

深度学习优化算法研究

仝卫国 李敏霞 张一可

(华北电力大学(保定)自动化系 河北 保定 071003)

摘要 深度学习是机器学习领域热门的研究方向,深度学习中的训练和优化算法也受到了较高的关注和研究,已成为人工智能发展的重要推动力。基于卷积神经网络的基本结构,介绍了网络训练中激活函数和网络结构的选择、超参数的设置和优化算法,分析了各算法的优劣,并以 Cifar-10 数据集为训练样本进行了验证。实验结果表明,合适的训练方式和优化算法能够有效提高网络的准确性和收敛性。最后,在实际输电线图像识别中对最优算法进行了应用并取得了良好的效果。

关键词 深度学习,卷积神经网络,激活函数,正则化,超参数,优化算法

中图分类号 TP391 **文献标识码** A

Research on Optimization Algorithm of Deep Learning

TONG Wei-guo LI Min-xia ZHANG Yi-ke

(Department of Automation, North China Electric Power University, Baoding, Hebei 071003, China)

Abstract Deep learning is a hot research field in machine learning. Training and optimization algorithm of deep learning have also been high concern and studied, and has become an important driving force for the development of artificial intelligence. Based on the basic structure of convolution neural network, the selection of activation function, the setting of hyperparameters and optimization algorithms in network training were introduced in this paper. The advantages and disadvantages of each training and optimization algorithm were analyzed and verified by Cifar-10 data set as training samples. Experimental results show that the appropriate training methods and optimization algorithms can effectively improve the accuracy and convergence of the network. Finally, the optimal algorithm was applied in the image recognition of actual transmission line and achieved good result.

Keywords Deep learning, Convolution neural network, Activate function, Regularization, Hyperparameter, Optimization algorithm

1 引言

2016 年 3 月,人工智能系统“AlphaGo”在人机世纪大战中的胜利,让公众对深度学习的关注空前高涨。深度学习已经成为近年来机器学习领域备受关注的方向^[1],在语音识别^[2]、图像识别和分类^[3]等多方面应用中都有突破性的进展和成果。

深度学习在各领域的成功应用一方面计算机硬件和大数据的支持,另一方面则是网络本身结构和超参数的合理设置和选择。合适的训练方式和网络参数直接影响网络的泛化能力,有助于加快网络收敛速度,抑制过拟合^[4]。因此,训练方式的选择和超参数的设置一直是深度学习研究中的难点和热点。

卷积神经网络是深度学习中最典型的网络结构之一,本文基于卷积神经网络的基本结构,对深度学习训练中的结构选择、参数设置和优化算法进行了研究,并以 Cifar-10 数据集作为输入数据,在卷积神经网络中比较分析了不同训练和优化算法对网络的正确率和收敛性的影响,最后将最优结果应

用在实际输电线图像识别中进行了验证。

2 卷积神经网络的基本结构

卷积神经网络(CNN)是深度学习中研究最深入、应用最广泛的网络结构。卷积神经网络的基本结构包括卷积、非线性映射和池化 3 个阶段。卷积阶段为特征提取阶段,其基于局部感受野的概念,通过卷积核实现,不同的卷积核能够提取特征图上的不同特征;非线性阶段是对卷积输出特征图采用非线性映射进行筛选,以避免线性模型表达能力不足的问题,非线性映射通过激活函数来实现;池化阶段是对图像不同位置的特征进行聚合统计,使网络不易过拟合,在保留有用特征信息的同时有效减少数据处理量。

卷积神经网络的训练是通过前向传播和反向传播过程来实现的。前向传播是通过输入计算出神经元的激活值,是自下向上的过程;反向传播根据损失计算梯度,更新网络权值,是自上向下的过程。

前向传播卷积和非线性过程的运算公式为:

$$z^{(l+1)} = W^{(l)} \otimes a^{(l)} + b^{(l)} \quad (1)$$

本文受河北省自然科学基金资助。

仝卫国(1967—),男,博士,副教授,主要研究方向为电工理论与新技术、图像处理技术与传感器,E-mail:twg1018@163.com;李敏霞(1993—),女,硕士生,主要研究方向为数字图像处理及模式识别,E-mail:15612136708@163.com(通信作者);张一可(1993—),男,主要研究方向为数字图像处理及模式识别。

$$a^{(t+1)} = f(z^{(t+1)}) \quad (2)$$

式(1)为加权求和的线性运算, W 为卷积核权重, \otimes 为卷积操作, a 为本层网络的输入, b 为偏置, z 为线性输出; 式(2)为通过激活函数 $f(x)$ 实现的非线性运算, a 为本层网络的实际输出, 也是下层网络的输入。

反向传播过程通过最小化代价函数实现, 通常使用 BP 传播算法, 运算公式为:

$$J(W, b) = \frac{1}{m} \sum J(W, b; x, y) + \frac{\lambda}{2} \sum W^2 \quad (3)$$

$$W^{(l)} = W^{(l)} - \alpha \left[\left(\frac{1}{m} \Delta W^{(l)} \right) + \lambda W^{(l)} \right] \quad (4)$$

$$b^{(l)} = b^{(l)} - \alpha \left[\frac{1}{m} \Delta b^{(l)} \right] \quad (5)$$

式(3)为代价函数, 第一项是均方差项, 第二项是规则化项(或权重衰减项), λ 为权重衰减系数; 式(4)和式(5)为权重更新公式, α 为学习率。

3 网络训练的优化

3.1 激活函数的选择

激活函数为神经网络引入非线性因素, 解决了神经网络的非线性问题, 提高了网络的表达能力, 也直接影响了神经网络的收敛速度。因此, 选择合适的激活函数是训练网络的关键一步。

传统的激活函数有 Sigmoid 和 Tanh 函数^[5], 这类函数会产生梯度消失现象, 使神经网络的收敛速度很慢甚至不收敛^[4], 并且不具备稀疏性。

2001年, 神经学家发现生物神经元的激活方式具有单侧抑制、兴奋边界宽阔以及输出具有稀疏性的特性。同年, Softplus 和 ReLU 激活函数被提出^[6], 激活曲线如图 1 所示。可以看出, Softplus 和 ReLU 函数符合生物神经元单侧抑制和兴奋边界宽阔的特性, 该特性能缓解梯度消失问题。但是, ReLU 函数具有稀疏表达能力, 可以直接以监督方式训练神经网络, 而 Softplus 不具备该特性。

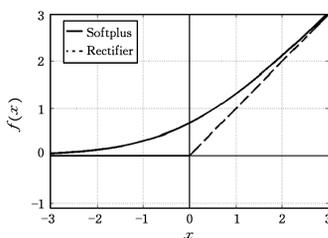


图1 Softplus 和 ReLU 激活曲线

不难发现, 所有激活函数中只有 ReLU 函数和生物神经元的激活方式十分接近。因此, 目前卷积神经网络的激活函数大多采用 ReLU 函数。文献[7]表明 ReLU 函数在训练正确率和网络收敛速度上均优于其他激活函数。

3.2 网络结构的选择

大规模神经网络的训练是非常费时的, 并且当网络训练样本较少时, 还会造成网络过拟合。为防止过拟合, 传统的方法是采用 ensemble 方法, 即将多个模型组合训练, 虽然训练效果很好, 但极大地增加了训练时间, 未解决费时问题。

2012年, Hinton^[8]提出的 Dropout 算法很好地解决了上述问题。Dropout 是通过修改网络本身来实现过拟合抑制的, 如图 2 所示, 该算法按照一定的概率在训练中的每次迭代随机“删除”隐层单元, 一方面, 一个 n 节点的网络相当于 $2n$

个模型的集合, 但训练参数数目却不变, 解决了费时问题; 另一方面, 随机选择训练降低了神经元节点间的联合适应性, 由此增强了网络的泛化能力。

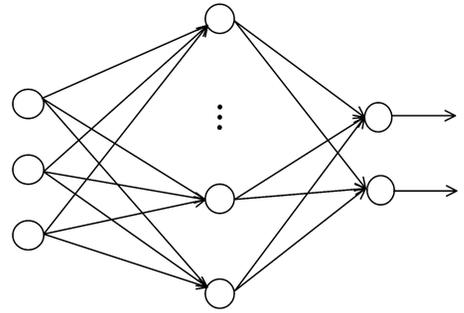


图2 Dropout 算法的网络结构

3.3 超参数的设置

卷积神经网络的典型超参数有网络层数、单层神经元数、权值初始化、正则惩罚项和学习率。

网络层数和各层神经元数目是神经网络中的重要参数, 数目越多, 网络的表达能力和拟合能力越强, 越有利于解决复杂的问题。但是, 在训练数据不足的情况下, 网络的层数和神经元数目过多又易发生过拟合。因此, 网络层数和神经元数目的设置需要与研究对象相适应。目前, 神经网络的层数和神经元数目的参数选取还没有经验公式可供参考, 实际的设置往往通过实验过程中的结果来不断调整或借鉴参考其他现有的典型网络。

初始化权重的设置对网络的训练会产生极大的影响, 不合适的初始化方法会损害基于梯度的学习方法, 使得效率得不到充分发掘^[4]。文献[5]给出了激活函数 Sigmoid 和 Tanh 的初始化经验公式, 权重初始化的随机值分布如式(6)和式(7)所示。文献[7]给出了 ReLU 系列的激活函数的初始化经验规则, 权重初始化的随机值分布如式(8)所示。

$$W \sim U \left[-\frac{4\sqrt{6}}{\sqrt{n_i + n_{i+1}}}, \frac{4\sqrt{6}}{\sqrt{n_i + n_{i+1}}} \right] \quad (6)$$

$$W \sim U \left[-\frac{\sqrt{6}}{\sqrt{n_i + n_{i+1}}}, \frac{\sqrt{6}}{\sqrt{n_i + n_{i+1}}} \right] \quad (7)$$

$$W_i \sim N \left[0, \frac{\sqrt{2}}{\sqrt{k_i^2 \cdot n_i}} \right] \quad (8)$$

其中, U 表示均匀分布, n_i 表示第 i 层神经元数, k_i 表示第 i 层卷积层的卷积核尺寸。

正则惩罚项的设置是为了减少网络过拟合的影响。网络的代价函数都是人为设定的, 一般比较容易出现过拟合现象, 导致模型的复杂度过高, 对未知数据的预测能力较差。引入正则惩罚项, 能够降低模型复杂度, 减少过拟合的影响, 提高模型的泛化能力。正则项系数的大小对网络的泛化能力也会有影响。如果正则项系数过大, 则对模型的复杂度惩罚大, 不会过分拟合数据, 但在训练数据上偏差较大, 在预测数据上方差较小, 可能出现欠拟合现象; 如果过小, 可能不能起到抑制过拟合的作用。常见的正则化方法有 L1 regularization 和 L2 regularization。

3.4 训练优化算法

深度学习算法的目标函数几乎都是非凸的, 只能通过迭代的方法来不断逼近最优解。常见的优化算法有梯度下降法、牛顿法和共轭梯度法等^[9]。

3.4.1 梯度下降法

梯度下降算法(Gradient Descent, GD)是机器学习中使用得非常广泛的优化算法,也是众多机器学习算法中最常用的优化方法,迭代公式为:

$$\theta = \theta - \nabla_{\theta} J(\theta) \quad (9)$$

批量梯度下降法(BGD)是梯度下降法最原始的形式,通过最小化所有训练样本的损失函数得到全局最优解。

算法描述为:

$$\theta_j = \theta_j + \frac{1}{m} \sum_{i=1}^m (y^i - h_{\theta}(x^i)) x^i \quad (10)$$

BGD 算法每次迭代会用到整个训练集,能够保证收敛于极值点,但若 m 很大,迭代速度将会很慢,也不能进行在线模型的参数更新。

随机梯度下降算法(SGD)是为解决 BGD 算法因训练样本数量增大导致训练速度变慢这一弊端而提出的。

算法描述为:

$$\theta_j = \theta_j + (y^i - h_{\theta}(x^i)) x^i \quad (11)$$

对比 BGD,SGD 能够进行在线更新,运算量大幅度减少,尤其是在 m 很大的情况下。但 SGD 算法的噪音比 BGD 多,这使得 SGD 不是每次迭代都向整体的最优化方向进行,从而带来优化波动,使收敛速度变慢,一般不单独使用。

小批量梯度下降法(MBGD)综合了 BGD 和 SGD,在每次更新速度与更新次数中取得一个平衡,每次更新从训练集中随机选择 $n(n < m)$ 个样本进行学习。

算法描述为:

$$\theta_j = \theta_j - \nabla_{\theta} J(\theta; x_{i:i+n}; y_{i:i+n}) \quad (12)$$

相对于 SGD,MBGD 降低了收敛波动性,使得更新更加稳定。相对于 BGD,其提高了每次学习的速度。

AdaGrad^[10]是借鉴正则化思路来更新参数的自适应梯度算法,更新公式如式(13)所示。该算法每次迭代都将沿路的梯度的平方根作为正则项,并将其作为分母。这一方面有利于解决梯度消失问题,另一方面使学习率随着时间减小,避免了常规梯度算法中人为设定学习率,对网络的训练进程和合理性更加有利。但是,该算法仍依赖人工设置的全局学习率。

$$\theta_{t+1} = \theta_t - \frac{\epsilon}{\sqrt{\sum_{\tau=1}^t (\nabla f(\theta_{\tau}))^2}} \nabla f(\theta_t) \quad (13)$$

AdaDelta^[11]是对 AdaGrad 的扩展,只积累有限个最新梯度的平方,防止了 AdaGrad 中梯度不断积累造成的训练停滞问题,相对 AdaGrad 在计算上进行了简化,并且不依赖于全局学习率。但是,AdaDelta 在训练后期进入局部极小值后,会反复在局部极小值附近抖动,并且精度上的抖动也比较大。

RMSProp^[12]也是 AdaGrad 的一种扩展,它使用均方根自动更新学习率,不需要设置初始学习率,但需要线性搜索初始学习率,没有根本实现自适应的学习率,效果在 AdaGrad 和 AdaDelta 之间,适合处理非平稳目标。

Adam^[13]算法是利用梯度的一阶矩估计和二阶矩估计动态调整每个参数的学习率,与 AdaDelta 和 RMSProp 的区别在于计算历史梯度衰减方式上。它没有使用历史平方衰减而是使用了类似动量的衰减方法,实质上就是带有动量项的 RMSProp。该算法结合了 AdaGrad 善于处理稀疏梯度和 PMSProp 善于处理非平稳目标的优点,在自适应学习率方面表现较好,适用于大多非凸优化,也适用于大数据集和高维空间。

Momentum 法是为解决 SGD 更新方向完全依赖于当前的 batch 从而更新十分不稳定而提出的。Momentum 即动量,它模拟的是物体运动时的惯性,积累之前的动量来替代真正的梯度。即更新时在一定程度上保留之前更新的方向,同时利用当前 batch 的梯度微调最终的更新方向,在一定程度上增加稳定性,从而更快地学习,并有摆脱局部最优的能力。Momentum 法用惯性系数来决定多大程度上保留之前的更新方向,惯性系数范围在 0~1 之间。在训练开始时,由于梯度可能会很大,因此初始值一般选为 0.5;当梯度较小时,改为 0.9。在下降初期,使用上一次参数更新,下降方向保持一致,乘上较大的惯性系数能够进行很好的加速;在下降中后期,局部最小值来回震荡,梯度趋于 0,惯性系数使得更新幅度增大,跳出陷阱。总之,Momentum 能够在相关方向加速 SGD,抑制震荡,从而加快收敛^[3]。

受 Nesterov 加速梯度算法^[14-15]的启发,Sutskever 等于 2013 年提出动量算法的一个变种 Nesterov 动量。Nesterov 动量和标准动量之间的区别体现在梯度计算上。Nesterov 动量中,梯度计算在施加当前速度之后,因此,Nesterov 动量可以解释为往标准动量方法中添加了一个校正因子。在凸批量梯度情况下,Nesterov 动量将额外误差收敛率从 $O(1/k)$ (k 步后)改进到 $O(1/k^2)$ (见文献^[14])。但在随机梯度情况下,Nesterov 动量没有改进收敛率^[16]。

3.4.2 牛顿法

牛顿法的基本思想是利用迭代点处的一阶导数(梯度)和二阶导数(Hessian 矩阵)对目标函数进行二次函数近似,然后把二次模型的极小点作为新的迭代点,并不断重复这一过程,直至求得满足精度的近似极小值。其公式为:

$$\Delta \theta_j = H_j^{-1} \cdot g_j \quad (14)$$

牛顿法用 Hessian 矩阵替代人工设置的学习率,不会陷入局部极小值。与梯度下降法相比,牛顿法是二阶收敛,而 GD 算法是一阶收敛,因此牛顿法的收敛速度更快,能高度逼近最优值,几何上下降路径也更符合真实的最优下降路径;而且因 Hessian 矩阵的逆在迭代中会不断减小,起到逐渐缩小步长的效果。但是,计算 Hessian 矩阵的逆的时间复杂度近似 $O(n^3)$,该算法不适合大数据。

为解决计算 Hessian 矩阵费时的问题,研究者提出了许多近似 Hessian 矩阵的方法,被称为拟牛顿算法,L-BFGS 算法就是其中应用最广泛的一种。L-BFGS 算法具备牛顿法收敛速度快的特点,但不需存储 Hessian 矩阵,节省了大量存储空间和计算资源^[17]。

3.4.3 共轭梯度法

共轭梯度法(CG)是介于 GD 和牛顿法之间的一个方法,只需要利用一阶导数的信息,克服了 GD 方法收敛速度慢的特点,又避免了牛顿法中的计算和存储 Hessian 矩阵信息,大大节省了时间和空间开销。相比最速梯度下降法,共轭梯度下降法仅需要 n 次迭代(这里 n 代表 n 个共轭方向)就可以收敛。但是共轭梯度法最大的缺点仍然是每次迭代需要用到所有的训练数据。

CG 算法的基本思想是把共轭性与最速下降方法相结合,利用已知点的梯度构造一组共轭方向,并沿这组方向进行搜索,求出目标函数的极小点,迭代公式为:

$$x_{k+1} = x_k + \alpha_k d_k$$

$$d_k = \begin{cases} -g_k, & k=0 \\ -g_k + \beta_k d_{k-1}, & k \geq 1 \end{cases} \quad (15)$$

其中, g_k 是目标函数在 x_k 点处的梯度, β_k 为标量。

研究表明^[9], L-BFGS 算法在参数纬度比较低时的效果优于 SGD 和 CG, 尤其是对带有卷积的模型。而针对高维参数网络, CG 的效果优于 L-BFGS 和 SGD。因此, 优化算法的优劣也因与目标函数的不同而有所差异, 不同的优化算法适用于不同的优化对象。

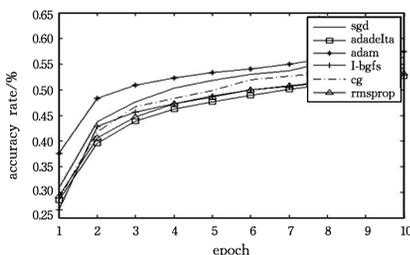
3.4.4 其他优化算法

除以上几种优化算法外, 还有以下几种方法: 1) 最小二乘法^[18]。最小二乘法 (Ordinary Least Squares, OLS) 的核心是保证所有数据偏差的平方和最小。2) 偏最小二乘法^[19]。偏最小二乘法是一种新的多元统计分析方法, 与以前普通的最小二乘回归研究一个因变量和多个自变量之间的关系不同, 偏最小二乘回归研究的是多因变量对多自变量的回归建模^[20]。特别是当变量存在多重相关性或样本点数据少于变量个数时, 采用偏最小二乘回归模型更为有效^[21]。3) 启发式优化方法 (包括模拟退火方法、遗传算法、蚁群算法以及粒子群算法等)^[22-23]。启发式算法是为了提高搜索效率而提出的, 是人们在解决问题时所采取的一种根据经验规则进行发现的方法。其特点是在解决问题时, 利用过去的经验, 选择已经行之有效的方法, 以随机或近似随机方法搜索非线性复杂空间中的全局最优解。启发式解决问题的方法是与算法对立的, 算法是把各种可能性都一一进行尝试, 最终找到问题的答案, 但它是在很大的问题空间内, 花费大量的时间和精力才能求得答案; 启发式方法在有限的搜索空间内, 大大减少了尝试的数量, 能迅速解决问题。

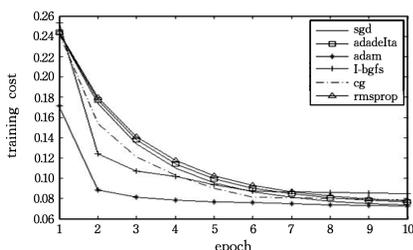
4 实验分析和应用

4.1 Cifar-10 数据集上的实验分析

Cifar-10 数据集是由 Alex Krizhevsky, Vinod Nair 和 Geoffrey Hinton 收集的。此数据集包含 60000 张共 10 类 32×32 大小的彩色图片, 每一类中有 6000 张图片, 将其中 5000 张作为训练样本, 1000 张作为测试样本。本实验以 Cifar-10 数据集为训练数据, 对不同的优化算法对网络训练的影响进行比较分析。分别采用 SGD, Adam, AdaDelta, L-BFGS, CG 和 RMSProp 算法对网络进行训练, 得到网络的正确率曲线和损失曲线, 如图 3 所示。



(a) 正确率曲线



(b) 收敛曲线

图3 Cifar-10 数据集在不同优化算法下的测试曲线

在正确率方面, Adam 算法最优; 其次是 SGD 算法; CG 算法略逊于 SGD; L-BFGS 算法在训练初期的正确率曲线略优于 AdaDelta 和 RMSProp, 但训练后期二者基本一致, 最终正确率最低。在代价损失方面, L-BFGS 的收敛速度最快, 明显优于其他算法; 其次是 Adam, 但它的最终损失趋于最小; SGD, RMSProp 和 AdaDelta 算法的损失曲线基本一致, 收敛速度一般, 最终损失略高于 Adam, 效果较好; CG 算法的收敛速度比 L-BFGS 慢, 但优于 SGD, 最终损失和 SGD 相近。

4.2 在实际输电线路图像识别中的应用

本文应用卷积神经网络对实际的输电线路图像进行识别。训练数据为 17000 张实际图像, 包括输电线路图像和其他图像, 大小为 $96 \times 96 \times 3$ 。

结合网络优化的分析, 网络的激活函数选择 ReLU 函数; 权重的初始化按照式 (8) 设置; 偏置初始化为 0; 综合网络损失和正确率, 权重更新采用 Adam 算法, 在全连接层使用 dropout, dropout 率设置为 0.5。

网络训练时, 分别采用六层和七层网络进行训练, 结果显示, 六层网络的训练集正确率为 82.9%, 检测集正确率为 76.9%, 七层网络分别为 89.9% 和 78.4%, 识别效果均较好。虽然整体上六层网络的正确率较七层网络低, 但六层网络的训练集和测试集正确率之差远低于七层网络, 故六层网络略优于七层网络, 因此最终选择六层网络, 网络设置如表 1 所列。

表1 网络结构设置

层号	1	2	3	4	5	6(输出)
方式	卷积池化	卷积池化	卷积	卷积池化	全连接	全连接
神经元数	64	128	192	192	384	4
卷积核大小	7 * 7	5 * 5	5 * 5	5 * 5	—	—
池化尺寸	6 * 6	—	—	3 * 3	—	—
输入大小	96 * 96	15 * 15	11 * 11	7 * 7	1 * 1	1 * 1

结束语 本文通过实验比较了激活函数、正则化系数和优化算法对网络训练的影响, 结果表明 ReLU 激活函数对特征的表达最强且最符合神经学特性。网络训练中 Adam 算法的收敛速度最快, 正确率最高; 当训练数据不足时, Dropout 方法能够有效抑制过拟合。该结论对大幅降低网络训练时间、优化网络性能、提高网络泛化能力具有重要的参考价值。

深度学习是机器学习领域的重要研究方向, 越来越多的学者对其进行研究。近几年, 深度学习得到了空前的发展, 但优化算法的选择是影响深度神经网络学习效果的重要因素, 很多时候算法的选择仍很费时, 因此自适应地选择优化学习过程将是深度学习发展的一个趋势。目前深度学习还没有在工程中大范围使用, 随着硬件和深度学习理论研究的成熟, 深度学习将更多地应用于工程中。

参考文献

- [1] 刘建伟, 刘媛, 罗雄麟. 深度学习研究进展[J]. 计算机应用研究, 2014, 31(7): 1921-1930.
- [2] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet Classification with Deep Convolutional Neural Networks[J]. Advances in Neural Information Processing Systems, 2012, 25(2): 2012.
- [3] SUTSKEVER I, MARTENS J, DAHL G, et al. On the importance of initialization and momentum in deep learning[C]// International Conference on Machine Learning. Dasgupta and Pro-

- ceedings of the 30th International Conference on Machine Learning (ICML-13). 2013:1139-1147.
- [4] NAIR V, HINTON G E. Rectified Linear Units Improve Restricted Boltzmann Machines[C]// International Conference on Machine Learning. Omnipress, 2010:807-814.
- [5] GLOROT X, BENGIO Y. Understanding the difficulty of training deep feedforward neural networks[J]. Journal of Machine Learning Research, 2010, 9:249-256.
- [6] DAHL G E, SAINATH T N, HINTON G E. Improving deep neural networks for LVCSR using rectified linear units and dropout[C]// IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2013:8609-8613.
- [7] GLOROT X, BORDES A, BENGIO Y. Deep Sparse Rectifier Neural Networks[J]. Journal of Machine Learning Research, 2010, 15:315-323.
- [8] HINTON G E, SRIVASTAVA N, KRIZHEVSKY A, et al. Improving neural networks by preventing co-adaptation of feature detectors[J]. Computer Science, 2012, 3(4):212-223.
- [9] LE Q V, JIQUAN N, ADAM C, et al. On optimization methods for deep learning[C]// Proceedings of the 28th International Conference on International Conference on Machine Learning. 2011:265-272.
- [10] DUCHI J, HAZAN E, SINGER Y. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization[J]. Journal of Machine Learning Research, 2010, 12(7):257-269.
- [11] ZEILER M D. ADADELTA: An Adaptive Learning Rate Method[J]. arXiv:1212.5701.
- [12] TIELEMANT, HINTON G. RMSProp: Divide the gradient by a running average of its recent magnitude[R]. COURSE: Neural Networks for Machine Learning. 2012.
- [13] KINGMA D, BA J. Adam: A Method for Stochastic Optimization[J]. arXiv:1412.6980.
- [14] NESTEROV Y. A method of solving a convex programming problem with convergence rate $O(1/k^2)$ [J]. Soviet Mathematics Doklady, 1983(27):372-376.
- [15] NESTEROV Y. Introductory lectures on convex optimization: a basic course[M]. Applied Optimization Kluwer Academic, 1998: 119-120.
- [16] GOODFELLOW I, BENGIO Y, COURVILLE A. 深度学习[M]. 符天凡等, 译. 北京: 人民邮电出版社, 2017:240-249.
- [17] FLETCHER R, POWELL M J D. A Rapidly Convergent Descent Method for Minimization[J]. Computer Journal, 1963, 6(6):163-168.
- [18] MILLER S J. The Method of Least Squares[D]. Brown University, 2006:1-7.
- [19] 朱志鹏, 喻芳, 曾青霞, 等. 基于深度学习与偏最小二乘的分析方法及其医学应用[J]. 江西中医药大学学报, 2017, 29(3):94-97.
- [20] SEBER G A F, LEE A J. Linear regression analysis[M]. Wiley, 2012.
- [21] CRISTIANINI N, SHAWER-TAYLOR J. An introduction to support vector machines and other kernel-based learning methods[M]. Cambridge University Press, 2000.
- [22] KENNY V, NATHAL M, SALDANA S. Heuristic algorithms [C]// ChE 345. Spring, 2014.
- [23] EISELT H A, SANDBLOM C L. Heuristic Algorithms[C]// Integer Programming and Network Models. 2000:229-258.

(上接第 145 页)

结束语 基于无线传感网络设计道路交通管理系统的最大挑战是实现最优的信号灯控制以尽可能提高交通率并最小化道路拥塞。强化学习给出了求解该问题的有效方法。本文为交通信号灯的自适应控制设计了基于特征表示的 Q 学习算法, Q 学习不需要构建系统模型, 能实时地适应交通状况。基于特征表示的函数近似可以有效地控制大规模交通网络的计算复杂度。仿真结果表明本文提出的 Q-learning 交通信号灯控制方法优于其他几个常用的算法。

未来的工作是在实际交通环境中进行实验, 验证本文算法的优势; 同时综合考虑多项信号控制指标, 使得奖惩函数的设计符合实际的道路状况。

参考文献

- [1] ADAM I, WAHAB A, YAAKOP M, et al. Adaptive fuzzy logic traffic light management system[C]// 2014 4th International Conference on Engineering Technology and Technopreneuship (ICE2T). IEEE, 2014:340-343.
- [2] COOLS S B, GERSHENSON C, D'HOOGHE B. Self-Organizing Traffic Lights: A Realistic Simulation[J]. Advances in Applied Self-Organizing Systems, 2016, 17(4):45-55.
- [3] KAUR T, AGRAWAL S. Adaptive Traffic Lights Based on Hybrid of Neural Network and Genetic Algorithm for Reduced Traffic Congestion[C]// Recent Advances in Engineering and Computational Sciences (RAECS). 2014:1-5.
- [4] SRINIVASAN D, CHOY M C, CHEU R L. Neural Networks for Real-Time Traffic Signal Control[J]. IEEE Transactions on Intelligent Transportation Systems, 2006, 7(3):261-272.
- [5] SUTTON R S, BARTO A G. Introduction to reinforcement learning [J]. IEEE Transactions on Neural Networks, 1992, 8(3/4):225-227.
- [6] 高阳, 陈世福, 陆鑫. 强化学习研究综述[J]. 自动化学报, 2004, 30(1):86-100.
- [7] 刘忠, 李海红, 刘全. 强化学习算法研究[J]. 计算机工程与设计, 2008, 29(22):5805-5809.
- [8] SALKHAM A, CUNNINGHAM R, GARG A, et al. A Collaborative Reinforcement Learning Approach to Urban Traffic Control Optimization[C]. IEEE/WIC/ACM International Conferent on Web Intelligence and Intelligent Agent Technology. 2008: 560-566.
- [9] XIE Y C. Development and evaluation of an arterial adaptive traffic signal control system using reinforcement learning[OL]. <http://holl.hardle.net/1969.1/ETD-TAMU-2480>.
- [10] WATKINS C, DAYAN P. Q-learning [J]. Machine Learning, 1992, 8(3/4):279-292.