有关，当问题的规模较小，例如分类问题的类别数较少时，用较少层的神经网络即可达到比较好的效果，过大的神经网络反而导致在训练集上“过拟合”从而影响测试集的识别性能. 具体到我们的识别任务，因为采用短时长的训练数

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

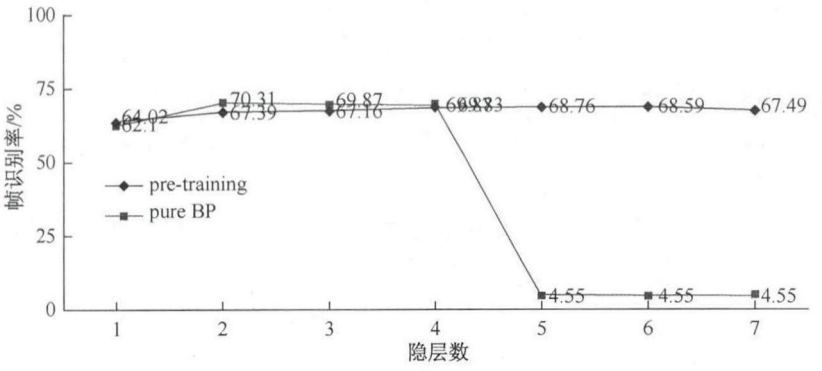

图 3 加 pre-training 的 BP 网络与纯 BP 网络帧识别率对比图

Fig. 3 Frame accuracy of BP network with pre-training vs pure BP

据测试 DNN pre-training 方法,分类数目为 9 类,所以可能会在较低层网络出现这种情况. 我们又对 DNN pre-training 的效果进行了调试,实验结果如表 1 所示.

表 1 DNN pre-training 调试结果表

Tab. 1 Fine-tuning results of DNN pre-training

神经网络模型	帧识别率/%		
	训练集	开发集	测试集
7×250	98.83	78.36	67.49
7×150	95.61	77.49	66.93
7×100	90.95	78.03	68.00
+ learnrate = 0.004	92.06	78.14	68.17
+ learnrate = 0.001	88.93	79.46	70.17

其中,神经网络模型中,"7×250"代表神经网络中有 7 个隐层,每个隐层有 250 个结点,"7×150"、"7×100"同理类推. "+ learnrate = 0.004"表示在"7×100"的网络基础上将学习率调整为 0.004,而之前的学习率是 0.008(pre-training 和 fine-tuning 阶段相同),"+ learnrate = 0.001"同理类推. 调试结果则用三个集合上的帧识别率表示. 从表 1 可以看出,采用适当的隐层结点数(即神经网络规模)可以提高识别率,网络大小要尽可能与问题和数据规模相匹配以避免过拟合或者训练不充分. 学习率对神经网络的性能也有较明显的影响,较大的学习率可能会加快收敛过程,也可能导致在局部极小值来回震荡,合适大小的学习率可以保证神经网络稳定地收敛到某一个最优解,如表中"+ learnrate = 0.001"所示.

5 总结及展望

为了让计算机特别是智能移动设备在任何情况下真正听懂人的语音,未来还需要很多工作去做. 汉语方言识别可能是未来汉语语音识别需要特别注意解决的重要问题之一. 在本文中,我们采用 DNN pre-training 的方法对解决方言种属问题进行了初步尝试,实验结果表明 DNN 逐层预训练的方法为提取数据的深层次特征提供了有力的手段,是大数据、复杂映射条件下极具潜力的选择.

本文下一步的工作将集中于以下两方面:①获取更多方言语音资源,构建方言语音数据

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

库.②将方言语音种属识别和方言语音识别结合起来，逐步做到对非特定人、非特定口音的语音识别.

[参 考 文 献]

[1] HINTON G, DENG L, YU D, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups[J]. Signal Processing Magazine, IEEE, 2012, 29(6): 82-97.

[2] BAKER J. The DRAGON system-An overview[J]. Acoustics, Speech and Signal Processing, IEEE Transactions on, 1975, 23(1): 24-29.

[3] OH K S, JUNG K. GPU implementation of neural networks[J]. Pattern Recognition, 2004, 37(6): 1311-1314.

[4] 顾明亮，沈兆勇. 基于语音配列的汉语方言自动辨识[J]. 中文信息学报，2006，20(5)：77-82.

[5] RUMELHART D E, HINTON G E, WILLIAMS R J. Learning representations by back-propagating errors[J]. Nature, 1986, 323(6088): 533-536.

[6] HINTON G E, OSINDERO S, TEH Y W. A fast learning algorithm for deep belief nets[J]. Neural Computation, 2006, 18(7): 1527-1554.

[7] LAROCHELLE H, BENGIO Y, LOURADOUR J, et al. Exploring strategies for training deep neural networks [J]. The Journal of Machine Learning Research, 2009(10): 1-40.

[8] BENGIO Y, LAMBLIN P, POPOVICI D, et al. Greedy layer-wise training of deep networks[J]. Advances in Neural Information Processing Systems, 2007(19): 153.

(C)1994-2023 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net