



## (12)发明专利申请

(10)申请公布号 CN 109102126 A

(43)申请公布日 2018.12.28

(21)申请号 201810999797.9

(22)申请日 2018.08.30

(71)申请人 燕山大学

地址 066004 河北省秦皇岛市海港区河北大街西段438号

(72)发明人 卢志刚 杨英杰 丁艺楠 顾媛媛 杨宇

(74)专利代理机构 北京挺立专利事务所(普通合伙) 11265

代理人 刘阳

(51)Int.Cl.

G06Q 10/04(2012.01)

G06Q 50/06(2012.01)

G06N 3/08(2006.01)

权利要求书4页 说明书10页 附图2页

## (54)发明名称

一种基于深度迁移学习的理论线损率预测模型

## (57)摘要

本发明公开了一种基于深度迁移学习的理论线损率预测模型,涉及人工智能算法在电力系统应用的技术领域,本发明包括以下步骤:首先将训练好的深度线损率学习模型作为源模型,把训练好的底层的DBN深度信念网络直接迁移到待预测的模型中(冻结)。然后,为避免深度学习模型训练时陷入局部最优,引入迁移学习的概念,结合预测的数据,通过MMD方法度量源、目标数据的分布差异并筛选源训练数据,用筛选的训练数据微调已训练好的DNN深度神经网络,最终得到基于TDBN-DNN的深度迁移学习模型。最后用电网运行数据作为模型输入来进行线损率预测;解决了在电网坚强、运行高效、节能环保的要求和智能电网的构建问题。



1. 一种基于深度迁移学习的理论线损率预测的方法,其特征在于,所述方法包括如下步骤:

步骤1,建立由多个RBM模型堆叠而成的DBN深度置信网络;其中,每个RBM模型所得对数似然函中的参数 $\theta$ 需用对比散度算法求导;

步骤2,将DBN深度置信网络的输出层连接到DNN模型的输入层,组成DBN-DNN深度学习模型;其中,DNN模型由多层普通神经网络组成,且最后一层为输出层;

步骤3,冻结DBN-DNN深度学习模型深度网络中的下层DBN,然后用MMD法度量源数据与任务预测数据的分布距离,并迁移出源样本中 $p_i > 0$ 的数据,获得基于迁移出的数据微调DBN-DNN中的DNN结构的模型,即TDBN-DNN迁移深度学习模型;

步骤4,利用TDBN-DNN迁移深度学习模型来模拟电网运行时的负荷数据、电源数据和母线电压数据与线损率之间的非线性映射关系,并对线损率进行预测。

2. 根据权利要求1所述的一种基于深度迁移学习的理论线损率预测的方法,其特征在于,所述步骤1的具体步骤如下:

#### 一、建立DBN深度置信网络

RBM模型是一个能量模型,对于一组已知的状态 $(v, h)$ ,它的能量函数定义为 $E_\theta(v, h)$ ,其中 $\theta$ 为网络参数 $\theta = \{a_i, b_j, w_{ij}\}$ ,它的隐含层与可视层的联合概率分布定义为 $p(v, h)$ ,即:

$$E_\theta(v, h) = -\sum_{i=1}^n \sum_{j=1}^m w_{ij} v_i h_j - \sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j \quad (1)$$

$$P(v, h) = \frac{1}{Z_\theta} e^{-E(v, h)} \quad (2)$$

式(1)中 $m$ 为输入单元个数, $n$ 为输出单元的个数, $a_i$ 为每个输入单元实值偏置量, $b_j$ 为每个隐单元的实值偏置, $v_i$ 用来表示输入数据, $h_i$ 为隐单元输出数据, $w_{ij}$ 为实值权重矩阵;

式(2)中 $Z_\theta$ 为配分函数, $Z_\theta = \sum_{v, h} e^{-E_\theta(v, h)}$ ;由受限波尔兹曼机各层之间的条件独立性可知,当给定输入数据时,输出层节点取值满足如下条件概率:

$$P(h_j = 1 | v, \theta) = \varphi(b_j + \sum_{i=1}^n w_{ij} v_i) \quad (3)$$

式(3)中 $\varphi(o) = \frac{1}{1 + \exp(-o)}$ 为sigmoid激活函数;输出层的数据确定后,输入节点的取值条件概率为:

$$P(v_i = 1 | v, \theta) = \varphi(a_i + \sum_{j=1}^m w_{ij} h_j) \quad (4)$$

给定一组训练样本 $G = \{g_1, g_2, \dots, g_s\}$ ,训练RBM模型,调整参数 $\theta$ ,以拟合给定的训练样本,通过最大化网络的似然函数 $L(\theta)$ 得到参数 $\theta$ ,即 $L(\theta) = \prod_{i=1}^n p(g_i)$ ,为简化计算,将其写为对数形式为:

$$\ln L(\theta) = \sum_{i=1}^n \ln p(g_i) \quad (5)$$

$$\theta = \arg \max L(\theta) = \arg \max \sum_{i=1}^n p(g_i) \quad (6)$$

二、CD训练算法

用对比散度算法对数似然函中的参数 $\theta$ 求导,如下:

$$\frac{\partial \ln p(v_i|\theta)}{\partial w_{ij}} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} \quad (7)$$

$$\frac{\partial \ln p(v_i|\theta)}{\partial a_i} = \langle v_i \rangle_{data} - \langle v_i \rangle_{model} \quad (8)$$

$$\frac{\partial \ln p(v_i|\theta)}{\partial b_j} = \langle h_j \rangle_{data} - \langle h_j \rangle_{model} \quad (9)$$

各参数的更新准则为:

$$\Delta w_{ij} = \epsilon_{CD} (\langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{recon}) \quad (10)$$

$$\Delta a_i = \epsilon_{CD} (\langle v_i \rangle_{data} - \langle v_i \rangle_{recon}) \quad (11)$$

$$\Delta b_j = \epsilon_{CD} (\langle h_j \rangle_{data} - \langle h_j \rangle_{recon}) \quad (12)$$

式(11) — (12) 中,  $\epsilon_{CD}$  为学习步长,  $\langle h_j \rangle_{recon}$  表示一步重构后的模型定义分布。

3. 根据权利要求1所述的一种基于深度迁移学习的理论线损率预测的方法,其特征在于,步骤2的具体步骤过程如下:

一、前向传播过程

$$f(p) = \frac{1}{1 + e^{-p}} \quad (13)$$

$$f(p) = p \quad (14)$$

$$h_{w,b}(p) = \delta(wp + b) \quad (15)$$

$$h_{w,b}(p_j)^{(l)} = 1 / \left( 1 + e^{-\left( \sum_i p_i^{l-1} \times w_{ij}^l + b_j^l \right)} \right) \quad (16)$$

$$h_{w,b}(p_j)^{k_l} = \sum_i p_i^{k_l-1} * w_{ij}^{k_l} + b_j^{k_l} \quad (17)$$

式(16)中:  $w_{ij}^l$  代表第1层第j节点与第1-1层第i节点之间的联接权重参数;  $b_j^l$  代表第1层第j节点的截距项;

式(17)中  $h_{w,b}(p_j)^{k_l}$  代表第 $k_l$ 层第j节点的输出值;

二、反向传播过程

对于作为输出层的第 $k_l$ 层的输出单元i,其残差计算公式如下:

$$\delta_i^{(k_l)} = \frac{\partial}{\partial z_i^{k_l}} \frac{1}{2} Ph_{w,b}(p) - tP^2 = -(t - h_{w,b}(p)^{k_l}) \bullet f'(z_i^{(k_l)}) \quad (18)$$

对于  $l = k_{l-1}, k_{l-2}, k_{l-3}, \dots, 2$  的各层,第1层的第i节点的残差计算式如下:

$$\delta_i^{(l)} = - \left( \sum_{j=1}^{s_{l+1}} w_{ji}^{(l)} \delta_j^{(l+1)} \right) \cdot f' \left( z_i^{(l)} \right) \quad (19)$$

对于一个固定训练样本集  $\{(u^1, y^1), \dots, (u^c, y^c)\}$ , 包含  $c$  个样本, 利用小批量梯度下降法求解深层神经网络, 其损失函数为:

$$J(w, b) = \left[ \frac{1}{c} \sum_{i=1}^c \frac{1}{2} p h_{w,b}(p) - t P^2 \right] + \frac{\lambda}{2} \sum_{l=1}^{k_{l-1}} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} (w_{ij}^{(l)})^2 \quad (20)$$

式 (20) 中, 权重衰减参数  $\lambda$  用于控制式 (20) 中两项的重要性; 因此, 小批量梯度下降法中每次迭代对于参数  $w$  和  $b$  的更新如下:

$$w^{(l)} = w^{(l)} - \beta^{(l)} \left[ \left( \frac{1}{c} (\Delta w^{(l)} + \delta^{(l+1)} (h_{w,b}(p)^{(l)})^T) \right) + \lambda w^{(l)} \right] \quad (21)$$

$$b^{(l)} = b^{(l)} - \beta^{(l)} \left[ \frac{1}{c} (\Delta b^{(l)} + \delta^{(l+1)}) \right] \quad (22)$$

4. 根据权利要求1所述的一种基于深度迁移学习的理论线损率预测的方法, 其特征在于, 步骤3中的具体过程如下:

一、度量源数据与任务数据分布差异:

假设分别存在一个满足  $d$  分布的源数  $X^{(s)} = [x_1^{(s)}, x_2^{(s)}, \dots, x_N^{(s)}]$  和一个满足  $q$  分布的目标数据  $X^{(t)} = [x_1^{(t)}, x_2^{(t)}, \dots, x_M^{(t)}]$ ;  $X^{(s)}$ ,  $X^{(t)}$  的最大均值差异MMD可以表示为:

$$MMD = \left\| \frac{1}{N_s} \sum_{i=1}^{N_s} \eta(x_i^{(s)}) - \frac{1}{M_t} \sum_{j=1}^{M_t} \eta(x_j^{(t)}) \right\| \quad (23)$$

式 (23) 中  $H$  表示再生核希尔伯特空间RKHS,  $\Phi(\cdot): X \rightarrow H$  表示原始特征空间映射到RKHS的非线性特征映射核函数; 由于RKHS通常是高维, 甚至是无限空间, 对应的核选择表示无穷维的高斯核:

$$k(x, x') = \exp \left( - \frac{\|x - x'\|^2}{2\sigma^2} \right) \quad (24)$$

式 (24) 其中  $\sigma$  函数的宽度参数; 在此用MMD的平方形式, 即  $MMD^2$ ; 将其展开为:

$$\begin{aligned} MMD^2 &= \frac{1}{M_t^2} \sum_{i=1}^{M_t} \sum_{j=1}^{M_t} \eta(x_i)^T \eta(x_j) + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} \eta(x_i)^T \eta(x_j) - \frac{1}{N_t M_t} \sum_{i=1}^{N_t} \sum_{j=1}^{M_t} \eta(x_i)^T \eta(x_j) \\ &= \frac{1}{M_t^2} \sum_{i=1}^{M_t} \sum_{j=1}^{M_t} \langle \eta(x_i), \eta(x_j) \rangle + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} \langle \eta(x_i), \eta(x_j) \rangle - \frac{1}{N_t M_t} \sum_{i=1}^{N_t} \sum_{j=1}^{M_t} \langle \eta(x_i), \eta(x_j) \rangle \end{aligned} \quad (25)$$

在此引入核技巧:

$$\langle \eta(x), \eta(x') \rangle_H = \langle K(x, \cdot), K(\cdot, x') \rangle_H = K(x, x') \quad (26)$$

式 (25) 简化为:

$$MMD^2 = \frac{1}{M_t^2} \sum_{i=1}^{M_t} \sum_{j=1}^{M_t} K(x_i, x_j) + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} K(x_i, x_j) - \frac{1}{N_t M_t} \sum_{i=1}^{N_t} \sum_{j=1}^{M_t} K(x_i, x_j) \quad (27)$$

### 三、最大均值差异贡献系数迁移源数据

为了找出源数据和目标数据不相关或相关性不大的数据,在此,通过定义每个源数据的最大均值差异贡献系数CCMMD来筛选源数据;

用 $\rho_i$ 表示第 $i$ 个样本的最大均值差异贡献系数;假设缺少第 $i$ 个源数据样本的 $MMD_{\gamma \neq i}$ 为:

$$MMD_{\gamma \neq i} = \left\| \frac{1}{N_s} \sum_{i=1, \gamma \neq i}^{N_s} \eta(x_i^{(s)}) - \frac{1}{M_t} \sum_{j=1}^{M_t} \eta(x_j^{(t)}) \right\| \quad (28)$$

式(28)中 $MMD_{\gamma \neq i}$ 表示缺少第 $i$ 个源数据样本的最大均值距离,其中 $\gamma = 1, 2, \dots, N_s$ 则

$$\rho_i = \frac{(MMD_{\gamma \neq i} - MMD)}{MMD} \quad (29)$$

式(28)中 $\rho_i$ 表示第 $i$ 个样本的最大均值差异贡献系数,若 $\rho_i > 0$ ,说明第 $i$ 个样本是对MMD有贡献的,相反若 $\rho_i \leq 0$ ,说明第 $i$ 个样本是对MMD有“负”贡献的;通过计算每个源数据的 $\rho_i$ ,迁移出 $\rho_i > 0$ 的样本数据,得到和目标数据分布更接近的源数据;

四、冻结DBN-DNN的DBN层,获得基于迁移出的数据微调DBN-DNN中的DNN结构的模型,即TDBN-DNN迁移深度学习模型。

5. 根据权利要求1所述的一种基于深度迁移学习的理论线损率预测的方法,其特征在于,步骤4的具体步骤如下:

#### (1) 数据处理

对输入数据负荷数据、电源出力数据、母线电压数据标准化处理,如式(32):

$$\alpha'_{i,j} = \frac{\alpha_{i,j} - \alpha_{\min,j}}{\alpha_{\max,j} - \alpha_{\min,j}} \quad (32)$$

式(32)中, $\alpha_{i,j}$ 代表第 $j$ 个数据特征的第 $i$ 个样本数据, $i = 1, 2, \dots, MC$ ,

$j = 1, 2, \dots, 2r$ ,  $\alpha'_{i,j}$ 代表第 $j$ 个数据特征的第 $i$ 个样本数据标准化后的数据, $\alpha_{\min,j}$ 代表第 $j$ 个数据特征在样本数据中的最小值, $\alpha_{\max,j}$ 代表第 $j$ 个数据特征在样本数据中的最大值;

(2) TDBN-DNN迁移深度学习模型模拟模型来模拟电网运行时的负荷数据、电源数据和母线电压数据与线损率之间的非线性映射关系

根据样本的规模及实时性要求,确定TDBN-DNN迁移深度学习模型的隐藏层数量及其节点数量,并初始化网络参数;然后用标准化后的负荷数据、电源出力数据、母线电压数据样本数据为输入数据,对应断面线损率为标签数据,利用贪心无监督学习算法逐层对整个DBN深度置信网络模型逐层训练;最后将DBN的输出特征向量给DNN一个较好的初值,采用BP算法用对应断面线损率有监督地训练DNN来拟合标签数据;

#### (3) TDBN-DNN迁移深度学习模型预测线损率

对任务目标输入数据进行标准化处理后,将此数据输入TDBN-DNN线损率预测模型,得到预测线损率值。

## 一种基于深度迁移学习的理论线损率预测模型

### 技术领域

[0001] 本发明涉及人工智能算法在电力系统应用的技术领域,具体涉及一种基于深度迁移学习的理论线损率预测的方法。

### 背景技术

[0002] 在2017年12月6日中国电科院召开的主题为“人工智能在电力领域的应用研究方向和关键”的二零八科学会议中,多为专家就人工智能技术进展及在电力系统中的应用实践和展望做专题报告,并初步达成共识,重点研究深度学习,强化学习,迁移学习,小样本学习等理论与技术应用。目前,由于电网坚强、运行高效、节能环保的要求和智能电网的构建,电网运行中产生大量数据,因此寻找一种新的快速计算方法是线损研究领域亟待解决的问题。难点是如何提高多维数据处理的速度与精度问题,以保证理论线损率计算的时效性与准确性。深度学习理论作为模式识别和机器学习领域最新的研究成果,以强大的建模和表征能力在图像和语音处理等领域的大数据处理方面取得了丰硕的成果。

[0003] 然而,传统机器学习方法通常存在两个前提假设:(1)训练数据集和测试数据集中样本必须遵循相同的概率分布(2)需要有充足的训练样本(数以万计)。这两个前提假设在实际应用中往往是不成立的。迁移学习出现成功解决了传统的机器学习上述的局限性。

[0004] 综上所述,有必要提出一种深度迁移学习的线损率算法。

### 发明内容

[0005] 本发明的目的在于提供一种基于深度迁移学习的理论线损率预测的方法。利用解决了传统机器学习上述的两个问题的迁移学习;以解决在电网坚强、运行高效、节能环保的要求和智能电网的构建。

[0006] 在电网运行中产生大量数据的背景下,为解决上述技术问题,本发明所采取的技术方案是:一种基于深度迁移学习的理论线损率预测的方法,其特征在于,所述方法包括如下步骤:

[0007] 步骤1,建立由多个RBM模型堆叠而成的DBN深度置信网络;其中,每个RBM模型所得对数似然函中的参数 $\theta$ 需用对比散度算法求导;

[0008] 步骤2,将DBN深度置信网络的输出层连接到DNN模型的输入层,组成DBN-DNN深度学习模型;其中,DNN模型由多层普通神经网络组成,且最后一层为输出层;

[0009] 步骤3,冻结DBN-DNN深度学习模型深度网络中的下层DBN,然后用MMD法度量源数据与任务预测数据的分布距离,并迁移出源样本中 $p_i > 0$ 的数据,获得基于迁移出的数据微调DBN-DNN中的DNN结构的模型,即TDBN-DNN迁移深度学习模型;

[0010] 步骤4,利用TDBN-DNN迁移深度学习模型来模拟电网运行时的负荷数据、电源数据和母线电压数据与线损率之间的非线性映射关系,并对线损率进行预测。

[0011] 进一步的技术方案在于,所述步骤1的具体步骤为如下:

[0012] 一、建立DBN深度置信网络

[0013] RBM模型是一个能量模型,对于一组已知的状态 $(v, h)$ ,它的能量函数定义为 $E_\theta(v, h)$ ,其中 $\theta$ 为网络参数 $\theta = \{a_i, b_j, w_{ij}\}$ ,它的隐含层与可视层的联合概率分布定义为 $p(v, h)$ ,即:

$$[0014] \quad E_\theta(v, h) = -\sum_{i=1}^n \sum_{j=1}^m w_{ij} v_i h_j - \sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j \quad (1)$$

$$[0015] \quad P(v, h) = \frac{1}{Z_\theta} e^{-E_\theta(v, h)} \quad (2)$$

[0016] 式(1)中 $m$ 为输入单元个数, $n$ 为输出单元的个数, $a_i$ 为每个输入单元实值偏置量, $b_j$ 为每个隐单元的实值偏置, $v_i$ 用来表示输入数据, $h_j$ 为隐单元输出数据, $w_{ij}$ 为实值权重矩阵;

[0017] 式(2)中 $Z_\theta$ 为配分函数, $Z_\theta = \sum_{v, h} e^{-E_\theta(v, h)}$ ;由受限波尔兹曼机各层之间的条件独立性可知,当给定输入数据时,输出层节点取值满足如下条件概率:

$$[0018] \quad P(h_j = 1 | v, \theta) = \varphi(b_j + \sum_{i=1}^n w_{ij} v_i) \quad (3)$$

[0019] 式(3)中 $\varphi(o) = \frac{1}{1 + \exp(-o)}$ 为sigmoid激活函数;输出层的数据确定后,输入节点的取值条件概率为:

$$[0020] \quad P(v_i = 1 | h, \theta) = \varphi(a_i + \sum_{j=1}^m w_{ij} h_j) \quad (4)$$

[0021] 给定一组训练样本 $G = \{g_1, g_2, \dots, g_s\}$ ,训练RBM模型,调整参数 $\theta$ ,以拟合给定的训练样本,通过最大化网络的似然函数 $L(\theta)$ 得到参数 $\theta$ ,即 $L(\theta) = \prod_{i=1}^n p(g_i)$ ,为简化计算,将其写为对数形式为:

$$[0022] \quad \ln L(\theta) = \sum_{i=1}^n \ln p(g_i) \quad (5)$$

$$[0023] \quad \theta = \arg \max L(\theta) = \arg \max \sum_{i=1}^n p(g_i) \quad (6)$$

[0024] 二、CD训练算法

[0025] 用对比散度算法对对数似然函中的参数 $\theta$ 求导,如下:

$$[0026] \quad \frac{\partial \ln p(v_i | \theta)}{\partial w_{ij}} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} \quad (7)$$

$$[0027] \quad \frac{\partial \ln p(v_i | \theta)}{\partial a_i} = \langle v_i \rangle_{data} - \langle v_i \rangle_{model} \quad (8)$$

$$[0028] \quad \frac{\partial \ln p(v_i | \theta)}{\partial b_j} = \langle h_j \rangle_{data} - \langle h_j \rangle_{model} \quad (9)$$

[0029] 各参数的更新准则为:

$$[0030] \quad \Delta w_{ij} = \varepsilon_{CD} (\langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{recon}) \quad (10)$$

[0031]  $\Delta a_i = \varepsilon_{CD} (\langle v_i \rangle_{data} - \langle v_i \rangle_{recon})$  (11)

[0032]  $\Delta b_j = \varepsilon_{CD} (\langle h_j \rangle_{data} - \langle h_j \rangle_{recon})$  (12)

[0033] 式(11) — (12) 中,  $\varepsilon_{CD}$  为学习步长,  $\langle h_j \rangle_{recon}$  表示一步重构后的模型定义分布。

[0034] 进一步的技术方案在于, 步骤2的具体步骤过程如下:

[0035] 将DNN的输入层连接到DBN的输出层, 建立DBN-DNN深度学习模型, 其中DNN的训练过程包括前向传播和反向传播, 内容如下:

[0036] 一、前向传播过程

[0037]  $f(p) = \frac{1}{1 + e^{-p}}$  (13)

[0038]  $f(p) = p$  (14)

[0039]  $h_{w,b}(p) = \delta(wp + b)$  (15)

[0040] 
$$h_{w,b}(p_j)^{(l)} = 1 / \left( 1 + e^{-\left( \sum_i p_i^{l-1} \times w_{ij}^l + b_j^l \right)} \right)$$
 (16)

[0041] 
$$h_{w,b}(p_j)^{k_l} = \sum_i p_i^{k_l-1} * w_{ij}^{k_l} + b_j^{k_l}$$
 (17)

[0042] 式(16)中:  $w_{ij}^l$  代表第1层第j节点与第1-1层第i节点之间的联接权重参数;  $b_j^l$  代表第1层第j节点的截距项;

[0043] 式(17)中  $h_{w,b}(p_j)^{k_l}$  代表第 $k_l$ 层第j节点的输出值;

[0044] 二、反向传播过程

[0045] 对于作为输出层的第 $k_l$ 层的输出单元i, 其残差计算公式如下:

[0046] 
$$\delta_i^{(k_l)} = \frac{\partial}{\partial z_i^{k_l}} \frac{1}{2} Ph_{w,b}(p) - tP^2 = -(t - h_{w,b}(p)^{k_l}) \bullet f'(z_i^{(k_l)})$$
 (18)

[0047] 对于  $l = k_{l-1}, k_{l-2}, k_{l-3}, \dots, 2$  的各层, 第1层的第i节点的残差计算式如下:

[0048] 
$$\delta_i^{(l)} = - \left( \sum_{j=1}^{s_{l+1}} w_{ji}^{(l)} \delta_j^{(l+1)} \right) \bullet f'(z_i^{(l)})$$
 (19)

[0049] 对于一个固定训练样本集  $\{(u^1, y^1), \dots, (u^c, y^c)\}$ , 包含c个样本, 利用小批量梯度下降法求解深层神经网络, 其损失函数为:

[0050] 
$$J(w, b) = \left[ \frac{1}{c} \sum_{i=1}^c \frac{1}{2} Ph_{w,b}(p) - tP^2 \right] + \frac{\lambda}{2} \sum_{l=1}^{k_{l-1}} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} (w_{ij}^{(l)})^2$$
 (20)

[0051] 式(20)中, 权重衰减参数 $\lambda$ 用于控制式(20)中两项的重要性; 因此, 小批量梯度下降法中每次迭代对于参数w和b的更新如下:

[0052]

$$w^{(l)} = w^{(l)} - \beta^{(l)} \left[ \left( \frac{1}{c} (\Delta w^{(l)} + \delta^{(l+1)} (h_{w,b}(p)^{(l)})^T) \right) + \lambda w^{(l)} \right] \quad (21)$$



$$[0053] \quad b^{(l)} = b^{(l)} - \beta^{(l)} \left[ \frac{1}{c} (\Delta b^{(l)} + \delta^{(l+1)}) \right] \quad (22)$$

[0054] 进一步的技术方案在于,,步骤3的具体过程如下:

[0055] 一、度量源数据与任务数据分布差异:

[0056] 假设分别存在一个满足d分布的源数  $X^{(s)} = [x_1^{(s)}, x_2^{(s)}, \dots, x_N^{(s)}]$  和一个满足q分布的目标数据  $X^{(t)} = [x_1^{(t)}, x_2^{(t)}, \dots, x_M^{(t)}]$ ;  $X^{(s)}, X^{(t)}$  的最大均值差异MMD可以表示为:

$$[0057] \quad MMD = \left\| \frac{1}{N_s} \sum_{i=1}^{N_s} \eta(x_i^{(s)}) - \frac{1}{M_t} \sum_{j=1}^{M_t} \eta(x_j^{(t)}) \right\| \quad (23)$$

[0058] 式(23)中H表示再生核希尔伯特空间RKHS,  $\Phi(\cdot): X \rightarrow H$  表示原始特征空间映射到RKHS的非线性特征映射核函数;由于RKHS通常是高维,甚至是无限空间,对应的核选择表示无穷维的高斯核:

$$[0059] \quad k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) \quad (24)$$

[0060] 式(24)其中 $\sigma$ 函数的宽度参数;在此用MMD的平方形式,即 $MMD^2$ ;将其展开为:

[0061]

$$\begin{aligned} MMD^2 &= \frac{1}{M_t^2} \sum_{i=1}^{M_t} \sum_{j=1}^{M_t} \eta(x_i)^T \eta(x_j) + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} \eta(x_i)^T \eta(x_j) - \frac{1}{N_t M_t} \sum_{i=1}^{N_t} \sum_{j=1}^{M_t} \eta(x_i)^T \eta(x_j) \\ &= \frac{1}{M_t^2} \sum_{i=1}^{M_t} \sum_{j=1}^{M_t} \langle \eta(x_i), \eta(x_j) \rangle + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} \langle \eta(x_i), \eta(x_j) \rangle - \frac{1}{N_t M_t} \sum_{i=1}^{N_t} \sum_{j=1}^{M_t} \langle \eta(x_i), \eta(x_j) \rangle \end{aligned} \quad (25)$$

[0062] 在此引入核技巧:

$$[0063] \quad \langle \eta(x), \eta(x') \rangle_H = \langle K(x, \cdot), K(\cdot, x') \rangle_H = K(x, x') \quad (26)$$

[0064] 式(25)简化为:

$$[0065] \quad MMD^2 = \frac{1}{M_t^2} \sum_{i=1}^{M_t} \sum_{j=1}^{M_t} K(x_i, x_j) + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} K(x_i, x_j) - \frac{1}{N_t M_t} \sum_{i=1}^{N_t} \sum_{j=1}^{M_t} K(x_i, x_j) \quad (27)$$

[0066] 三、最大均值差异贡献系数迁移源数据

[0067] 为了找出源数据和目标数据不相关或相关性不大的数据,在此,通过定义每个源数据的最大均值差异贡献系数CCMMD来筛选源数据;

[0068] 用 $\rho_i$ 表示第i个样本的最大均值差异贡献系数;假设缺少第i个源数据样本的 $MMD_{\gamma \neq i}$ 为:

$$[0069] \quad MMD_{\gamma \neq i} = \left\| \frac{1}{N_s} \sum_{i=1, \gamma \neq i}^{N_s} \eta(x_i^{(s)}) - \frac{1}{M_t} \sum_{j=1}^{M_t} \eta(x_j^{(t)}) \right\| \quad (28)$$

[0070] 式(28)中 $MMD_{\gamma \neq i}$ 表示缺少第i个源数据样本的最大均值距离,其中 $\gamma = 1, 2, \dots, N_s$

则

$$[0071] \quad \rho_i = \frac{(MMD_{\gamma \neq i} - MMD)}{MMD} \quad (29)$$

[0072] 式(28)中 $\rho_i$ 表示第 $i$ 个样本的最大均值差异贡献系数,若 $\rho_i > 0$ ,说明第 $i$ 个样本是对MMD有贡献的,相反若 $\rho_i \leq 0$ ,说明第 $i$ 个样本是对MMD有“负”贡献的;通过计算每个源数据的 $\rho_i$ ,迁移出 $\rho_i > 0$ 的样本数据,得到和目标数据分布更接近的源数据;

[0073] 四、冻结DBN-DNN的DBN层,获得基于迁移出的数据微调DBN-DNN中的DNN结构的模型,即TDBN-DNN迁移深度学习模型。

[0074] 进一步的技术方案在于,步骤4具体过程如下:

[0075] (1) 数据处理

[0076] 对输入数据负荷数据、电源出力数据、母线电压数据标准化处理,如式(32):

$$[0077] \quad \alpha'_{i,j} = \frac{\alpha_{i,j} - \alpha_{\min,j}}{\alpha_{\max,j} - \alpha_{\min,j}} \quad (32)$$

[0078] 式(32)中, $\alpha_{i,j}$ 代表第 $j$ 个数据特征的第 $i$ 个样本数据, $i=1,2,\dots,MC$ ,

[0079]  $j=1,2,\dots,2r$ , $\alpha'_{i,j}$ 代表第 $j$ 个数据特征的第 $i$ 个样本数据标准化后的数据, $\alpha_{\min,j}$ 代表第 $j$ 个数据特征在样本数据中的最小值, $\alpha_{\max,j}$ 代表第 $j$ 个数据特征在样本数据中的最大值;

[0080] (2) TDBN-DNN迁移深度学习模型模拟模型来模拟电网运行时的负荷数据、电源数据和母线电压数据与线损率之间的非线性映射关系

[0081] 根据样本的规模及实时性要求,确定TDBN-DNN迁移深度学习模型的隐藏层数量及其节点数量,并初始化网络参数;然后用标准化后的负荷数据、电源出力数据、母线电压数据样本数据为输入数据,对应断面线损率为标签数据,利用贪心无监督学习算法逐层对整个DBN深度置信网络模型逐层训练;最后将DBN的输出特征向量给DNN一个较好的初值,采用BP算法用对应断面线损率有监督地训练DNN来拟合标签数据;

[0082] (3) TDBN-DNN迁移深度学习模型预测线损率

[0083] 对任务目标输入数据进行标准化处理后,将此数据输入TDBN-DNN线损率预测模型,得到预测线损率值。

[0084] 本发明与现有技术相比有益效果如下:

[0085] 1、考虑因素全面,样本输入数据与潮流计算所用数据一致,模型更能有效拟合输入与输出的非线性映射关系;

[0086] 2、本文所提线损率预测模型不用经过线损计算中潮流计算的过程,而是直接用模型的非线性拟合能力直接计算结果,提高了运算速度。

[0087] 3、提出深度学习线损率预测模型,比浅层网络更能有效地模拟电网运行数据与线损率的复杂非线性映射关系。

[0088] 4、本文在对待预测任务前,通过迁移学习中的MMD法检验源训练数据与目标数据的分布距离,并用最大均值差异贡献系数(CCMD)来筛选源数据中与目标数据分布更接近的训练数据来微调DNN。这样解决了深度学习中需要大量样本数据,源、任务数据分布不一致,容易过拟合的问题,并能提高线损率预测精度。

## 附图说明

[0089] 图1是本发明方法的DBN基本组成单元RBM结构图；

[0090] 图2是本发明方法的DBN-DNN模型结构；

[0091] 图3是本发明方法的流程图。

[0092] 具体实施方式具体实施方式

[0093] 下面结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本发明的一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0094] 在下面的描述中阐述了很多具体细节以便于充分理解本发明，但是本发明还可以采用其他不同于在此描述的其它方式来实施，本领域技术人员可以在不违背本发明内涵的情况下做类似推广，因此本发明不受下面公开的具体实施例的限制。

[0095] 如图3所示，本发明所述方法的步骤如下：

[0096] 步骤1，建立DBN深度置信网络模型，与其CD算法训练，具体建模过程如下：

[0097] 一、深度置信网络

[0098] DBN是一种多隐层的生成性结构图模型，由若干层受限玻尔兹曼机 (restricted Boltzmann machine, RBM) 栈式叠加而成，拥有强大的特征提取能力。RBM结构图如图1所示。

[0099] RBM模型是一个能量模型，对于一组已知的状态  $(v, h)$ ，它的能量函数可定义为  $E_{\theta}(v, h)$ ，其中  $\theta$  为网络参数  $\theta = \{a_i, b_j, w_{ij}\}$ ，它的隐含层与可视层的联合概率分布可定义为  $p(v, h)$ ，即：

$$[0100] \quad E_{\theta}(v, h) = -\sum_{i=1}^n \sum_{j=1}^m w_{ij} v_i h_j - \sum_{i=1}^n a_i v_i - \sum_{j=1}^m b_j h_j \quad (1)$$

$$[0101] \quad P(v, h) = \frac{1}{Z_{\theta}} e^{-E_{\theta}(v, h)} \quad (2)$$

[0102] 式(1)中  $m$  为输入单元个数， $n$  为输出单元的个数， $a_i$  为每个输入单元实值偏置量， $b_j$  为每个隐单元的实值偏置， $v_i$  用来表示输入数据， $h_i$  为隐单元输出数据， $w_{ij}$  为实值权重矩阵；

[0103] 式(2)中  $Z_{\theta}$  为配分函数， $Z_{\theta} = \sum_{v, h} e^{-E_{\theta}(v, h)}$ ；由受限波尔兹曼机各层之间的条件独立性可知，当给定输入数据时，输出层节点取值满足如下条件概率：

$$[0104] \quad P(h_j = 1 | v, \theta) = \varphi(b_j + \sum_{i=1}^n w_{ij} v_i) \quad (3)$$

[0105] 式(3)中  $\varphi(o) = 1 + \frac{1}{\exp(-o)}$  为sigmoid激活函数；相应的，当输出层的数据确定后，输入节点的取值条件概率为：

$$[0106] \quad P(v_i = 1 | v, \theta) = \varphi(a_i + \sum_{j=1}^m w_{ij} h_j) \quad (4)$$

[0107] 给定一组训练样本  $G = \{g_1, g_2, \dots, g_s\}$ ，训练RBM意味着调整参数  $\theta$ ，以拟合给定的训练样本，训练RBM的目标就是通过最大化网络的似然函数  $L(\theta)$  得到参数  $\theta$ ，即

$L(\theta) = \prod_{i=1}^n p(g_i)$ 为简化计算,将其写为对数形式为:

$$[0108] \quad \ln L(\theta) = \sum_{i=1}^n \ln p(g_i) \quad (5)$$

$$[0109] \quad \theta = \arg \max L(\theta) = \arg \max \sum_{i=1}^n p(g_i) \quad (6)$$

[0110] 二、CD训练算法

[0111] 对比散度 (contrastive divergence, CD). 用对比散度算法对对数似然函中的参数 $\theta$ 求导,如下:

$$[0112] \quad \frac{\partial \ln p(v_i | \theta)}{\partial w_{ij}} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} \quad (7)$$

$$[0113] \quad \frac{\partial \ln p(v_i | \theta)}{\partial a_i} = \langle v_i \rangle_{data} - \langle v_i \rangle_{model} \quad (8)$$

$$[0114] \quad \frac{\partial \ln p(v_i | \theta)}{\partial b_j} = \langle h_j \rangle_{data} - \langle h_j \rangle_{model} \quad (9)$$

[0115] 各参数的更新准则为:

$$[0116] \quad \Delta w_{ij} = \epsilon_{CD} (\langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{recon}) \quad (10)$$

$$[0117] \quad \Delta a_i = \epsilon_{CD} (\langle v_i \rangle_{data} - \langle v_i \rangle_{recon}) \quad (11)$$

$$[0118] \quad \Delta b_j = \epsilon_{CD} (\langle h_j \rangle_{data} - \langle h_j \rangle_{recon}) \quad (12)$$

[0119] 式(11) — (12) 中,  $\epsilon_{CD}$ 为学习步长,  $\langle h_j \rangle_{recon}$ 表示一步重构后的模型定义分布。

[0120] 步骤2,在DBN的基础上建立DBN-DNN深度学习模型,如图2所示,其中DNN的模型构建具体过程如下:

[0121] 将DNN的输入层连接到DBN的输出层,建立DBN-DNN深度学习模型,其中DNN的训练过程包括前向传播和反向传播,内容如下:

[0122] 一、前向传播过程

$$[0123] \quad f(p) = \frac{1}{1 + e^{-p}} \quad (13)$$

$$[0124] \quad f(p) = p \quad (14)$$

$$[0125] \quad h_{w,b}(p) = \delta(wp + b) \quad (15)$$

$$[0126] \quad h_{w,b}(p_j)^{(l)} = 1 / \left( 1 + e^{-\left( \sum_i p_i^{l-1} \times w_{ij}^l + b_j^l \right)} \right) \quad (16)$$

$$[0127] \quad h_{w,b}(p_j)^{k_l} = \sum_i p_i^{k_l-1} * w_{ij}^{k_l} + b_j^{k_l} \quad (17)$$

[0128] 式(16)中:  $w_{ij}^l$ 代表第1层第j节点与第1-1层第i节点之间的联接权重参数;  $b_j^l$ 代表第1层第j节点的截距项;

[0129] 式(17)中  $h_{w,b}(p_j)^{k_l}$ 代表第 $k_l$ 层第j节点的输出值;

[0130] 二、反向传播过程

[0131] 对于第 $k_l$ 层(输出层)的输出单元 $i$ ,其残差计算公式如下:

$$[0132] \quad \delta_i^{(k_l)} = \frac{\partial}{\partial z_i^{k_l}} \frac{1}{2} P h_{w,b(p)} - t P^2 = -(t - h_{w,b(p)}^{k_l}) \cdot f'(z_i^{(k_l)}) \quad (18)$$

[0133] 对于 $l=k_{l-1}, k_{l-2}, k_{l-3}, \dots, 2$ 的各层,第 $l$ 层的第 $i$ 节点的残差计算式如下:

$$[0134] \quad \delta_i^{(l)} = - \left( \sum_{j=1}^{s_{l+1}} w_{ji}^{(l)} \delta_j^{(l+1)} \right) \cdot f'(z_i^{(l)}) \quad (19)$$

[0135] 对于一个固定训练样本集  $\{(u^1, y^1), \dots, (u^c, y^c)\}$ , 包含 $c$ 个样本, 利用小批量梯度下降法求解深层神经网络, 其损失函数为:

$$[0136] \quad J(w, b) = \left[ \frac{1}{c} \sum_{i=1}^c \frac{1}{2} p h_{w,b}(p) - t P^2 \right] + \frac{\lambda}{2} \sum_{l=1}^{k_l-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} (w_{ij}^{(l)})^2 \quad (20)$$

[0137] 其(20)中, 权重衰减参数 $\lambda$ 用于控制式(20)中两项的重要性; 因此, 小批量梯度下降法中每次迭代对于参数 $w$ 和 $b$ 的更新如下:

$$[0138] \quad w^{(l)} = w^{(l)} - \beta^{(l)} \left[ \left( \frac{1}{c} (\Delta w^{(l)} + \delta^{(l+1)} (h_{w,b}(p)^{(l)})^T) \right) + \lambda w^{(l)} \right] \quad (21)$$

$$[0139] \quad b^{(l)} = b^{(l)} - \beta^{(l)} \left[ \frac{1}{c} (\Delta b^{(l)} + \delta^{(l+1)}) \right] \quad (22)$$

[0140] 进一步的, 在步骤3中, 所述迁移深度学习模型, 其具体过程如下:

[0141] 用源数据训练DBN-DNN深度学习模型, 训练过程如步骤一、二所示。

[0142] 一、度量源数据与任务数据分布差异:

[0143] 假设分别存在一个满足 $d$ 分布的源数  $X^{(s)} = [x_1^{(s)}, x_2^{(s)}, \dots, x_N^{(s)}]$  和一个满足 $q$ 分布的目标数据  $X^{(t)} = [x_1^{(t)}, x_2^{(t)}, \dots, x_M^{(t)}]$ ;  $X^{(s)}$ ,  $X^{(t)}$  的最大均值差异 (maximum mean discrepancy, MMD) 可以表示为:

$$[0144] \quad MMD = \left\| \frac{1}{N_S} \sum_{i=1}^{N_S} \eta(x_i^{(s)}) - \frac{1}{M_t} \sum_{j=1}^{M_t} \eta(x_j^{(t)}) \right\| \quad (23)$$

[0145] 式(23)中 $H$ 表示再生核希尔伯特空间 (reproducing kernel Hilbert space, RKHS),  $\phi(\cdot): X \rightarrow H$  表示原始特征空间映射到RKHS的非线性特征映射核函数; 由于RKHS通常是高维, 甚至是无限空间, 对应的核一般选择表示无穷维的高斯核:

$$[0146] \quad k(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right) \quad (24)$$

[0147] 式(24)其中 $\sigma$ 函数的宽度参数; 为了计算方便, 我们通常用MMD的平方形式, 即 $MMD^2$ ; 将其展开为:

[0148]

$$\begin{aligned}
MMD^2 &= \frac{1}{M_t^2} \sum_{i=1}^{M_t} \sum_{j=1}^{M_t} \eta(x_i)^T \eta(x_j) + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} \eta(x_i)^T \eta(x_j) - \frac{1}{N_t M_t} \sum_{i=1}^{N_t} \sum_{j=1}^{M_t} \eta(x_i)^T \eta(x_j) \\
&= \frac{1}{M_t^2} \sum_{i=1}^{M_t} \sum_{j=1}^{M_t} \langle \eta(x_i), \eta(x_j) \rangle + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} \langle \eta(x_i), \eta(x_j) \rangle - \frac{1}{N_t M_t} \sum_{i=1}^{N_t} \sum_{j=1}^{M_t} \langle \eta(x_i), \eta(x_j) \rangle
\end{aligned} \quad (25)$$

[0149] 在此引入核技巧(kernel trick):

$$[0150] \quad \langle \eta(x), \eta(x') \rangle_H = \langle K(x, \cdot), K(\cdot, x') \rangle_H = K(x, x') \quad (26)$$

[0151] 式(25)简化为:

[0152]

$$MMD^2 = \frac{1}{M_t^2} \sum_{i=1}^{M_t} \sum_{j=1}^{M_t} K(x_i, x_j) + \frac{1}{N_t^2} \sum_{i=1}^{N_t} \sum_{j=1}^{N_t} K(x_i, x_j) - \frac{1}{N_t M_t} \sum_{i=1}^{N_t} \sum_{j=1}^{M_t} K(x_i, x_j) \quad (27)$$

[0153] 二、最大均值差异贡献系数迁移源数据

[0154] 为了找出源数据和目标数据不相关或相关性不大的数据,在此,本文通过定义每个源数据的最大均值差异贡献系数(contributing coefficient of maximum mean discrepancy, CCMMD)来筛选源数据;

[0155] 用 $\rho_i$ 表示第 $i$ 个样本的最大均值差异贡献系数;假设缺少第 $i$ 个源数据样本的 $MMD_{\gamma \neq i}$ 为:

$$[0156] \quad MMD_{\gamma \neq i} = \left\| \frac{1}{N_s} \sum_{i=1, \gamma \neq i}^{N_s} \eta(x_i^{(s)}) - \frac{1}{M_t} \sum_{j=1}^{M_t} \eta(x_j^{(t)}) \right\| \quad (28)$$

[0157] 式(28中) $MMD_{\gamma \neq i}$ 表示缺少第 $i$ 个源数据样本的最大均值距离,其中 $\gamma = 1, 2, \dots, N_s$ 则

$$[0158] \quad \rho_i = \frac{(MMD_{\gamma \neq i} - MMD)}{MMD} \quad (29)$$

[0159] 式(29)中 $\rho_i$ 表示第 $i$ 个样本的最大均值差异贡献系数,若 $\rho_i > 0$ ,说明第 $i$ 个样本是对MMD有贡献的,相反若 $\rho_i \leq 0$ ,说明第 $i$ 个样本是对MMD有“负”贡献的;通过计算每个源数据的 $\rho_i$ ,迁移出 $\rho_i > 0$ 的样本数据,得到和目标数据分布更接近的源数据。

[0160] 四、冻结DBN-DNN的DBN层,由迁移出的源数据用于微调高层的DNN神经网络,得到TDBN-DNN深度迁移学习模型。

[0161] 进一步的,在步骤4中,所述基于TDBN-DNN的理论线损预测模型的建立,其具体过程如下:

[0162] 一、理论线损问题描述

[0163] 利用深度学习模型来模拟电网运行时的负荷数据、电源数据和母线电压数据与线损率之间的非线性映射关系。考虑深度学习模型的强特征提取能力以及强拟合表现能力,将电网络等值模型中各节点有功功率数据、无功功率数据和母线电压数据作为深度模型的输入矩阵 $X$ ,将相应网络的理论线损率作为输出矩阵 $T$ .假设电网有 $r$ 个节点,其中 $e$ 个PQ节点, $r-e-1$ 个PV节点,一个平衡节点,深度学习模型的输入、输出矩阵如式(30)、(31);

$$[0164] \quad X = \begin{bmatrix} P_{1,1} & P_{1,2} & \cdots & P_{1,MC} \\ Q_{1,1} & Q_{1,2} & \cdots & Q_{1,MC} \\ \vdots & \vdots & \vdots & \vdots \\ P_{e,1} & P_{e,2} & \cdots & P_{e,MC} \\ Q_{e,1} & Q_{e,2} & \cdots & Q_{e,MC} \\ P_{e+1,1} & P_{e+1,2} & \cdots & P_{e+1,MC} \\ V_{e+1,1} & V_{e+1,2} & \cdots & V_{e+1,MC} \\ \vdots & \vdots & \vdots & \vdots \\ P_{r-1,1} & P_{r-1,2} & \cdots & P_{r-1,MC} \\ V_{r-1,1} & V_{r-1,2} & \cdots & V_{r-1,MC} \\ \theta_{r,1} & \theta_{r,2} & \cdots & \theta_{r,MC} \\ V_{r,1} & V_{r,2} & \cdots & V_{r,MC} \end{bmatrix} \quad (30)$$

$$[0165] \quad T = [\Delta \xi_1, \Delta \xi_2, \cdots, \Delta \xi_{MC}] \quad (31)$$

[0166] 式(30)中,其中MC为断面数量,r为电网中节点数量,e为负荷PQ节点数量;

[0167] 式(31)中, $\Delta \xi_i$ 为第i个样本的线损率标签数据;

[0168] 二、TDBN-DNN模型构建与求解

[0169] 具体步骤如下:

[0170] (1) 数据处理。因负荷数据、电源出力数据、母线电压数据取值范围和单位不同,为了不受量纲的影响,防止特征之间的绝对值不同而出现数据“大吃小”的情况,对输入数据标准化处理,如式(32):

$$[0171] \quad \alpha'_{i,j} = \frac{\alpha_{i,j} - \alpha_{\min,j}}{\alpha_{\max,j} - \alpha_{\min,j}} \quad (32)$$

[0172] 式(32)中, $\alpha_{i,j}$ 代表第j个数据特征的第i个样本数据, $i=1,2,\cdots,M$ ,

[0173]  $j=1,2,\cdots,2n$ , $\alpha'_{i,j}$ 代表第j个数据特征的第i个样本数据标准化后的数据, $\alpha_{\min,j}$ 代表第j个数据特征在样本数据中的最小值, $\alpha_{\max,j}$ 代表第j个数据特征在样本数据中的最大值。

[0174] (2) 预训练深度学习模型。首先根据样本的规模及实时性要求,确定深度学习网络的隐藏层数量及其节点数量,并初始化网络参数。然后用标准化后的电网电源、负荷、电压等样本数据为输入数据,对应断面线损率为标签数据,利用贪心无监督学习算法逐层对整个DBN深度置信网络模型逐层训练。最后将DBN的输出特征向量给DNN一个较好的初值,采用BP算法用对应断面线损率有监督地训练DNN来拟合标签数据。

[0175] (3) 迁移深度学习模型。首先,将通常认为深度学习模型中的底层通用网络冻结。然后,采用最大均值差异法(MMD)度量源训练数据与目标数据的分布差异,并计算源数据中每个样本的最大均值差异贡献系数(CCMD),并选出 $\rho_i > 0$ 的样本数据。最后用选出的源样本数据微调预训练好的DNN,得到深度迁移学习模型TDBN-DNN。

[0176] (4) 线损率预测。对任务目标输入数据进行标准化处理后,将此数据输入TDBN-DNN线损率预测模型,得到预测线损率值。

[0177] 以上所述的实施例仅仅是对本发明的优选实施方式进行描述,并非对本发明的范围进行限定,在不脱离本发明设计精神的前提下,本领域普通技术人员对本发明的技术方案做出的各种变形和改进,均应落入本发明权利要求书确定的保护范围内。

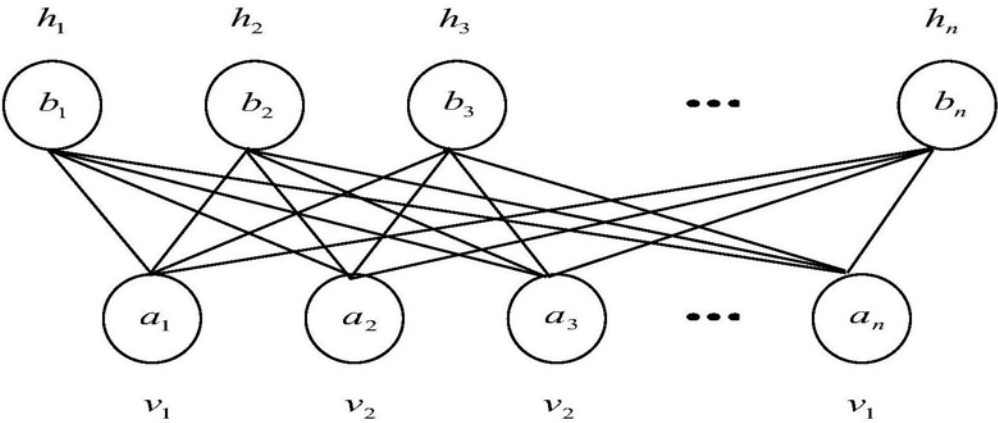


图1

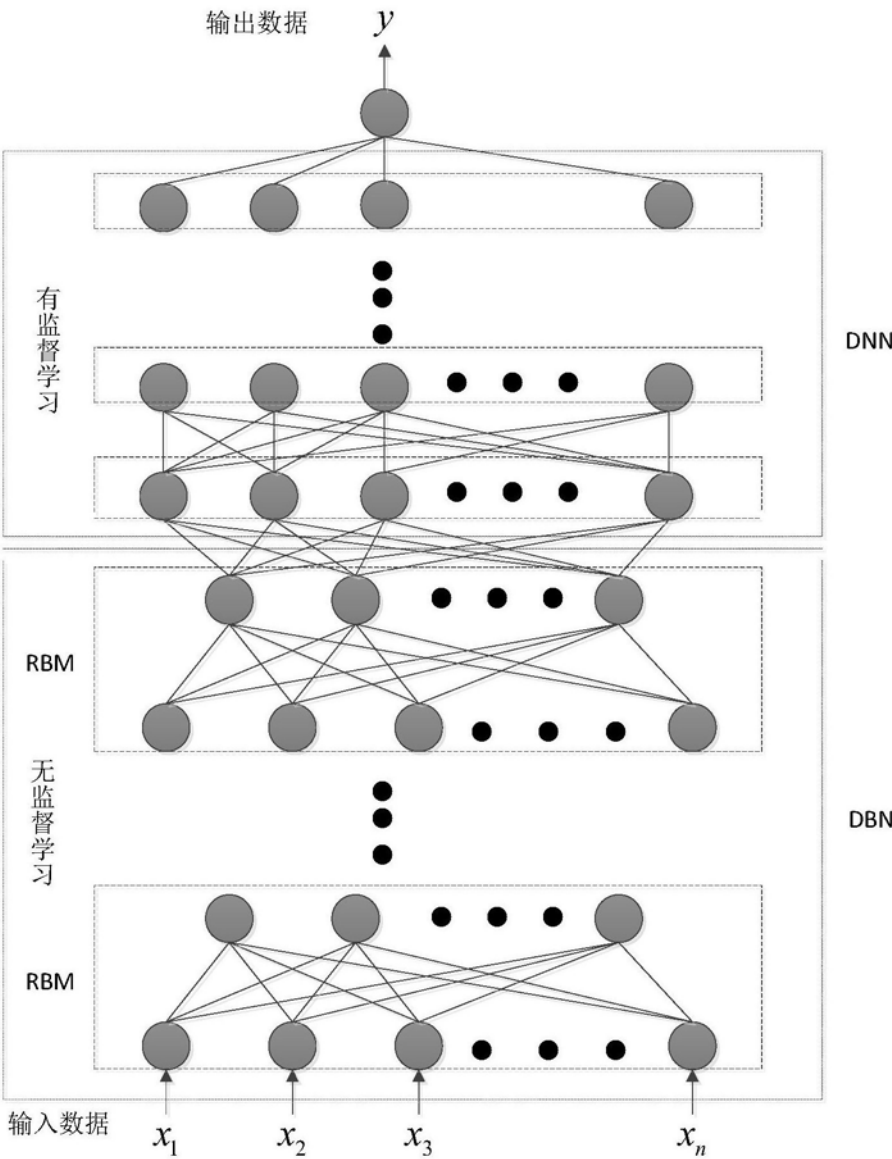


图2



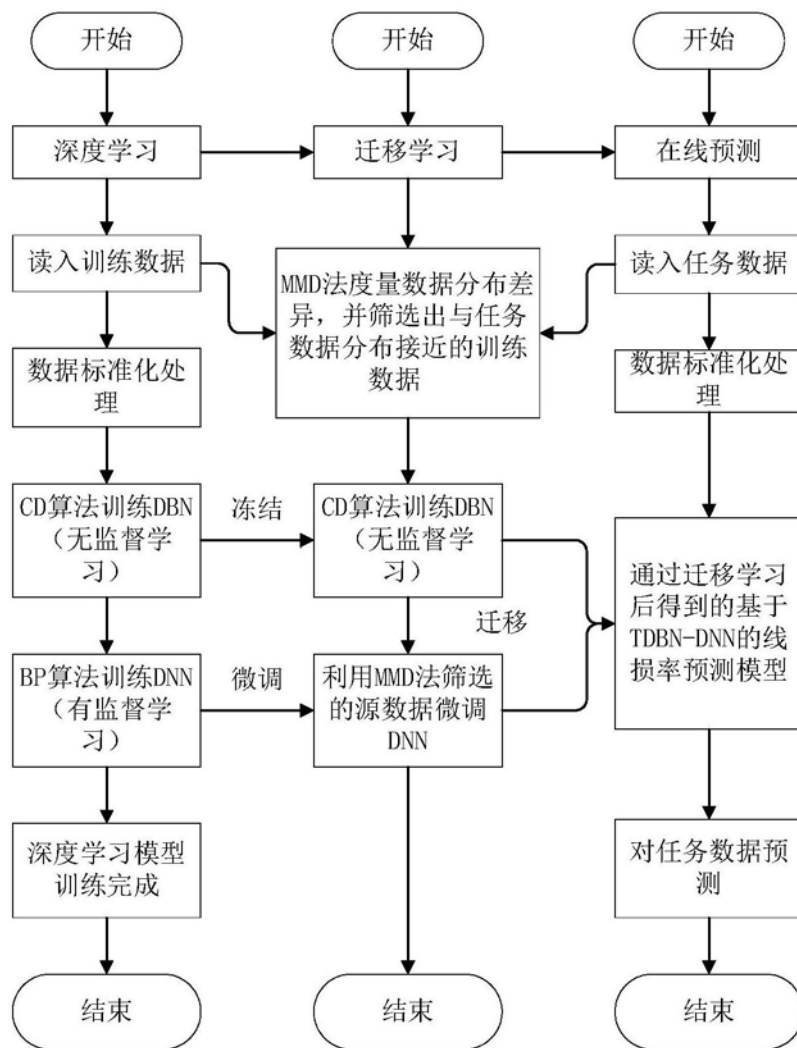


图3