

文章编号:1001-9081(2006)08-1940-03

基于 BP 网络的权值更新快速收敛算法

周昌能,余雪丽

(太原理工大学 计算机学院,山西 太原 030024)

(zcn@263.net)

摘要:针对标准 BP 网络学习算法收敛慢的问题,提出了两种权值更新的快速收敛算法,即基于梯度变化率的快速传递算法和基于梯度方向的弹性传递算法,并在煤矿事故救援游戏式训练系统中进行仿真和比较,让游戏角色根据井下空气成分学习判断危险程度,以便受训人员或仿生机器人采取相应的措施。仿真结果表明,所提算法的收敛时间比标准算法有一定改善。

关键词:快速收敛算法;游戏式训练;BP 人工神经网络

中图分类号: TP183 **文献标识码:**A

Rapid convergence algorithms for weight values updating based on BP network

ZHOU Chang-neng, YU Xue-li

(College of Computer Science and Software Engineering, Taiyuan University of Technology, Taiyuan Shanxi 030024, China)

Abstract: To solve the slow convergence of standard learning algorithm in BP network, two rapid convergence algorithms were suggested for weight values updating. One is rapid transmission algorithm based on gradient change rate. The other is flexible transmission algorithm based on gradient orientation. The two algorithms were simulated and compared in Game Style Training System for Mine Accident Rescuing. Here the algorithms would help game roles learn to estimate the danger degree according to ingredients of mine air, and then help trainees or biorobots take corresponding actions. The simulating results show that shorter convergence time is taken for the two algorithms than the standard algorithm.

Key words: quick convergence algorithm; game style training; BP artificial neural network

BP 网络是在感知机基础上提出的一种多层人工神经网络模型,它能够学习大量的模式映射关系,却无需深奥的数学函数知识来描述复杂的输入—输出模式间的映射。假如输入层单元数为 I,输出层单元数为 O,则 BP 网络能实现从 I 维欧氏空间到 O 维欧氏空间的线性或非线性映射,应用领域非常广泛。但 BP 网络收敛速度慢,且容易陷入局部极小,影响了它的学习和应用效果。本文对标准 BP 网络学习算法进行了改进,给出了两种权值更新的快速收敛算法,并进行了仿真。

1 标准 BP 网络学习算法简介

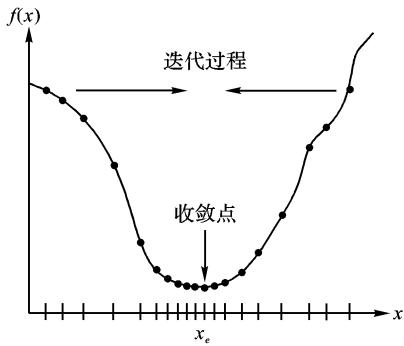


图 1 梯度最速下降法迭代过程

BP 网络学习算法实质是梯度最速下降法的一种应用。梯度最速下降法利用函数的斜率信息搜索答案 x_e ,而 x_e 对应的函数值 $f(x_e)$ 为极小值。从根本上说,梯度最速下降法是一

个反复迭代的过程,对 x_e 的逼近通过逐渐增减 x 的值来校正。从初始估计值 x_0 开始,函数在某一特定点 x_i ($i = 0, 1, \dots, n$, \dots) 处的斜率(表示为 $\nabla f(x_i)$) 被用来在合适的方向校正估计值,从而得到 x_{i+1} ,即 $x_{i+1} = x_i - \eta \nabla f(x_i)$,保证了 x_i 朝着 $f(x_i)$ 下降的方向移动,如图 1 所示。

从图 1 中可以看到,斜率绝对值越大的区域, x_i 更新越快,即更新步幅越大,相反,斜率绝对值越小的区域, x_i 更新越慢,即更新步幅越小。 η 称为学习速率,用于度量每一次迭代中所采用的步长。

BP 算法的目的是找到每一个连接权值的最佳值以最大限度地减少输出错误。对于最后一层,输出误差可以对照理想输出立即得到,对于隐含层,由于没有理想输出作为参考,不能直接得到误差,但是能够找到与输出中发生错误的神经元相连接的隐含神经元,通过在神经网络中反向传递误差,可以将误差分布到前面的神经元上。反向传递的过程可以看作错误梯度的递归定义过程,不管有多少层,都可以从输出开始反向处理,而隐含单元的错误梯度就是输出单元错误梯度的加权和。为便于讨论,取激发函数为 Sigmoid 函数,如式(1)所示:

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

下面给出算法实现过程,其推导过程可参考文献[1,2]:

(1) 对于每一个训练样本输入 s :

a) 计算输出节点的实际输出 O_s ;

收稿日期:2006-02-13;修订日期:2006-04-28

基金项目:山西省自然科学基金资助项目(20041043);山西省留学回国人员科研资助项目(200336)

作者简介:周昌能(1971-),男,云南昆明人,博士研究生,主要研究方向:人工智能、Game AI、网格应用; 余雪丽(1942-),女,北京人,教授,博士生导师,主要研究方向:人工智能、网格应用、智能决策支持。

b) 计算输出节点实际输出与理想输出(d_z)的误差梯度:

$$\delta_z = O_z(1 - O_z)(d_z - O_z) \quad (2)$$

c) 递归计算其他节点的误差梯度:

$$\delta_j = O_j(1 - O_j) \sum_k W_{jk} \delta_k \quad (3)$$

这里 W_{jk} 为节点 j 到后续节点 k 的连接权值, O_k 为节点 k 根据激活函数计算得到的输出。

d) 计算权值更新:

$$\Delta W_{ij} = \eta O_i \delta_j \quad (4)$$

这里 η 为学习比率, 需要通过实验确定, O_i 为节点 i 的输出, O_j 为节点 j 的输出, δ_j 为节点 j 的递归误差梯度。

(2) 对所有训练样本输入得到的同一权值的更新即(4)式求和, 并根据权值更新计算新的权值。

(3) 重复以上两步直到输出误差(用方差表示)足够小, 即:

$$E = \sum_s (\sum_z (d_{sz} - O_{sz}))^2 \leq \varepsilon \quad (5)$$

这里 s 为训练样本序列, z 为输出节点序列, d_{sz} 为样本 s 在节点 z 的理想输出, O_{sz} 为样本 s 在节点 z 的实际输出。

2 基于梯度变化率的快速传递算法

标准BP算法的权值更新基于普遍Delta规则, 我们根据二阶导数的知识, 用梯度变化率的倒数作为倍乘因子, 由上一训练周期的历史权值更新计算本次训练的权值更新, 可得到一种快速传递算法, 即:

$$\Delta W_{ij}(t) = \Delta W_{ij}(t-1) \cdot \frac{\nabla E_{ij}(t)}{\nabla E_{ij}(t-1) - \nabla E_{ij}(t)} \quad (6)$$

这里 t 代表当前训练周期, $\nabla E_{ij}(t)$ 是全部 s 个训练样本的误差梯度:

$$\nabla E_{ij}(t) = \frac{\partial E}{\partial W_{ij}} = -\delta_j O_i \quad (7)$$

这是一个批处理算法, 一个训练周期的每一个训练样本的所有梯度要累加到一起, 并在下一个训练周期开始时复位。这样得到的权值更新包含了对梯度变化的测量, 实践证明在很多问题上比标准的反向传递效率更高。

快速传递算法不再完全依靠当前周期的误差梯度和输出计算权值更新, 而是根据梯度变化率和上一周期的权值更新计算本周期的权值更新, 使得权值更新朝着更有利于收敛的方向进行。

3 基于梯度方向的弹性传递算法

计算权值更新时, 不仅要考虑当前周期梯度的符号, 还要考虑连续两个周期梯度符号的变化, 依此调整权值的更新方向和更新幅度。具体而言, 当梯度上升时, 权值向下调整, 梯度下降时, 权值向上调整; 如果梯度保持同一个方向, 就增大权值调整幅度, 如果梯度改变了方向, 就减小权值幅度。以上规律可用两组规则表示, 第一组规则用于确定权值调整步幅, 第二组规则用于确定权值调整方向。梯度的计算仍采用(7)式。

第一组规则描述如下:

(1) 如果梯度保持同一个方向, 就增大权值调整幅度。即如果 $\nabla E_{ij}(t-1) \cdot \nabla E_{ij}(t) > 0$, 则本周期步幅:

$$\Delta_{ij}(t) = \eta^+ \Delta_{ij}(t-1) \quad (8)$$

这里 $\nabla E(t)$ 为本周期梯度, $\nabla E_{ij}(t-1)$ 为上周期梯度, $\Delta_{ij}(t)$ 为本周期权值调整步幅, $\Delta_{ij}(t-1)$ 为上周期权值调整步幅, $\eta^+ > 1$ 。

(2) 如果梯度改变了方向, 就减小权值调整幅度。即如果 $\nabla E_{ij}(t-1) \cdot \nabla E_{ij}(t) < 0$, 则本周期步幅:

$$\Delta_{ij}(t) = \eta^- \Delta_{ij}(t-1) \quad 0 < \eta^- < 1 \quad (9)$$

(3) 如果梯度为 0, 则保持权值调整幅度不变。即如果 $\nabla E_{ij}(t-1) \cdot \nabla E_{ij}(t) = 0$, 则本周期步幅:

$$\Delta_{ij}(t) = \Delta_{ij}(t-1) \quad (10)$$

第二组规则描述如下:

(1) 当梯度上升时, 权值向下调整。即如果 $\nabla E_{ij}(t) > 0$, 则本周期权值更新:

$$\Delta W_{ij}(t) = -\Delta_{ij}(t) \quad (11)$$

(2) 当梯度下降时, 权值向上调整。即如果 $\nabla E_{ij}(t) < 0$, 则本周期权值更新:

$$\Delta W_{ij}(t) = +\Delta_{ij}(t) \quad (12)$$

(3) 当梯度为 0 时, 权值不作调整, 此时算法已经收敛于极小值。即如果 $\nabla E_{ij}(t) = 0$, 则本周期权值更新:

$$\Delta W_{ij}(t) = 0 \quad (13)$$

弹性传递算法不再完全依靠当前周期的误差梯度和输出计算权值更新, 而是根据梯度方向和梯度变化方向, 由上一周期的权值调整步幅计算本周期的权值更新, 使得权值更新朝着快速收敛的方向进行。

4 仿真

在煤矿事故救援游戏式训练系统中使用弹性传递算法, 根据井下空气成分计算危险程度, 以便受训人员或仿生机器人采取相应的措施。

根据煤矿安全规程的规定, 井下空气成分必须符合下列要求:

(1) 采掘工作面进风流中, 氧气浓度不低于 20%, 二氧化碳浓度不超过 0.5%, 瓦斯浓度不超过 1.0%, 氢气浓度不超过 0.5%。

(2) 有害气体浓度不超过表 1 规定值。

表 1 空气中有害气体浓度安全上限

气体	浓度安全上限(%)
CO(一氧化碳)	0.0024
NO ₂ (二氧化氮)	0.00025
SO ₂ (二氧化硫)	0.0005
H ₂ S(硫化氢)	0.00066
NH ₃ (氨)	0.004

以上两条规定给出了井下空气中不同成分的临界值, 每一种成分在临界值上下浮动, 都会影响井下环境的危险程度。因此我们用以上 9 种气体的浓度作为 BP 网络的输入, 即输入单元数为 9; 输出单元只有一个, 就是危险程度; 采用一层隐含层, 隐含单元数按以下公式(参见文献[1])计算:

$$h = \log_2 N \quad (14)$$

式(14)中 N 为输入单元数 9, 则隐含单元数 h 可取 4。

危险程度值的有效范围是 [0.0, 1.0], 当危险程度取值在 [0.0, 0.4] 范围为安全区, 井下人员可正常作业; 取值在 [0.4, 0.6] 为一般危险区, 井下人员应密切注意环境变化; 取值在 [0.6, 0.8] 为比较危险区, 井下人员身体健康会受到伤害, 不能久留; 取值在 [0.8, 0.9] 为非常危险区, 井下人员健康度和生命值会很快降低, 应立即撤离; 取值 [0.9, 1.0] 为伤亡区, 处在相应环境中的人员会很快死亡。

某煤矿救护队的安全专家结合实践经验, 按危险程度的区域划分, 为我们设计了 200 个训练样本作为网络训练输入。

我们设定 BP 网络的各训练参数为:最大容许误差(方差) ϵ 为 0.001, 网络各层之间的初始连接权值 W_{ij} 为 $[-1, 1]$ 内的随机值, 初始学习步长 η 为 0.8。分别采用标准 BP 算法、快速传递算法、弹性传递算法在 PIII800 笔记本电脑上训练网络的迭代次数(训练周期数)和收敛时间如表 2 所示。

表 2 不同算法的迭代次数和收敛时间

算法	迭代次数	收敛时间/s
标准 BP 算法	10448	19.23
快速传递算法	6834	11.52
弹性传递算法	3365	5.64

实验中我们还发现, 最大容许误差 ϵ 越小, 不同训练算法的迭代次数相差越大, 弹性传递算法的优势也越明显。

5 结语

本文所描述的快速传递算法和弹性传递算法并未考虑局部极小问题, 局部极小问题通常采用增加动量或模拟退火的思想解决, 并且以牺牲收敛速度为代价。实际应用中人们往往可以接受接近最优的方案, 更希望神经网络能快速收敛, 以实现在线学习。尽管离线学习效果更佳, 但弹性传递算法在目前主流配置的微机上已经可以为游戏式训练系统中的角色进行在线学习。

当我们使用角色在游戏过程中的日志数据作为样本对网络进行训练时, 即使采取了各种防范措施, 数据中仍会有干扰项, 使得学习结果难于预测, 在线学习难度更大。但如果采用较完整的数据集进行离线学习, 错误数据的干扰经过平均后可以大大减小其影响, 改善学习效果。

(上接第 1939 页)

在较小的解空间内搜索最优策略, 收敛速度自然会有所提高, 如图 4 所示。

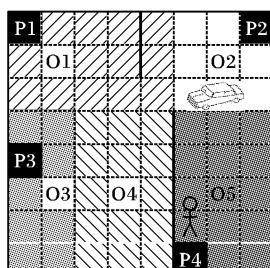


图 3 Option 状态子空间分布图

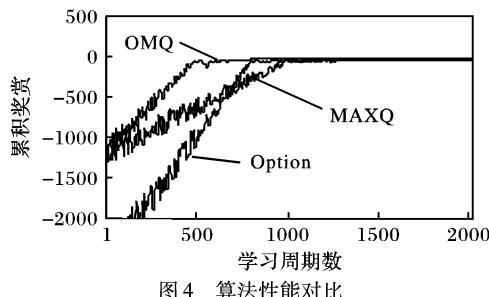


图 4 算法性能对比

图 4 中给出了 MAXQ、Option 和 OMQ 三种方法求解出租车问题的性能对比结果。由图可见, 在初始阶段, OMQ 与 MAXQ 学习获得的累加奖赏相当, 均高于 Option, 这是由于 OMQ 与 MAXQ 均利用先验知识进行了子任务的预先分解, 而 Option 方法则从完全盲目探索开始。此外, Option 和 OMQ 的收敛速度要快于 MAXQ, 这是因为 OMQ 和 Option 均能对状态

弹性传递算法对 BP 网络的收敛速度有明显的改善, 可以在游戏式训练系统中为角色提供在线和离线学习, 在其他需要快速收敛的学习网络中也应该是有应用价值的。

参考文献:

- [1] 余雪丽, 孙承意, 冯秀芳, 等. 神经网络与学习实例 [M]. 北京: 中国铁道出版社, 1996.
- [2] 蔡自兴, 徐光佑. 人工智能及其应用 [M]. 第 2 版. 北京: 清华大学出版社, 1996.
- [3] (美) CHAMPANDARD AJ. Artificial Intelligence from Theory to Fun! [EB/OL]. <http://AiGameDev.com/>, 2006.
- [4] (美) ROLLINGS A, ADAMS E. 游戏设计技术 [M]. 金名, 张长富, 译. 北京: 北京希望电子出版社, 2004.
- [5] MADEIRA C, et al. Bootstrapping the Learning process for the semi-automated Design of a challenging Game AI [A]. Proceeding of AAAI04[C]. 2004.
- [6] MENDEZ G, HERRERO P, DE ANTONIO A. Intelligent virtual environments for training in nuclear power plants [A]. Proceeding of the 6th International Conference on Enterprise Information Systems (IC-ES 2004) [C]. Proto, Portugal, 2004.
- [7] MANTOVANI F, GIANLUCA C, GAGGIOLI A, et al. Virtual Reality Training for Health-Care Professionals [J]. Cyberpsychology & Behavior, 2003, 6(4).
- [8] YU XL. Studying on Granularity of Reinforcement Learning Agents [A]. IFPS Proceeding of ICAMT/2002[C]. 2002.
- [9] 杨源杰, 黄道. 人工神经网络算法及其应用 [J]. 华东理工大学学报 2002, 28(5).
- [10] 徐昕, 贺汉根. 神经网络增强学习的梯度算法研究 [J]. 计算机学报, 2003, 26(2).

空间较大的行驶子任务进行进一步划分, 降低了子任务解空间从而加速收敛。OMQ 与 Option 收敛趋势的斜率相当, 但 OMQ 早于 Option 收敛到最优解, 这是因为 OMQ 利用了先验知识从而“起点”比 Option 高。

4 结语

本文给出了 OMQ 的理论框架和学习算法, 并以环境空间不完全可知的出租车问题为背景对算法性能进行了仿真和对比分析。实验结果表明, OMQ 能够结合 MAXQ 便于利用先验知识和 Option 易于自动生成的优势, 较好地解决了分层、分段混合、环境空间不完全可知条件下的强化学习问题。但 OMQ 算法的收敛性尚未从理论上给出证明, 这将是本文后续的工作。

参考文献:

- [1] BARTO AG, MAHADEVAN S. Recent advances in hierarchical reinforcement learning [J]. Discrete Event Dynamic Systems: Theory and Applications, 2003, 13(4): 41–77.
- [2] SUTTON RS, PRECUP D, SINGH SP. Between MDPs and semi-MDPs: a framework for temporal abstraction in reinforcement learning [J]. Artificial Intelligence, 1999, 112(1): 181–211.
- [3] PARR R. Hierarchical control and learning for markov decision processes [D]. Berkeley: University of California, 1998.
- [4] DIETTERICH TG. Hierarchical reinforcement learning with the MAXQ value function decomposition [J]. Journal of Artificial Intelligence Research, 2000, 13(1): 227–303.
- [5] 沈晶, 顾国昌, 刘海波. 分层强化学习中 Option 自动生成算法研究 [J]. 计算机工程与应用, 2005, 41(34): 4–6, 15.