

基于 SFLA 优化 BP 神经网络的语音信号分类

叶红卫, 戴光智

(1. 河源职业技术学院 电子与信息工程学院, 广东 河源 517000;

2. 深圳职业技术学院 计算机工程学院, 广东 深圳 518055)

摘要: BP 神经网络是一种多层前馈网络, 数据经过网络的输入层、隐含层逐层处理后, 由输出层进行输出, 通过和期望输出的对比进行反向传播, 调整网络参数使输出不断逼近期望输出; 在使用 BP 神经网络对语音特征信号进行分类的过程中, 会出现 BP 神经网络易陷入局部最优解、学习收敛速度慢的问题; 针对此问题提出一种基于 SFLA 优化 BP 神经网络权值和阈值的方法, 引入 SFLA 算法优化网络权值和阈值, 利用 SFLA 优化后的 BP 网络模型进行语音特征信号分类; 仿真结果表明, 经 SFLA 优化后的 BP 神经网络与未优化的神经网络相比, 不仅训练速度快, 而且误差小, 语音特征信号分类的正确率平均提高 1.31%。

关键词: BP 神经网络; SFLA; 优化; 分类

Classification of Speech Based on BP Neural Network Optimized by SFLA

Ye Hongwei, Dai Guangzhi

(1. School of Electronic and Information Engineering, Heyuan Polytechnic, Heyuan 517000, China

2. School of Computer engineering, Shenzhen Polytechnic, Shenzhen 518055, China)

Abstract: A back-propagation (BP) neural network consists of an input layer, one or more hidden layers and an output layer. An input vector is presented to the network, it is propagated forward through the network, layer by layer, until it reaches the output layer. The output of the network is then compared to the desired output, using a loss function, The error values are then propagated backwards, starting from the output, until each neuron has an associated error value which roughly represents its contribution to the original output. The BP neural network easily falls into a local extreme values and the slow convergence, during the Classification of Speech using it. A new method is put forward to optimize weights and threshold of BP neural network using SFLA. The new model was used in the classification of four typical speech, results of which were analysed and compared with that BP neural network. BP neural network based on SFLA has both fast training speed and small number of errors, produced average increase of 1.31% in the accuracy.

Keywords: BP neural network; SFLA; optimization; Classification

0 引言

BP 神经网络在工程领域有着广泛的应用, BP 神经网络也存在自身的缺点, 在训练网络时学习收敛速度慢。同时, 易陷入局部的最优解而非全局最优解。为了改进 BP 神经网络的缺点, 提出了许多优化 BP 网络的方法, 利用遗传算法优化^[1]、基于粒子群算法优化^[2]、利用萤火虫算法优化^[3]、克隆选择粒子群优化^[4]。混合蛙跳算法 (SFLA) 广泛应用在优化问题的求解, SFLA 算法结合了元算法和粒子群优化算法的优点^[5], SFLA 具有参数少, 运算速度快、寻优能力强的特点, 对 BP 神经网络的各层之间的初始权值与阈值, 采用 SFLA 算法进行优化。利用得到的最优值去训练网络, 应用于语音特征信号分类。

1 混合蛙跳算法

1.1 算法简介

SFLA 是以群体协同合作搜索的模因算法为基础, 通过启

发式算法搜寻全局最优解。基于模因之间或内部可进行经验、知识和信息的共享和交流^[6]。在 SFLA 中, 种群由许多蛙构成, 每只蛙表示待解决问题的解向量。种群被划分成多个子群, 由一定数量的蛙组成一个子群, 称为一个模因 (memplex)。蛙群由不同的模因构成, 每个模因中的蛙都有自己的信息, 同时受到同一模因中蛙的影响, 彼此之间进行信息交流。通过 Memetic 算法来进化发展。经过一定次数的交流和跳跃后, 不同的交流信息在所有模因组中进行传播。然后根据给定的优化条件, 继续进行局部迭代搜索和跳跃, 直到达到迭代次数或满足优化目标。

SFLA 的流程如图 1 所示。首先, 随机生产初始蛙群, 对于一个 S 维解的优化问题, 每个蛙表示为 $X_i = (x_{i1}, x_{i2}, \dots, x_{iS})$, 根据自由度对蛙群进行降序排序, 然后, 整个蛙群划分为 m 个模因组, 每个模因组分配 n 只蛙。例如 $m=3$, 则分配原则为: 第 1 只蛙划分到模因组 1, 第 2 只蛙划分到模因组 2, 第 3 只蛙划分到模因组 3, 第 4 只蛙划分到模因组 1, 第 m 只蛙划分到模因组 m , 第 $m+1$ 只蛙划分到模因组 1, 以此类推^[7-8]。在模因组中进行设定次数的局部搜索, 在每一个模因组中具有最好适应度的蛙和最差适应度的蛙分别记为 X_b 和 X_w , 具有全局最优适应度的蛙记为 X_g 。在每一轮的循环中, 对具有最差适应度的蛙进行演化运算, 对其位置按式 (1) 和式 (2) 进行更新。

$$(D_i) = rand() \cdot (X_b - X_w) \quad (1)$$

收稿日期: 2017-02-20; 修回日期: 2017-03-13。

基金项目: 国家自然科学基金 (61671309)。

作者简介: 叶红卫 (1979-), 男, 广东省河源市人, 讲师, 硕士, 主要从事智能算法、机器学习方向的研究。

戴光智 (1974-), 男, 内蒙古人, 副教授, 博士, 主要从事计算机视觉方向的研究。

$$X_{\omega}^2 = X_{\omega}^1 + D_i \quad (-D_{\max} \leq D_i \leq D_{\max}) \quad (2)$$

其中为 $[0, 1]$ 的随机数, 为蛙的目前所处的位置, 为更新后的位置, 为更新的最大步长, 在此演化过程中, 用得到的更好的蛙去替代最差的蛙, 否则, 若更新后没有得到更好的蛙, 则随机产生一只蛙替代 X_{ω} , 在模因组中重新进行迭代, 达到局部搜索迭代次数为止。接着对全部模因组中的蛙重新混杂, 并按适应度进行排序。更新种群最蛙的位置, 然后重新划分蛙群, 进行局部深度搜索, 次数达到全局循环次数或满足终止的优化目标^[9]。

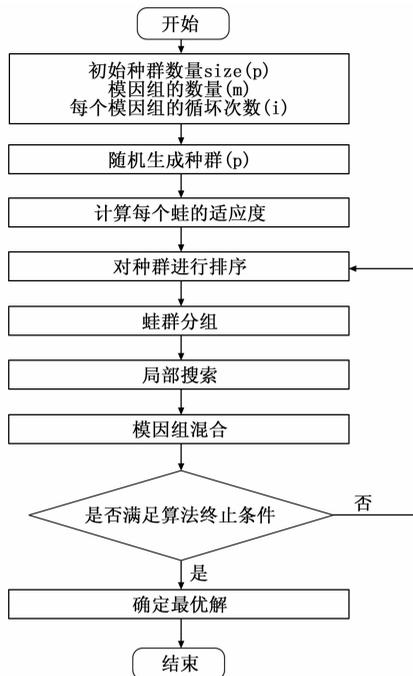


图 1 SFLA 算法流程

1.2 算法实现

SFLA 算法搜寻最优解通过启发式算法, 实现 SFLA 算法首先初始化蛙群的各个参数。蛙群模因组个数为 m , 每个模因组中蛙的个数为 n , 种群中蛙的个数 $F = m * n$; 每只蛙包含解空间维数为 S 。全局迭代最大次数 $LoopMax$, 局部迭代最大次数 (每个模因组的最大迭代次数) $Nmax$, 初始种群范围上下限分别为 $Pmax$ 、 $Pmin$ 。模因组内青蛙位置的最大变化步长为 $Smax$, 适应度 $fitness$ 的计算由自定义的 fun 函数决定。全局最优蛙为 $Frog_GlobalBest$, 局部最优蛙为 $Frog_LocalBest$, 局部最差蛙为 $Frog_localWorse$, SFLA 算法 MATLAB 实现的伪代码如下:

```

Initialize:m,n,F,S,LoopMax,.....
For LoopNum=1:LoopMax
Compute fitness
For im=1:m
For i=1:Nmax
Determine the Frog_LocalBest and Frog_localWorse
Improve the Frog_localWorse using equations (1) or (2)
End
End
Shuffle the evolved memplexes
Sort(fitness)
    
```

```

Save the Frog_GlobalBest
Check if termination=true
End
    
```

2 BP 神经网络

BP 神经网络是一种多层前馈网络, BP 神经网络结构如图 2 所示, 由输入层、隐含层、输出层组成。网络的节点数包含了 i 个输入节点, j 个隐含节点和 k 个输出节点, 所有的输入节点通过权值 ω_{ji} 和隐含节点相连, 隐含节点通过权值 ω_{kj} 相连。通过有监督的训练, 参数的调整在 BP 网络中分为两个阶段, 第一阶段, 在 BP 网络中通过输出层得到输入层的预测值, 通过预测值和实际值的比较, 调整各层之间的连接权值和参数, 得到新的参数值。同时, 进行不断的更新学习, 若满足设定的训练目标, 则停止网络的参数调整^[10]。

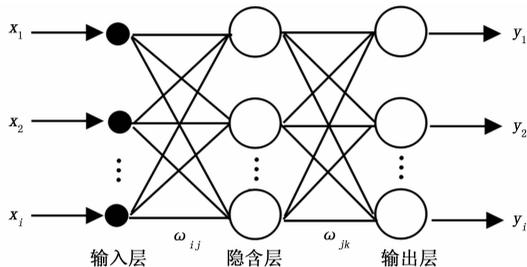


图 2 BP 网络结构

BP 神经网络训练的具体步骤如下:

步骤 1: BP 网络初始化, 确定 BP 神经网络层次结构, 输入层节点个数 n , 隐含层节点个数 p , 输出层节点个数 q 。初始化网络输入层和隐含层、隐含层和输出层神经元之间的连接权值 ω_{ij} 和 ω_{jk} , 隐含层和输出层的阈值分别为 h 和 s 。同时, 确定神经元的激励函数和学习速率。

步骤 2: 计算隐含层输出, 隐含层输出 H_j 。

$$H_j = f\left(\sum_{i=1}^n \omega_{ij} x_i - h_j\right) \quad j = 1, 2, \dots, p$$

其中, p 为隐含层节点个数, f 为隐含层的激励函数, x 为输入向量。

步骤 3: 计算输出层输出。 $O_k = \sum_j H_j \omega_{jk} - s_k, k = 1, 2, \dots, q$

步骤 4: 计算预测误差。 $e_k = T_k - O_k \quad k = 1, 2, \dots, q$
 T_k 为期望输出向量。

步骤 5: 更新权值。 $\omega_{kj} = \omega_{kj} + \eta H_j (1 - H_j) x(i) \sum_{k=1}^p \omega_{jk} e_k$

$$\omega_{jk} = \omega_{jk} + \eta H_j e_k$$

步骤 6: 更新阈值。 $h_j = h_j + \eta H_j (1 - H_j) x(i) \sum_{k=1}^p \omega_{jk} e_k$

$$s_k = s_k + e_k \quad k = 1, 2, \dots, q$$

步骤 7: 训练次数和设定的次数进行比较, 若没有达到。返回步骤 2。若达到, 则 BP 神经网络训练结束。

3 实验仿真

3.1 数据处理

语音特征信号分类过程为: 对获取到语音信号进行预处理后, 采用合适的算法把语音特征信号从中提取出来。该语音特征信号可看成语音的一个模式, 再通过对已有参考模式的比

对。获得最佳匹配的参考模式, 从而得到该语音的分类结果。本实验选用了古筝、民歌、摇滚和流行四类不同音乐, 每段音乐采用倒谱系数法提取 500 组 24 维语音特征信号^[11], 部分语音特征数据如表 1 所示。

表 1 部分语音特征数据

类别	特征 1	特征 2	特征 3	……	特征 24
1	-14.82713171	-3.001085509	1.520907785	……	0.213976743
1	-16.22886124	-2.801874337	-0.4108157	……	0.320707761
1	-15.1242838	-2.59871264	-0.359965674	……	0.596002471
1	-15.70314246	-2.530056621	0.255003254	……	0.491487215
2	-7.124675028	-3.291221296	-1.989485515	……	-0.190324063
2	-7.491168268	-3.876559747	-6.284460902	……	-0.164035009
2	-7.008073253	-3.046781147	-4.248054958	……	0.193836521
2	-6.470520074	-2.935980946	-3.135171374	……	0.114851434
3	-14.54861636	-2.546153699	-1.364847328	……	-0.074555423
3	-15.05664211	-1.842875047	-1.12674255	……	-0.20296232
3	-15.00872378	-0.756188396	1.334223717	……	-0.33626362
3	-12.49013913	-3.718298992	-2.615074106	……	0.22889579
4	-1.217017482	1.080606606	1.841872375	……	0.653276583
4	-3.388706732	3.036554219	0.355659097	……	0.547523167
4	-6.15672141	1.608502459	0.868304617	……	0.327454193
4	-7.527673153	0.994359732	4.094927525	……	0.248972345

在训练 BP 神经网络之前, 为了减少数据对网络预测误差的影响, 将各维的数据进行数据归一化处理, 将各维数据转换到在 [0, 1] 的数。归一化样本数据计算由式 (3) 确定, X_n 为归一化后的数据, x_{\min} 为语音类别序列最小值, x_{\max} 为语音类别序列最大值。

$$X_n = \frac{x_n - x_{\min}}{x_{\max} - x_{\min}} \quad (3)$$

3.2 算法流程

实验流程如图 3 所示, 首先确定网络结构、网络输入参数和各层的节点数, 然后初始化网络的各层之间的连接权值和阈值。通过 SFLA 进行优化, 获得优化后网络进行预测。

网络的预测精度与网络隐含层节点数的选取有关, 节点数过大, 训练时间久, 易出现过拟合; 节点数过小, 训练时间短, 但网络训练过早结束, 精度较低。最佳网络隐含层节点数 k 由式 (4) 确定。其中, m 为网络输入层节点数, n 为网络输出层节点数, l 为 0~10 的常数。取最佳网络隐含层节点数 $k=9$ 。

$$k < \sqrt{m+n} + l \quad (4)$$

使用 24-9-4 的 BP 神经网络结构, 24 个输入层节点, 9 个隐含层节点, 4 个输出层节点。网络训练次数为 20, 网络训练误差值为 0.000 1, 网络学习速率为 0.1。共有 2 000 组样本语音数据, 前 1 500 组数据为训练数据, 后 500 组为测试数据。蛙群的适应度 F 由如式 (5) 确定。

$$F = \frac{1}{\sum_{i=1}^q |y_i - o_i|} \quad (5)$$

其中, q 为 BP 网络输出层的输出节点, y_i 为第 i 个测试数据, o_i 为 BP 网络的第 i 个预测数据。适应度 F 越高, 得到的 BP 网络参数越优。

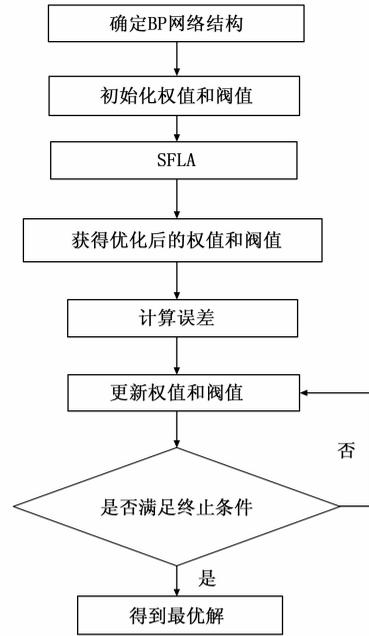


图 3 SFLA 优化 BP 网络流程图

3.3 实验结果与分析

实验硬件环境: CPU Intel (R) Core (TM) i5 - 6500 3.20 GHz, 内存: 16 GB, 软件环境: Win7 x64 操作系统, 算法实现软件采用 MATLAB R2012b。SFLA 算法的初始参数: $m=5, n=5, Loop_{\max}=5, N_{\max}=10, S_{\max}=0.01$ 。通过对算法的编程和 MATLAB 的仿真计算, 得到了实验的仿真结果。BP 神经网络训练的均方误差如图 4 所示, SFLA 优化后的 BP 神经网络均方误差如图 5 所示。未优化的网络在第 14 次训练时均方误差达到最小值 0.0494 78。优化后的网络在第 12 次训练时均方误差达到最小值 0.0334 84, 误差比未优化的网络优。通过实验的数据分析与对比, SFLA-BP 在训练次数和均方误差, 都比未优化的 BP 神经网络更优。

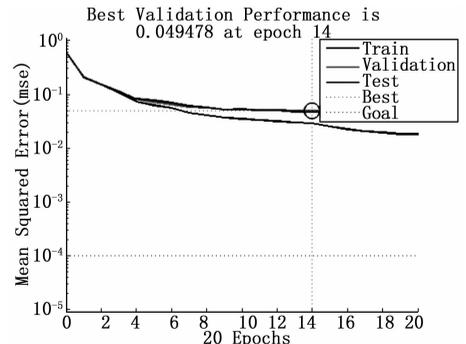


图 4 BP 神经网络训练均方误差

为了测试优化后网络的稳定性, 对样本数据进行混杂处理, 进行 100 次的优化循环测试。最终的分误差如图 6 所示, 语音类别分类正确率如表 2 所示。第一类语音的分类正确率提高 0.99%, 第二类几乎相同, 第三类正确率提高 2.35%, 第四类正确率提高 1.94%, 语音特征信号分类的正确率平均提高 1.31%。实验结果表明, 优化后的 BP 神经网络在语音

(下转第 231 页)