

基于递归神经网络的桥梁变形趋势预测研究

王刚

(中交基础设施养护集团有限公司, 北京 100011)

摘要: 随着物联网的应用普及, 桥梁的安全监测逐步向全自动、全天候、实时、高精度方向发展, 但这些监测手段仅能对现有桥梁安全问题进行监测预警, 无法提前预测其安全隐患, 使桥梁养护止步于“管”和“控”, 而无法实现“防”。鉴于此, 文中利用递归神经网络的循环连接特性建立桥梁健康状况评估模型, 结合工程实例进行应用分析, 结果表明递归神经网络在桥梁变形趋势预测、隐患发掘及健康状况评估方面具有较大优势。

关键词: 工程管理; 桥梁; 变形趋势; 递归神经网络; 按时间反向传播算法(BPTT)

中图分类号: U415.12

文献标志码: A

文章编号: 1671-2668(2018)02-0164-06

据不完全统计, 目前中国有 1/3 的桥梁存在各类缺陷, 危桥已超过 1 万座。对桥梁进行健康状况评估具有重要的社会意义与经济意义。现有桥梁健康评估方法主要为多层次评估模型、桥梁质量模糊综合评价方法、多级模糊综合评判法、基于设计规范的桥梁承载能力评定等, 这些方法都是根据大量动静荷载试验、桥梁结构计算结果等对桥梁运营状况进行评估。随着 GPS、北斗卫星定位技术的发展及信息化时代的来临, 桥梁自动化监测逐渐普及, 全天候、自动化、实时的桥梁监测模式成为可能。自动化监测也为递归神经网络在桥梁健康评估中的应用提供了数据支撑。该文利用递归神经网络对武汉市白沙洲大桥往期监测数据进行训练和预测, 探讨递归神经网络在桥梁变形趋势预测中的应用。

1 递归神经网络理论研究

递归神经网络(RNNs)是一种神经元有一个或多个自身反馈环的网络, 反馈可以是局部的, 也可以是全局的。其与传统深层神经网络的不同之处在于, 传统深层神经网络中所有层的输入数据处理方向都是统一的, 而递归神经网络具有额外的循环连接, 在隐藏层间使用循环连接, 不再使用固定的多层次非线性结构。这些独特的反复连接使神经网络可存储供后续使用的信息并捕获输入数据之间的长距离依赖, 被视为一种可高效处理非线性任务的模型, 多用于处理和预测序列数据。

1.1 前向传播与反向传播

前向传播和反向传播是计算神经网络非常重要的部分。前向传播是通过网络的输入计算输出结果

的过程, 即产生误差值的过程; 反向传播是通过网络的输出结果计算误差梯度, 对连接权重进行调整的过程, 即修正误差的过程。前向传播时, 输入的样本从输入层经过隐层神经元一层一层进行处理, 通过所有隐层后传向输出层。在逐层处理的过程中, 每一层神经元的状态只对下一层神经元的状态产生影响。在输出层把现行输出和期望输出进行比较, 如果现行输出不等于期望输出, 则进入反向传播。反向传播时, 把误差信号按原来正向传播的通路反向传回, 并对每个隐层的各神经元的权系数进行修改, 使误差信号趋向最小。

在输入数据不变的情况下, 神经网络中的各种超参数是反向传播算法通过神经网络的输出敏感度计算出来的。由于训练过程中要学习输入的数据, 需反复对神经网络中的神经元进行调整, 而神经元的表示方式为权值矩阵, 方法是通过计算得出损失函数的梯度。损失函数的定义见式(1)。在得到损失函数后, 即可运用偏微分求导的链式法则不断向序列前的层计算, 从而得到梯度, 并最终传到输入层。图 1 为前向和反向传播过程示意图。

$$Loss = [t - f(X_n; w_0, w_1, \dots, w_n)]^2 \quad (1)$$

式中: t 为期望输出; f 为递归神经网络实际输出。

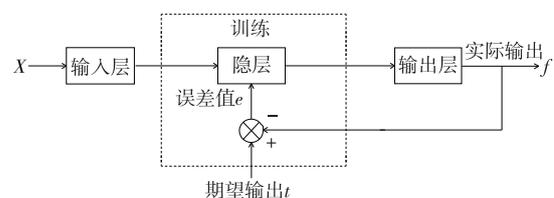


图 1 递归神经网络前向和反向传播示意图

将样本 X 作为输入提供给递归神经网络的输入层,由训练部分对网络的权值矩阵进行调整,最后由输出层输出结果。如图 1 所示,期望输出 t 可作为教师信号(神经网络学习中由外部提供的模式样本信号)输入,并将其与网络训练的实际输出 f 进行比较,根据产生的误差信号 e 控制修改学习系统的权值矩阵。通过反复修改,使误差 e 逐渐减小,直至达到期望的误差目标,这时网络训练得到的实际输出 f 和期望输出 t 极为接近,学习过程结束。

神经网络的学习过程往往包含大量重复训练,误差值在该过程中通过不断调整权值矩阵而逐渐减小,慢慢趋近于零甚至达到零,得到所期望的输出。正是因为学习过程中的反复训练,且每次训练都会对网络权值进行调整,神经网络的学习往往比较耗时,训练级次达到上万次的也较常见。目前,提高神经网络学习速度的一个重要手段就是减少学习重复次数,这也是实时控制中的关键所在。

反向传播误差是训练人工神经网络的方法之一,它是一种监督学习方法,其学习途径是利用标记的训练数据。在监督学习中,训练集是已标注的,换言之,对于一些给定的输入,事先已知其对应的期望输出。反向传播算法中,初始化过程的全部边权重都是随机分配的。对于输入的训练数据集,人工神经网络会对其进行训练并校验对应的输出。方法如下:将已知期望输出与实际输出进行比较得到误差信号,并将该信号回传给上一层,根据误差信号对权重作出调整。该过程经过多次重复后,当输出误差达到所期待的标准时,训练结束。反向传播算法完成后,所得到的就是一个学习过的人工神经网络,该网络可接受新的输入数据集进行应用。

反向传播算法是训练神经网络模型的核心算法,它可根据定义好的损失函数优化神经网络中的参数,从而使神经网络模型在训练数据集上的损失函数达到较小值。神经网络模型中的参数优化好坏直接决定模型的最终质量,是使用神经网络非常重要的一部分。

1.2 梯度下降算法

梯度下降是一种迭代方法,在机器学习领域,其主要用于获得损失函数的最小值。其中损失指的是衡量预测错误代价的量化指标。给定一组参数,使用梯度下降计算对应模型的预测损失,然后通过调整这些参数以减少损失。重复该过程,直到损失无法继续减少,则最后一组参数就是最终的拟合模型。

梯度下降是遵循成本函数的梯度来最小化一个目标函数的过程,该目标函数即为梯度下降法的核心。在每次迭代过程中,对各变量按照目标函数在该变量梯度的反方向更新对应的参数值,而目标函数局部到达最小值的迭代次数是由学习速率决定的。梯度下降算法无法保证被优化的函数达到全局最优解,可能得出的只是局部最优,这与参数的初始值有很大关联。仅当损失函数为凸函数时,梯度下降算法才能保证达到全局最优。梯度下降算法的另一问题是计算时间太长,每轮迭代中都需计算在全部训练数据上的损失函数,在海量样本数据下,计算量过于庞大。

针对上述问题,开发了随机梯度下降算法。这种优化算法优化的不是在全部训练数据上的损失函数,而是在每轮迭代中随机优化某一条训练数据上的损失函数,从而大幅提升整体计算速度。但某一条数据上损失函数小并不代表在全部数据上的损失函数小,因此仅使用随机梯度下降算法优化得到的神经网络模型可能无法达到局部最优。而且现在的机器计算性能大大提高,随机梯度下降法的效率相对较低,远没有达到机器的性能上限,会浪费大部分计算资源。

1.3 递归神经网络结构

递归神经网络的设计步骤与前向神经网络相似,都需确定以下内容:1) 网络结构,包括网络类型、层数、输入/输出层节点数及各层的激励函数;2) 网络学习算法,包括梯度下降法、学习速率等;3) 训练程序的编写,包括流程图、代码等;4) 网络参数的选取,包括训练前初始学习速率的选取、隐层节点数的确定、初始权值的选取、最优学习速率的确定、学习速率调节顺序的选择、权值动量因子的调节;5) 网络的最后选定。按上述步骤经过反复调试,就能设计出一个理想的网络。

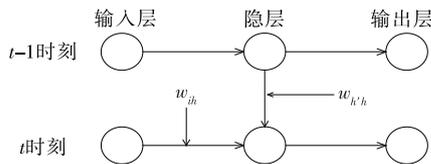
网络结构通常包括以下几方面:1) 网络类型。目前,递归神经网络结构已有多种类型,且其功能各异。对不同类型的待处理系统,选取合适类型的递归神经网络结构尤为重要,是能否成功控制和辨识系统的关键所在。2) 网络层数。递归神经网络的网络层数与待处理系统的复杂程度相关。待处理系统越复杂,则隐层数量增加,相应地网络参数和训练时间增加;反之,系统越简单,则选用单个隐层处理最理想。选择合适的隐层使训练时间短同时精确度尽可能高是需重点考虑的问题。3) 输入/输出层节

点数。递归神经网络的输入和输出节点数取决于待处理问题本身,考虑问题的方式不同,输入及输出层节点数各异。4) 网络各层的激活函数。递归神经网络各层通常采用 S 形激活函数,主要目的是体现其非线性逼近能力。激活函数通常可位于网络的隐层和输出层中。S 形激活函数虽然可减少网络训练时间,但网络输出被限制在 $(-1, 1)$ 区间内,而一般的网络输出区间通常为 $(-m, n)$, 其中 m 和 n 均为大于 1 的常数。针对该情况,对输入/输出作归一化处理必不可少。

2 递归神经网络模型的训练

2.1 按时间反向传播算法

按时间反向传播算法(BPTT)源于 BP 算法的拓展。二者区别在于,BP 算法只考虑上下层级之间梯度的纵向传播,而 BPTT 同时考虑层级之间的纵向传播和时间上的横向传播,同时在两个方向进行参数优化。按时间序列将 BPTT 算法展开,其内部之间同样使用 BP 算法训练;其前向传播过程的结果按时间序列的顺序依次通过激励函数计算得到。RNNs 输入和输出层的每次迭代过程都要加上时刻,隐层的输入向量需加上前一时间刻隐层的输出向量。图 2 为递归神经网络计算过程示意图。



w_{ih} 表示隐藏层和输入层的权值关系; $w_{h'h}$ 表示 t 时刻隐层和 $t-1$ 时刻隐层的权值关系。

图 2 递归神经网络计算过程示意图

定义 t 时刻的输入值为 x_i^t 、 $t-1$ 时刻隐层的输出值为 b_h^{t-1} 、 t 时刻隐层的输入向量为 a_h^t , 则 a_h^t 的表达式为:

$$a_h^t = \sum_{i=1}^I w_{ih} x_i^t + \sum_{h=1}^H w_{h'h} b_h^{t-1} \quad (2)$$

t 时刻隐层的输入向量为:

$$b_h^t = \theta_h(a_h^t) \quad (3)$$

式中: θ_h 为隐层的激励函数。

t 时刻输出层的输入向量为:

$$a_o^t = \sum_{h=1}^H w_{ho} b_h^t \quad (4)$$

式中: w_{ho} 为输出层和隐层之间的权值关系。

t 时刻输出层的输出向量为:

$$b_o^t = \theta_o(a_o^t) \quad (5)$$

根据 BPTT 前向传播结果,将误差函数记为 L , 定义 $\delta_j^t = \partial L / \partial a_j^t$, 则 t 时刻误差信号为:

$$\delta_h^t = \theta'(a_h^t) \left(\sum_{o=1}^O \delta_o^t w_{oh} + \sum_{h=1}^H \delta_h^{t+1} w_{hh'} \right) \quad (6)$$

误差针对权值的偏导数为:

$$\nabla L(w_{ij}) = \frac{\partial L}{\partial w_{ij}} = \sum_{t=1}^T \frac{\partial L}{\partial a_j^t} \frac{\partial a_j^t}{\partial w_{ij}} = \sum_{t=1}^T \delta_j^t b_i^t \quad (7)$$

从而得到权值更新公式:

$$w_{ij} = w'_{ij} - \eta \nabla L(w_{ij}) \quad (8)$$

2.2 递归神经网络的训练过程

递归神经网络的一般训练流程:1) 在 t 时刻输入训练特征向量进入输入层神经单元得到输出向量。2) 将输入层的输出向量加上 $t-1$ 时刻隐层的输出作为隐层的输入向量。3) 隐层 t 时刻的输入向量通过激励函数和阈值计算出隐层的输出向量,该输出向量即为输出层的输入向量,再通过激励函数求得输出层的值。4) 将第三步得到的结果与期望输出结果对比进行误差计算。5) 根据误差进行反向传播更新各层的权值。6) 重复该过程,得到期望误差,训练结束。

递归神经网络训练的实现过程:1) 加载数据并进行归一化处理。为消除指标间不同量纲的影响,对数据进行标准化处理,使数据指标之间具有可比性。将输入向量中的各元素除以向量的模,形成的新向量即为归一化结果。为避免出现欠拟合现象,使用局部加权线性回归对训练数据集中的样本进行加权处理,初始化权重均为零。2) 网络结构初始化。网络结构初始化的内容为输入层、隐层及输出层的节点数。其中输入层节点数为 75,隐层为 250,输出层为 25。3) 连接权重初始化。为各连接权值分别赋一个 $(-1, 1)$ 内的随机数。如图 3 所示,以 V 表示输入层到隐层的连接权重, W 表示隐层到输出层的连接权重, U 表示上一次隐层到本次隐层的连接权重。4) 计算隐层各神经元节点的输入和输出。5) 误差计算。利用网络期望输出和实际输出,计算误差函数对输出层的各神经元的偏导数。6) 权值更新。利用步骤 5 求得的输出层、隐层各神经元的偏导数和隐层、输入层各神经元节点的输出修正连接权值。7) 计算全局误差,绘制 Error-Cost 曲线,该曲线体现模型的收敛过程。

综上,递归神经网络模型的训练实际上是误差计算和权值更新的过程。为避免因模型复杂度较低

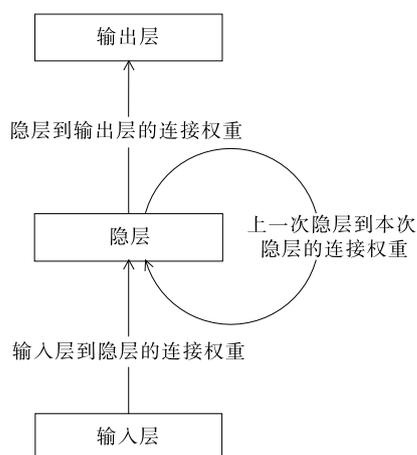


图3 递归神经网络权值表示

而产生欠拟合现象,训练过程中使用局部加权线性回归方法对训练数据集中的样本进行加权处理。

3 实例应用

3.1 试验数据

试验数据来源于武汉白沙洲大桥安全监测结果,共8期,每期数据为该桥一年的监测结果,包含分布在桥面的60个监测点的高程信息,监测点均匀分布在上下游两侧,且基本对称,除去塔顶偏位等监测点,取下游即武昌至汉阳方向的25个监测点数据。这些点位数据体现了桥梁的整体线形,据此可对桥梁监测点所处部位的高程变化进行趋势预测,找出高程变化较大的监测点进行预警,为桥梁养护提供依据。8期数据分别按年份排序,可形成时间序列数据,满足递归神经网络的应用条件。

3.2 试验设计

根据试验目标需求,输出层节点数确定为25。因白沙洲大桥各安全监测点沿桥面下游侧均匀分布,各相邻点位监测数据具有一定的关联性,输入层节点数确定为75,隐层确定为250,远大于输入层节点数,以使递归神经网络模型的隐层学习25个特征之间更多的交叉组合,充分学习各特征之间的复杂关系。隐层之间的激励函数采用sigmoid函数。基于随机选值的思想,结合统计学原理,对算法的初始权值进行选定,采用sigmoid函数,实际获得的输出值范围为 $[0,1]$ 。

除输入、输出层节点数及激励函数外,还需确定以下参数:1)学习速率,即每次迭代的权重调整比例。学习速率由程序员选择,用于限制每次更新时被修正系数的数量,决定参数每次更新的幅度。学

习率太高,可能导致参数在极值的两侧来回移动而无法收敛;学习速率过小,虽然能保证收敛,但会大大降低优化速度。2)最大训练次数(epochs)。前向传播、损失计算、反向传播、参数更新过程也称为一个epoch,即更新系数的同时运行训练集的次数。每次epoch,都会更新数据集里每行系数。3)迭代目标,即网络训练误差精度。每次epoch都会更新系数,系数的更新基于模型生成的误差。该误差被算作候选系数的预测值和预期输出值之间的差。

针对以上3个试验参数,利用控制变量法进行试验选取,其中每个参数设定3个数值,通过对8期样本数据中几组进行预测确定一组最优参数。方法为:通过前3期数据预测第4期数据,以此类推,利用第8期数据作为检验;通过第5、6、7期数据对第8期数据进行预测,根据预测结果分析模型的准确性。其中,试验网络含75个输入节点、250个隐层节点及25个输出节点。以误差系数评价试验结果,方法为:1)将训练输出的25个数据与实际25个数据进行反归一化,即将向量中各归一化元素乘以该输出向量的模。2)求训练输出的25个数据与实际25个数据相对应差的平方和。3)将步骤2中求得的平方和除以25,得误差系数 μ 。误差系数越小,训练效果越好;反之,训练效果越差。表1为部分参数的训练结果。

表1 不同参数的训练结果

学习速率	训练次数/次	迭代目标	训练时间/s	误差系数 μ
0.100	4 000	10^{-3}	2.84	2.125 7
0.100	4 000	10^{-4}	*	*
0.010	4 000	10^{-3}	1.327	0.838 1
0.010	4 000	10^{-4}	1.806	0.193 4
0.010	4 000	10^{-5}	354.837	0.018 0
0.001	4 000	10^{-3}	8.707	0.996 2
0.001	4 000	10^{-4}	10.092	0.155 2
0.001	4 000	10^{-5}	11.067	0.011 4

注:*表示训练失败,主要原因为学习速率较大而迭代的目标误差要求较低,导致模型无法收敛。

通过对表1中不同参数组合的试验结果分析,选定学习速率为0.001、最大迭代次数为4 000次、迭代目标为 10^{-5} 的参数组合,其训练误差系数最低,且训练时间适中。

3.3 模型验证

采用确定的参数组合进行试验,通过对两个目

标函数 $y = \sin x$ 和 $y = x \sin x$ 进行趋势预测,对模型进行验证。方法为:按照 x 等距离递增顺序分别输入 280 个 $\sin x$ 和 $x \sin x$ 的函数值,以前 75 个作为训练样本,通过模型训练预测后续数据,并与原样本数据进行对比。训练结果见表 2、图 4、图 5。

表 2 递归神经网络模型验证结果

函数名称	训练次数/次	训练时长/s	误差系数	准确率/%
$\sin x$	271	10.644	0.000 696	99.5
$x \sin x$	288	12.214	0.034 136	99.0

注:准确率计算方法为统计预测值和相应目标值单调性相同的点,将其总数除以全部预测值个数。

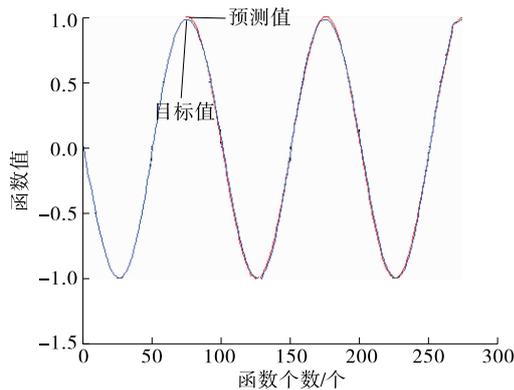


图 4 函数 $\sin x$ 预测结果

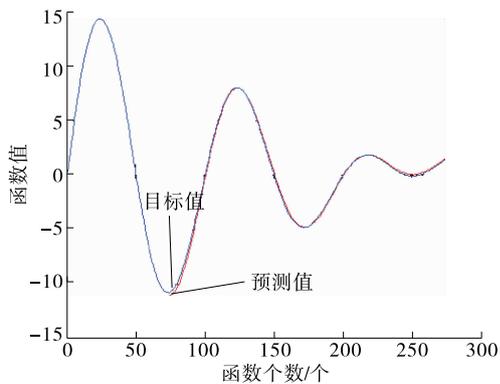


图 5 函数 $x \sin x$ 预测结果

根据验证结果,模型训练结果较好,训练时间短,误差系数小,且未发生欠拟合现象,说明局部加权线性回归方法可行。

3.4 预测结果及分析

以第 5、6、7 期样本数据作为模型训练的输入,预测第 8 期桥梁监测点的高程,并将模型训练的输出结果与实际样本数据进行对比,分析递归神经网络在桥梁变形趋势预测中的适用性。图 6 为模型训

练的 Error-Cost 曲线,图 7 为模型训练结果与实际数据对比,图 8 为预测误差。模型的训练曲线为平滑的下降曲线,说明模型训练的收敛性较好,训练效果较理想。误差系数为 0.001,预测结果较准确。

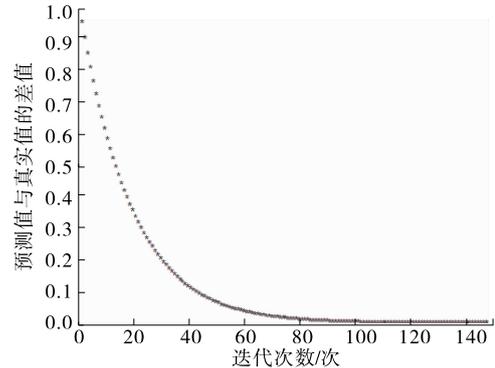


图 6 递归神经网络模型训练的 Error-Cost 曲线

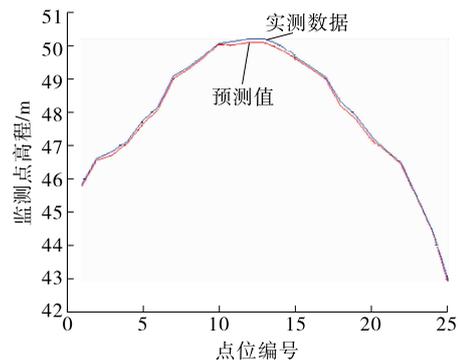


图 7 递归神经网络模型训练结果

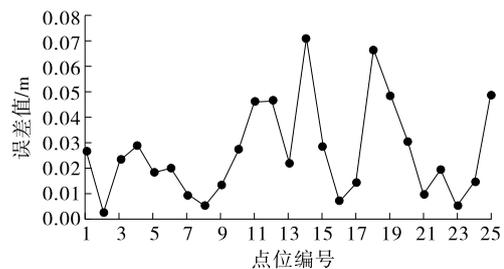


图 8 递归神经网络模型的预测误差

综上,递归神经网络在桥梁变形趋势预测方面具有较好的适用性。在训练时间和精度方面,不同的学习速率,对训练所需时间影响较小,但对训练结果的误差精度影响较大。学习速率较大时,训练时误差的减小速度会加快,但学习速率过大时误差反而会变大,导致误差曲线出现振荡乃至发散。因此,并非学习速率越大训练效果越好。当学习速率较大、迭代目标较小时,往往需牺牲训练时间从而获得较好的训练结果。因此,在试验过程中需平衡学习

速率与迭代目标之间的关系,在训练效果好的前提下尽量减少训练时间。

4 结语

该文利用递归神经网络的循环连接特性,通过对桥梁变形趋势预测对桥梁健康状况进行评估。工程实例应用分析结果表明,递归神经网络在桥梁变形趋势预测、隐患发掘及健康状况评估方面具有较大优势,通过预测使桥梁的管养集“防、管、控”为一体,使桥梁的健康评估既科学又便捷。该方法可推广应用用于滑坡、地面塌陷等不良地质体的安全监测,具有一定参考价值和实践意义。

参考文献:

[1] 任宝双,钱嫁茹,聂建国,等.在用钢筋混凝土简支梁式桥结构综合评估方法[J].土木工程学报,2002,35(2).

[2] 袁海庆,杨燕,范剑锋,等.模糊层次分析法在桥梁综合评估中的应用[J].武汉理工大学学报,2005,29(6).

[3] 马亚丽,王东伟,张爱林.在役桥梁结构健康等级的多级模糊综合评判[J].北京工业大学学报,2005,31(1).

[4] Han H, Qiao J. A self-organizing fuzzy-neural network based on growing and pruning algorithm [J]. IEEE Transactions on Fuzzy Systems, 2010, 18(6).

[5] Wang X G, Tang Z, Tamura H, et al. An improved back-propagation algorithm to avoid the local minima problem [J]. Neurocomputing, 2004, 56.

[6] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[A]. Advances in Neural Information Processing Systems[C]. 2014.

[7] 徐丽娜.神经网络控制[M].北京:电子工业出版社,2003.

[8] Hinton G E, Osinde Ro S, Teh Y. A fast learning algorithm for deep belief nets [J]. Neural Computation, 2006, 18(7).

[9] 李克强.基于 Spark 的大规模 RNNLM 系统[D].镇江:江苏大学,2016.

[10] 冯秀芳,肖文炳.神经网络的数据分类算法在物联网中的应用[J].计算机技术与发展,2012,22(8).

[11] 周惠芳,王迎旭,陈文明.一种改进的 BP 神经网络[J].自动化技术与应用,2013,32(9).

[12] 彭晓波,桂卫华.动态 T-S 递归神经网络及其应用[J].湖南工业大学学报,2011,25(3).

[13] 丛爽,陆婷婷,戴谊.递归神经网络的设计与应用[J].计算机工程与应用,2009,4(1).

[14] 陈启丽.递归神经网络结构设计方法及应用研究[D].北京:北京工业大学,2014.

收稿日期:2017-12-10

(上接第 159 页)

续表 1

里程桩号	加载过程	荷载/kN	位移/mm	被测锚杆状态	卸载后锚杆状态
K167+780	40%	72	—		
	70%	126	—	锚杆与钢拱架未焊接;加载至设计	
	90%	162	—	荷载 40%时,油压表急剧变化,锚	锚杆被拔出
	100%	180	—	杆被拔出	
	卸载	0	—		

现场拉拔试验结果显示:K167+1560 处锚杆由于施工时点焊在钢拱架上,加载至设计荷载 90%时,锚杆与拱架焊点分离,油压表急剧变化,位移量急剧变化,锚杆被拔松动,再次加压无法至设计荷载;K167+1350 处锚杆施工时被焊接在钢拱架上,加载至设计荷载 100%时,衬砌局部剥落,锚杆未见明显位移;K167+780 处锚杆与钢拱架未焊接,加载至设计荷载 40%时,油压表急剧变化,锚杆被拔出。据此判断 K167+1560、K167+780 处锚杆施工质量未达到设计要求。

参考文献:

[1] CECS 22:2005 岩土锚杆(索)技术规程[S].

[2] 柯玉军.锚杆检测技术研究及应用[D].兰州:兰州大学,2006.

[3] GB 50007-2011,建筑地基基础设计规范[S].

[4] 郑卫华,邓超.锚杆拉拔垫板[P].中国专利:ZL2013 2 0405258.0.

[5] 郑卫华,张春晓,邓超,等.锚杆锚索拉拔仪[P].中国专利:ZL2015 2 0319871.X.

收稿日期:2017-08-17