软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2019,30(12):3665−3682 [doi: 10.13328/j.cnki.jos.005640]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于深度置信网络的广告点击率预估的优化*

陈杰浩, 张 钦, 王树良, 史继筠, 赵子芊

(北京理工大学 计算机学院,北京 100081)

通讯作者: 陈杰浩, E-mail: cjh@bit.edu.cn

摘 要: 随着互联网广告的飞速发展,如何预测目标用户对互联网广告的点击率(click-through rate,简称 CTR),成为精确广告推荐投放的关键技术,并成为计算广告领域的研究热点和深度神经网络的应用热点.为了提高广告点击率预估的精确度,提出了基于深度置信网络的广告点击率预估模型,并通过基于 Kaggle 数据挖掘平台数据集的 1 000 万条随机数据的实验,研究不同的隐藏层层数和隐含节点数目对预测结果的影响.为了解决深度置信网络在数据规模较大的工业界解决方案中的训练效率问题,通过实验证明:广告点击率预估中,深度置信网络的损失函数存在大量的驻点,并且这些驻点对网络训练效率有极大的影响.为了提高模型效率,从发掘网络损失函数特性入手,进一步提出了基于随机梯度下降算法和改进型粒子群算法的融合算法,以优化网络训练.融合算法在迭代步长小于阈值时可以跳出驻点平面,继续正常迭代.实验结果表明,与传统的基于梯度提升决策树和逻辑回归的广告点击率预估模型以及模糊深度神经网络模型相比,基于深度置信网络的预估模型具有更好的预估精度,在均方误差、曲线下面积和对数损失函数指标上分别提升 2.39%,9.70%,2.46%和 1.24%,7.61%,1.30%;使用融合方法训练深度置信网络,训练效率提高 30%~70%.

关键词: 广告点击率预估;深度置信网络;驻点;粒子群算法;融合算法

中图法分类号: TP18

中文引用格式: 陈杰浩,张钦,王树良,史继筠,赵子芊.基于深度置信网络的广告点击率预估的优化.软件学报,2019,30(12):3665−3682. http://www.jos.org.cn/1000-9825/5640.htm

英文引用格式: Chen JH, Zhang Q, Wang SL, Shi JY, Zhao ZQ. Click-through rate prediction based on deep belief nets and its optimization. Ruan Jian Xue Bao/Journal of Software, 2019,30(12):3665−3682 (in Chinese). http://www.jos.org.cn/1000-9825/5640.htm

Click-through Rate Prediction Based on Deep Belief Nets and Its Optimization

CHEN Jie-Hao, ZHANG Qin, WANG Shu-Liang, SHI Ji-Yun, ZHAO Zi-Qian

(School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China)

Abstract: With the rapid development of Internet advertising, how to predict the target user's click-through rate of Internet advertisement has become a key technology for accurate advertising and has become a hot topic in the field of computational advertising and the application of deep neural networks. To improve the accuracy of CTR (click-through rate) prediction, this work proposed a prediction model based on deep belief nets and studied the influence of the number of hidden layers and the number of units in each layer on prediction results by taking experiments on the 10 million samples in the dataset provided by Kaggle Data Mining platform. In order to solve the problem of training efficiency of deep belief nets in large-scale industrial solutions, this study took wide experiments to prove that there are a lot of stagnation points in the loss function of deep belief nets and it has great negative effect on the training process. To improve the efficiency of training, starting from the characteristics of network loss function, this study further proposed a network optimization fusion model based on stochastic gradient descent algorithm and improved particle swarm optimization algorithm. The fusion algorithm can jump out of the stagnation ground and continue the normal training process. The experiment results show that

* 收稿时间: 2018-06-22; 修改时间: 2018-08-10; 采用时间: 2018-09-01

compared with the traditional prediction model based on gradient boost regression tree and logistic regression, and the deep learning model based on fuzzy deep neural network, the proposed training model has better accuracy in prediction and performs 2.39%, 9.70%, 2.46% and 1.24%, 7.61%, 1.30% better in mean squared error, area under curves, and LogLoss. The fusion method will improve the training efficiency of deep belief nets at the level of 30%~70%.

Key words: click-through rate prediction; deep belief net; stagnation point; particle swarm algorithm; fusion algorithm

近年来,互联网广告已成为大部分互联网公司的主要盈利手段,得到了极大的发展.为了达到最佳互联网广告投放效果,广告点击率(click-through rate,简称 CTR)预估成为了计算广告(computational advertising)[1]领域工业界和学术界研究的焦点.

广告精确投放,依赖于预测目标受众对相应广告的 CTR[2].例如,搜索广告为互联网广告的重要组成部分,该类广告依据受众搜索关键字的相关性进行广告投放.广告投放的变现能力,即广告此次投放后的期望收益,是决定广告主是否愿意投放此次广告的根本依据.千次有效点击(effective cost per mile,简称 eCPM)是期望收益的重要指标,其公式为

$$eCPM=\mu(a,u,c)\times v(a,u,c) \tag{1}$$

其中,$eCPM$ 为点击的变现能力,即期望收益;μ为广告 CTR;v 为点击的期望收益.由公式可知,广告的变现能力由广告 CTR 和点击期望收益决定.由于点击期望收益与产品本身有关,因此对于广告投放平台和广告主,精确的广告 CTR 预估成为了决定广告变现能力的关键.

为提高 CTR 预估的精确度,本文将深度置信网络(deep belief net,简称 DBN)应用于 CTR 预估领域,进行了大量实证研究,并提出了有效的优化策略.本文的主要贡献总结如下.

(1) 在 CTR 预估问题中引入 DBN,设计了模型结构及训练方法,通过实验探讨了不同的隐藏层层数、隐含节点数目对预测结果的影响.

(2) 将本文模型与基于梯度提升决策树(gradient boost decision tree,简称 GBDT)和逻辑回归(logistics regression,简称 LR)[3]的传统模型以及模糊深度神经网络(fuzzy deep neural network,简称 FDNN)的深度模型进行对比.实验结果表明,基于 DBN 的预估方法相比现有的 CTR 预估算法具有更好的预估精度,在均方误差、曲线下面积和对数损失函数指标上优于 GBDT+LR 模型 2.39%,9.70%和 2.46%,优于 FDNN 模型 1.24%,7.61%和 1.30%.

(3) 为了解决 DBN 在大数据规模较大的工业界解决方案中的训练效率问题,本文通过实验证明:CTR 预估中 DBN 的损失函数存在大量的驻点,并且这些驻点对网络训练效率有极大的影响.

(4) 为了进一步提高模型效率,从发掘网络损失函数特性入手,提出了基于随机梯度下降算法(stochastic gradient descent,简称 SGD)和改进型粒子群算法(particle swarm optimization,简称 PSO)[4,5]的融合算法来优化网络训练.最终,融合算法在迭代步长小于阈值时可以跳出驻点平面,继续正常迭代.实验证明,使用融合方法训练深度置信网络训练效率可提高 30%~70%.

本文第 1 节介绍优化深度神经网络(deep neural network,简称 DNN)训练效率的两种主要思路,并结合文献进行了分析.第 2 节提出基于 DBN 的 CTR 预估模型,并给出了训练方式.第 3 节设计实验取得模型参数最佳取值,验证模型 CTR 的预估效果.第 4 节根据训练过程讨论驻点对 DBN 训练的影响,针对驻点的特性,提出基于 SGD 和 PSO 的训练优化算法,并通过实验证明训练优化策略的有效性,进行结果分析.最后总结全文,并对未来值得关注的研究方向进行初步探讨.

1 相关工作

1.1 广告点击率预估

成熟的 CTR 预估模型常采用机器学习的方法.Chapelle 等人[6]提出了基于 LR 的 CTR 预估机器学习框架,主要解决雅虎的 CTR 预估问题,其采用了 4 组特征(包含类别特征和连续特征)作为模型输入:广告主特征、网

页出版商特征、用户特征和时间特征.该框架在 LR 模型的基础上加入了参数的二范数正则化项,该方法可以产生更稀疏的模型,使得非零参数增加避免过拟合的问题;本文还提出了针对稀疏数据的哈希方法,通过哈希函数使得原本很稀疏的数据映射到一个固定长度的数据空间中.Facebook[3]于 2014 年提出了基于 GBDT 方法,针对其广告系统进行 CTR 预估研究,利用用户的自身信息以及网页信息等作为特征,由决策树模型进行模型训练,得到的输出结果是一个固定维度的二值向量;而后,使用决策树模型的输出结果作为 LR 模型的输入重新进行模型训练,并根据误差进行权值的调整.这种模型融合的方法结合了非线性决策树模型能够拟合非线性特征的特点,以及线性 LR 模型优秀的扩展性、模型训练速度快的特点.

现有的 CTR 预估模型发展比较完备,在经过大量的训练后均可达到较好的预估效果,但需要人工投入大量精力进行特征的选取、处理与构造工作,广告数据特征提取不充分,不同特征之间的非线性关联无法得到充分体现,其预测结果准确程度多与数据分析人员经验丰富程度正相关.深度学习具有多层非线性映射的深层结构,可以完成复杂的函数逼近,近年来,在计算广告领域成为研究热点[7,8].Zhang 等人[9]提出一种基于循环神经网络(recurrent neural networks,简称 RNN)的新型 CTR 预估模型.与传统模型相比,该模型直接对用户行为顺序进行建模,优于 LR 模型 1.11%~3.35%.Chen 等人[10]提出了基于长短时记忆模型(long-short term memory,简称 LSTM)的 RNN 模型,该模型优化了 RNN 的梯度问题,预测结果优于 LR 模型 5%.Jiang 等人[11]提出了 FDNN 模型,该算法基于模糊受限玻尔兹曼机(fuzzy restricted Boltzman machine,简称 FRBM)和高斯-伯努利受限玻尔兹曼机(Gaussian-Bernoulli restricted Boltzmann machine,简称 GBRBM),从原始数据集自动提取隐藏的解释性信息,放大长尾重要信息,削弱无关信息,预测结果优于 LR 模型 3.25%.

然而,RNN 模型应用于 CTR 领域时,需基于用户或广告的历史行为进行时序性建模,现有的广告数据集大多不包含该类特征,并且在实际工业运用时采集难度大.

1.2 深度神经网络训练效率优化算法

由于 DNN 模型的并行性能相对较差,在训练数据规模较大时,其训练效率是影响网络性能的关键.目前,对网络训练效率的研究主要集中于设计更出色的通用优化算法和发掘网络损失函数特性,并根据特性寻求特定优化算法两个方面.这些方法在 DBN 的训练中同样适用.

1.2.1 通用优化算法

在通用优化算法[12]方面,Rumelhart 等人[13]提出了基于动量(momentum)的优化算法,期望使用前一次迭代的信息改进当前迭代的步长; Sutskever 等人[14]基于 Nesterov 的工作提出 Nesterov Momentum 算法,该算法考虑并尝试解决了 Momentum 算法迭代过程中的震荡问题;Gabriel 等人[15]则提出了自适应梯度(adagrad)算法,该算法通过历史梯度调整学习率,使学习率跟随梯度变化变化,从而达到调整步长的目的.

1.2.2 针对损失函数特性的优化算法

针对损失函数特性的优化算法主要挖掘损失函数特性并针对性地进行优化[16].在高维空间中,存在各方向梯度均为 0 的点,被称为驻点.驻点分为局部极值点与鞍点.近年来,很多研究人员认为,损失函数的局部极值点是影响 DNN 训练效率的最大原因.Lo 等人[17]对多层感知器中局部极值点的影响进行了研究;Chen 等人[18]提出了随机模糊向后传播(random fuzzy back-propagation,简称 RFBP)学习算法,使得迭代时能够逃离局部值.

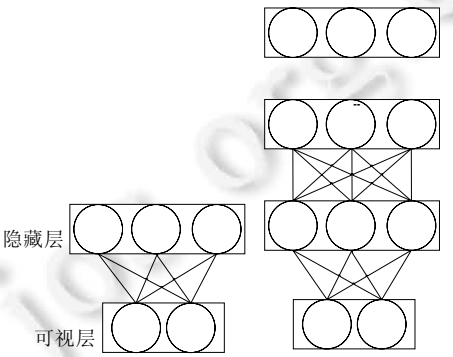
然而,最新研究证明:在实际的高维问题中,局部极小值相对数量较少[18,19]:Leonard 等人[19]发现,在某些神经网络的损失函数中不存在局部极小值;Daniel 等人[20]研究表明,高维神经网络中存在一些局部极小值点,但他们的质量相差不大(即通过不同极值点对应的神经网络得到的最终预测正确率相差不大),并且通用优化算法(如 momentum 等)的效果优劣并不明确,甚至有时不如最基本的参数调优算法.

目前,继续挖掘特定 DNN 模型的损失函数特性、寻找对其训练效率影响最大的因素并提出解决方案,成为 DNN 训练优化策略的研究热点.

2 基于深度置信网络的广告点击率预估模型

2.1 深度置信网络基本原理

DBN 由多个受限玻尔兹曼机(restricted Boltzmann machine,简称 RBM)堆叠而成.RBM 是一个两层的无向图模型,结构如图 1 所示.RBM 可以被用来作为构建 DBN 的组成部分,它包括一层可视单元和一层隐藏单元,可视单元与隐藏单元之间任意连接,但可视单元、隐藏单元自身没有连接关系.RBM 中的可见层和隐层单元可以是二值节点,也可以是任意指数族单元,例如高斯单元、泊松单元、softmax 单元等.



(a) 受限玻尔兹曼机 (b) 深度置信网络

Fig.1 Structure of RBMs

图 1 RBM 的结构

DBN 最顶的两层之间是无向连接,称为联想记忆层(associative memory);而其余层与层之间均为有向连接,分为向下的认知权重和向上的生成权重.其结构如图 2 所示.

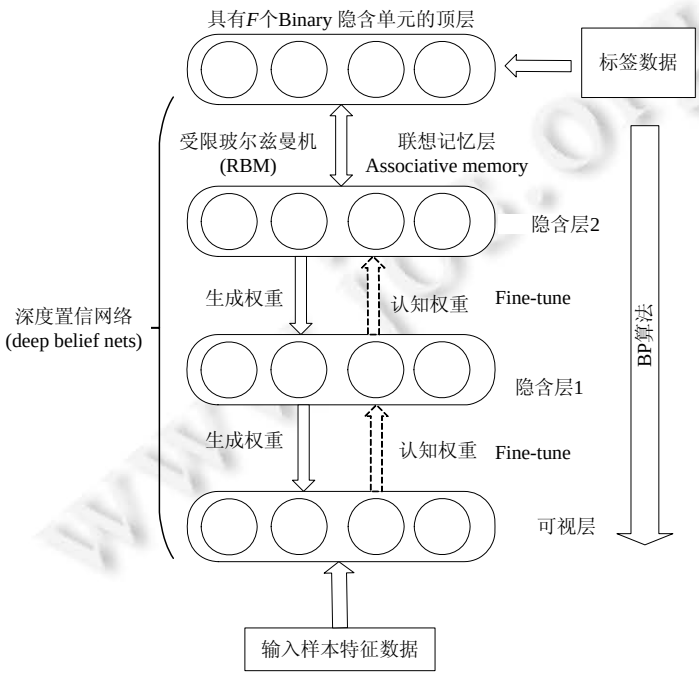


Fig.2 Structure of DBN

图 2 DBN 结构

本文的研究中,对 DBN 的学习和训练包括两个部分.

(1) 对模型进行无监督的预训练.将 DBN 每两层看作一个 RBM 模型,下层 RBM 的隐藏层作为上层的可视层,依次进行训练.最终训练获得的参数作为 DBN 的初始化参数.

(2) 使用 Wake-sleep 算法,无监督地对预训练之后的模型参数进行调优.除了顶端的两层为联想记忆层之外,网络中其他层之间的连接均为有向连接,分为自底向上的认知权重和自顶向下的识别权重.调优的过程分为“wake”和“sleep”两个阶段.

- “wake”阶段是一个识别事物的过程,使用认知权重将底层的输入样本数据自底向上的抽象为各层的隐含特征.在此过程中,通过对比散度差算法对生成权重进行调整.在联想记忆层中进行 T 步吉布斯采样,并对联想记忆层的权重进行调整.
- “sleep”阶段是生成样本数据的过程,使用生成权重自顶向下利用各隐含层抽象出的隐藏特征对输入样本数据进行生成重构.

为了避免过拟合,在本文使用的 DBN 的训练过程中,采用权值衰减策略(weight-decay).模型各节点间的权重更新方法如公式(2)所示.

$$\Delta w_{ij}=\varepsilon(\langle v_i h_j\rangle_{data}-\langle v_i h_j\rangle_T-\lambda\cdot w_{ij}(t-1))+\alpha\cdot\Delta w_{ij}(t-1) \tag{2}$$

其中,$\Delta w_{ij}(t-1)$为上一次权重的更新值.模型训练参数见表 1.

Table 1 Training parameters

表 1 训练参数

参数名称	参数设置
模型输入层节点	22
模型输出层节点	1
学习率	ε=0.02
动量学习率	α=0.5
权损失值(weight-cost)	λ=0.001
迭代周期	150
节点激活函数	sigmoid 函数
权重初始化	高斯分布 $N(0,0.01^2)$
节点偏置值初始化	0
RBM 中吉布斯采样步数	T=1

2.2 广告点击率预估模型及训练

本文采用的 CTR 预估模型为 DBN,共分为 1 个可视层、多个隐藏层和一个输出层.输出层使用 LR 模型进行输出,整体结构如图 3 所示.

CTR 预估模型训练主要分为以下 3 个阶段.

(1) 利用 DBN 进行深层次的有效特征提取,将降维后的特征数据作为模型的输入,通过 DBN 的认知权重,转化为深层次的抽象特征.

(2) 结合 LR 模型进行 CTR 预估.使用 DBN 模型的最顶层的隐藏层作为 LR 模型的输入;同时,在模型的顶层增加一层(包含 1 个节点)作为 LR 模型的输出;接着,对 LR 模型进行训练.

(3) 在对 LR 模型进行训练的同时,使用标签数据,对深度网络模型的参数进行有监督的调优.将整个网络模型(包括 DBN 和 LR 模型)看作是由向上的认知权重进行连接的标准前馈型神经网络,模型可视层输入样本数据进行前向传播,而后利用训练样本的标签数据计算模型输出误差,使用反向误差传播算法(error back propagation,简称 BP)进行误差传播对模型的参数进行进一步调优.

在整个 DBN 的模型中,各个节点的激活函数均采用 sigmoid 函数,以便引入非线性,并保证数据在传递的过程中不发散.经过处理后的样本特征数据从模型的可视层进入网络,经过多个隐藏层的抽象转化,在模型的顶层形成了深层抽象特征.最后,模型利用 DBN 提取得到的深层抽象特征作为 LR 模型的输入,加入标签信息,进行 CTR 的预测.模型的输出层包含一个 sigmoid 节点,其与 DBN 的顶层隐藏节点共同构成 LR 模型.

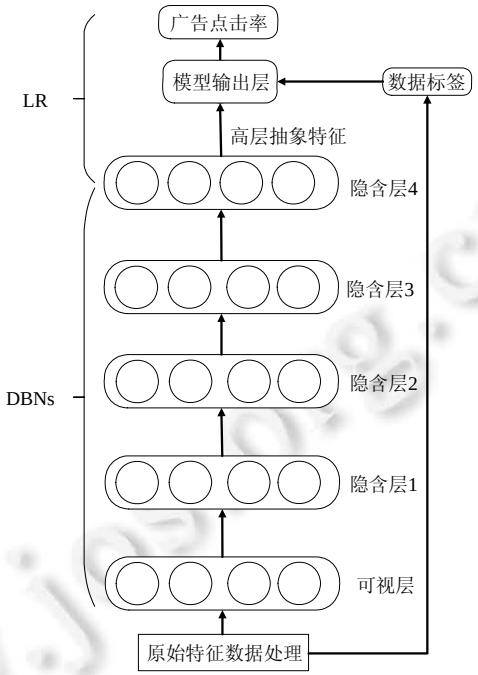


Fig.3 Structure of CTR prediction model

图 3 CTR 预估模型结构

模型输出层节点的激活概率如公式(3)所示.

$$p(Y=1|x,\theta)=\frac{1}{1+e^{-\theta^T x+b}} \tag{3}$$

其中,x 为 DBN 顶层节点状态,θ为模型输出层与 DBN 顶层之间的连接参数,而 b 则表示输出层节点的偏置.

$$E=-\frac{1}{N}\sum_{i=1}^{N}y_i\times\log(p_i)+(1-y_i)\times\log(1-p_i) \tag{4}$$

其中,N 为测试样本总数,y_i 为第 i 个样本的目标值(点击为 1,未点击为 0), $p_i=1/e^{-\theta x_i+b}$ 为第 i 个样本模型给出的估计值(预估 CTR).使用梯度下降算法可以求出参数θ和 b 的梯度,如公式(5)和公式(6)所示.

$$\frac{\partial E}{\partial\theta}=-\frac{1}{N}\sum\nolimits_{i=1}^{N}\frac{\partial E}{\partial p_i}\frac{\partial p_i}{\partial\theta}=\frac{1}{N}\sum\nolimits_{i=1}^{N}(p_i-y_i)x_i \tag{5}$$

$$\frac{\partial E}{\partial b}=-\frac{1}{N}\sum\nolimits_{i=1}^{N}\frac{\partial E}{\partial p_i}\frac{\partial p_i}{\partial b}=\frac{1}{N}\sum\nolimits_{i=1}^{N}(p_i-y_i) \tag{6}$$

3 广告点击率预估实验

3.1 数据集及衡量标准

本文进行的实验基于 Kaggle 数据挖掘比赛平台的 Click-Through Rate Prediction 比赛数据集.该数据集由 Avazu 提供,数据集描述见表 2.本文使用数据集中随机选取的 1 000 万条数据,并使用 10 折交叉法作为实验验证方法.

在本文实验中,针对整体模型 CTR 预估效果的指标为均方误差(mean squared error,简称 MSE)、曲线下面积(area under curve,简称 AUC)和对数损失函数(LogLoss).针对 DBN 训练效率的衡量指标为损失值(loss)、迭代次数(iterations)和训练时间.

Table 2 Description of dataset

表 2 数据集描述

字段	描述	类型	特征值数量
id	记录标识	离散型	–
click	0 未点击,1 点击	离散型	2
hour	发生时间	离散型	–
C1	匿名字段	离散型	7
banner_pos	Banner 位置	离散型	7
site_id	站点 id	离散型	2 865
site_domain	站点域名	离散型	3 394
site_category	站点类型	离散型	22
app_id	App 的 id	离散型	4 154
app_domain	App 的域名	离散型	287
app_category	App 类型	离散型	31
device_id	设备 id	离散型	368 962
device_ip	设备 ip	离散型	1 078 153
device_model	设备模式	离散型	6 098
device_type	设备类型	离散型	4
device_conn_type	设备通讯类型	离散型	4
C14~C21	匿名字段	多为离散型,少数连续型	–

3.2 广告点击率预估准确率实验

本节将首先针对 DBN 中的重要参数进行单独实验,考察不同的参数对预估结果的影响,获取算法最优的预测结果.实验中,使用 AUC 指标对不同模型的 CTR 预估结果进行评估,实验总共分为 3 个部分.

(1) 通过实验,探讨 DBN 隐藏层层数对 CTR 预估结果的影响.

(2) 通过实验,对模型逐层验证不同的隐藏节点数目对 CTR 预估结果的影响.

(3) 通过实验,研究基于 DBN 的 CTR 预估模型预估效果.

3.2.1 深度置信网络隐藏层数实验

深度模型的隐藏层的个数是模型预估能力关键因素,经验表明,模型的特征表征和提取能力在一定程度上与隐藏层层数正相关;同时,层数越多,深度模型训练过程越复杂.在实际应用中,通常隐藏层层数的设置要根据实际情况,通过实验确定.在本节中,对 DBN 中隐藏层的层数设定进行实验分析,在进行实验的时候,各隐藏层节点数目均设置为 256,逐步添加隐藏层的层数,最终得出的实验结果如图 4 所示.

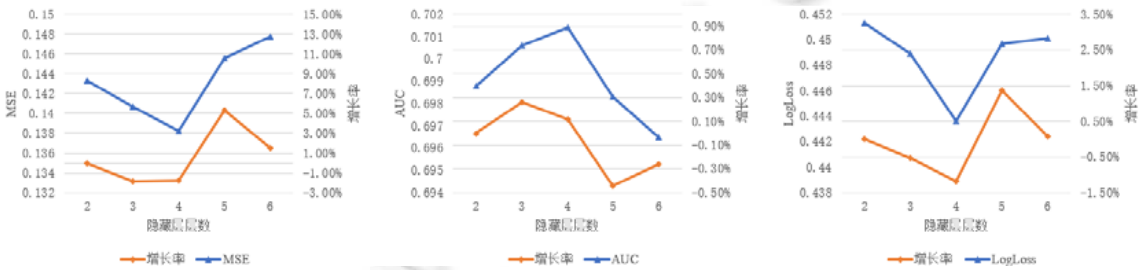


Fig.4 Curves of the MSE, AUC and LogLoss while training DBN with different hidden layers

图 4 用不同隐藏层数训练 DBN 的 MSE、AUC 和 LogLoss 曲线

随着神经网络隐藏层数目的增加,CTR 预估准确率出现了明显的上升,说明在利用深度模型进行深层次特征提取的时候,隐藏层的数目越多,对于输入层样本数据特征学习的就越充分,能够更好地反映数据深层的特征;而当隐藏层数目大于 4 层时,实验结果的准确率出现了较快的下滑,说明仅通过增加深度模型的隐藏层数量不能无限制地增加模型预测效果,且进行实验的过程中隐藏层数目过多,容易导致模型出现过拟合,因此需要通过实验来确定最合适的隐藏层数目.

3.2.2 深度置信网络隐藏层节点数实验

在本节中,主要考察 DBN 模型中各个隐藏层节点数目对于实验结果的影响.首先,考察第 1 层隐藏层对实验结果的影响,此时固定第 2 层~第 4 层隐藏层节点数为 256.实验结果如图 5 所示.

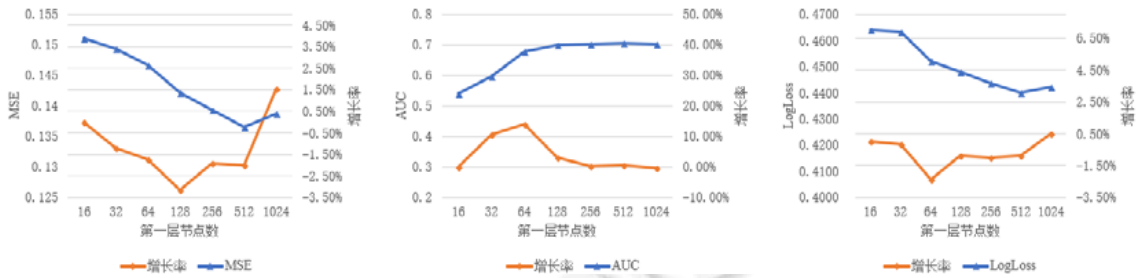


Fig.5 Curves of the MSE, AUC and LogLoss while training DBN with different units in the 1st hidden layer

图 5 用不同的隐藏层第 1 层节点数训练 DBN 的 MSE、AUC 和 LogLoss 曲线

在固定隐藏层数和其他层节点数的实验中,随着第 1 层隐藏层节点数的增加,实验结果的准确率逐渐升高.当节点数大于 512 时,整体模型同样出现过拟合的现象,因此 512 为最佳取值.此时,固定第 1 层节点数为最佳取值 512,其他层节点数为 256,进行第 2 层节点数的实验,其结果如图 6 所示.

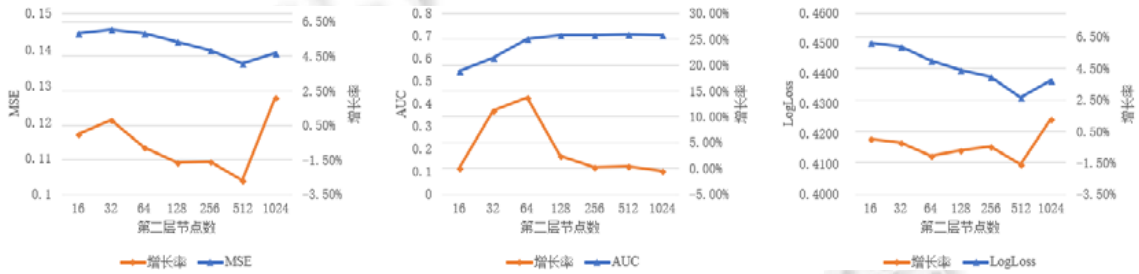


Fig.6 Curves of the MSE, AUC and LogLoss while training DBN with different units in the 2nd hidden layer

图 6 用不同的隐藏层第 2 层节点数训练 DBN 的 MSE、AUC 和 LogLoss 曲线

随着隐藏层节点数的增加,实验结果的准确率逐渐升高.当节点数目大于 512 时,模型同样出现了过拟合的现象,因此 512 为第 2 层节点数最佳取值.

同理可以得到第 3 层、第 4 层隐藏层节点数目对实验结果的影响,如图 7 和图 8 所示.可以看出,第 3 层、第 4 层隐藏层节点数目同样也应该设置为 512.

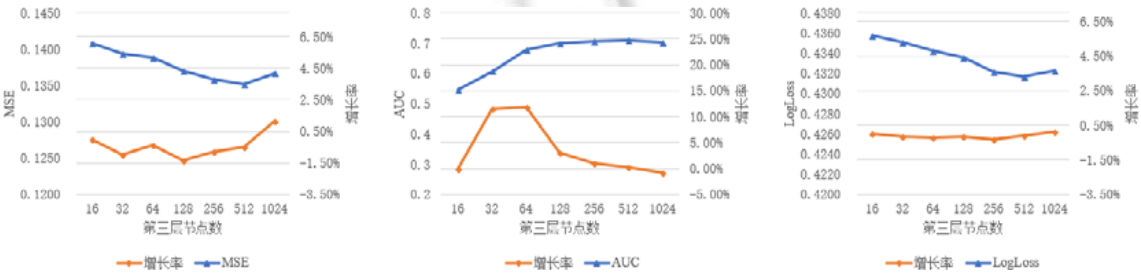


Fig.7 Curves of the MSE, AUC and LogLoss while training DBN with different units in the 2nd hidden layer

图 7 用不同的隐藏层第 2 层节点数训练 DBN 的 MSE、AUC 和 LogLoss 曲线

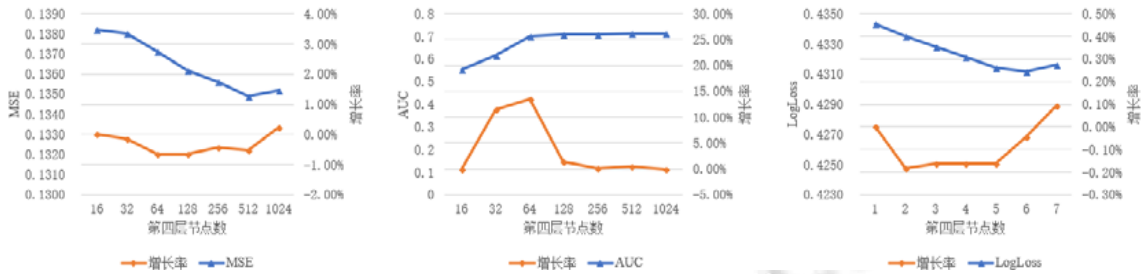


Fig.8 Curves of the MSE, AUC and LogLoss while training DBN with different units in the 2nd hidden layer

图 8 用不同的隐藏层第 2 层节点数训练 DBN 的 MSE、AUC 和 LogLoss 曲线

通过实验结果可以看出,随着隐藏层节点数目的增加,模型 CTR 预估能力出现了先增后减的现象.这是因为,一方面,随着隐藏层节点数目的增加,模型对于样本数据的隐含特征学习的更加充分;另一方面,如果隐藏层节点过多的话,容易致使特征学习的过于充分,极度放大不同的特征对预测结果的影响,使得模型对 CTR 的预测结果过于敏感和不稳定.

3.2.3 对比实验及结果

本节中进行了本文模型与 GBDT+LR 及 FDNN 模型的对比实验,其中,GBDT+LR 模型结构及训练过程参考文献[2],FDNN 模型结构及训练过程参考文献[6].

利用与第 3.2.2 节相似的参数调优方法,逐步固定某一参数进行调优,取得 GBDT+LR 以及 FDNN 模型的最优参数,见表 3.

Table 3 Bset parameters of GBDT+LR and FDNN

表 3 GBDT+LR 和 FDNN 模型的最优参数取值

模型	参数	取值
GBDT+LR	最大叶节点数量	10
	最大决策树棵数	30
FDNN	隐藏层层数	3
	第 1 层隐藏层节点数	170
	第 2 层隐藏层节点数	1 700
	第 3 层隐藏层节点数	17

最终对比实验结果如表 4 所示,本文提出的基于 DBN 的互联网广告 CTR 模型,CTR 预测效果分别在 MSE、AUC 和 LogLoss 指标上优于 GBDT+LR 的融合模型 2.39%,9.70%和 2.46%,优于 FDNN 模型 1.24%,7.61%和 1.30%.证明了 DBN 在隐含特征提取和抽象方面更加强大,提取出的深层次特征更能反映事物的本质,这也说明了本文设计的融合模型在互联网广告 CTR 预估的准确率方面达到了预期的实验效果.

Table 4 MSE, AUC and LogLoss of GBDT+LR, FDNN and DBN

表 4 GBDT+LR,FDNN 和 DBN 实验的 MSE、AUC 和 LogLoss 指标

模型	MSE	AUC	LogLoss
GBDT+LR	0.138 2	0.649 7	0.442 1
FDNN	0.136 6	0.662 3	0.436 9
DBN	0.134 9	0.712 7	0.431 2

3.3 分析与结论

本节分别进行了 DBN 隐藏层数实验、DBN 隐藏层节点数实验和融合模型与其他模型的对比实验.随着隐藏层层数和隐藏层节点数目的增加,模型对于样本数据的隐含特征学习的更加充分;隐藏层层数和隐藏层节点过多,容易致使特征学习的过于充分,极度放大不同的特征对预测结果的影响,使模型对 CTR 的预测结果过于敏感和不稳定.最终,基于 DBN 的模型 CTR 预测效果分别在 MSE、AUC 和 LogLoss 指标上优于 GBDT+LR 的融

合模型 2.39%,9.70%和 2.46%,优于 FDNN 模型 1.24%,7.61%和 1.30%.

4 深度置信网络训练优化策略

在将 DNN[21,22]用于训练数据规模较大的工业界解决方案时,常因其有限的并行性而受到训练效率的制约.作为 DNN 中的一种,DBN 存在着相同的问题.本节基于网络损失函数特性,发现在 DBN 的损失函数中存在大量严重影响训练效率的驻点.针对这些驻点的特性,提出了一种基于 SGD 和改进型 PSO 的网络训练优化算法.

4.1 驻点对深度置信网络训练的影响实验与结果

驻点包括局部极值点与鞍点.本节通过实验分别证明在 DBN 的损失函数平面中存在这两种点,以及他们对网络训练的影响.

4.1.1 极值点与鞍点判定方式

在数学中,Hessian 矩阵是一个多变量实值函数的二阶偏导数组成的方块矩阵,假设有一实数函数 $f(x_1,x_2,\ldots,x_n)$,若函数 f 的所有二阶偏导数都存在,则 f 的 Hessian 矩阵的第 ij 项为

$$H(f(x))_{ij}=D_iD_jf(x) \tag{7}$$

其中,$x=(x_1,x_2,\ldots,x_n)$,即

$$H(f)=\begin{bmatrix}\dfrac{\partial^2 f}{\partial x_1^2} & \dfrac{\partial^2 f}{\partial x_1\partial x_2} & \cdots & \dfrac{\partial^2 f}{\partial x_1\partial x_n}\\ \dfrac{\partial^2 f}{\partial x_2\partial x_1} & \dfrac{\partial^2 f}{\partial x_2^2} & \cdots & \dfrac{\partial^2 f}{\partial x_2\partial x_n}\\ \vdots & \vdots & \ddots & \vdots\\ \dfrac{\partial^2 f}{\partial x_n\partial x_1} & \dfrac{\partial^2 f}{\partial x_n\partial x_2} & \cdots & \dfrac{\partial^2 f}{\partial x_n^2}\end{bmatrix} \tag{8}$$

当 Hessian 矩阵为正定矩阵时,该点为极小值点;当 Hessian 矩阵为不定矩阵时,该点为鞍点.

4.1.2 局部极值点

局部极值点是指梯度在各方向上都为 0,并在邻域内取值最大或最小的点,在本文研究的问题中,为网络损失函数的极小值点,而网络训练的最终目的在于找到一个达到精度要求的极小值点.本节通过在梯度下降过程中使用不同训练算法和不同网络参数初始值,来对比不同训练过程和不同初始状态下网络的训练结果.其结果如图 9、图 10、表 5、表 6 所示.

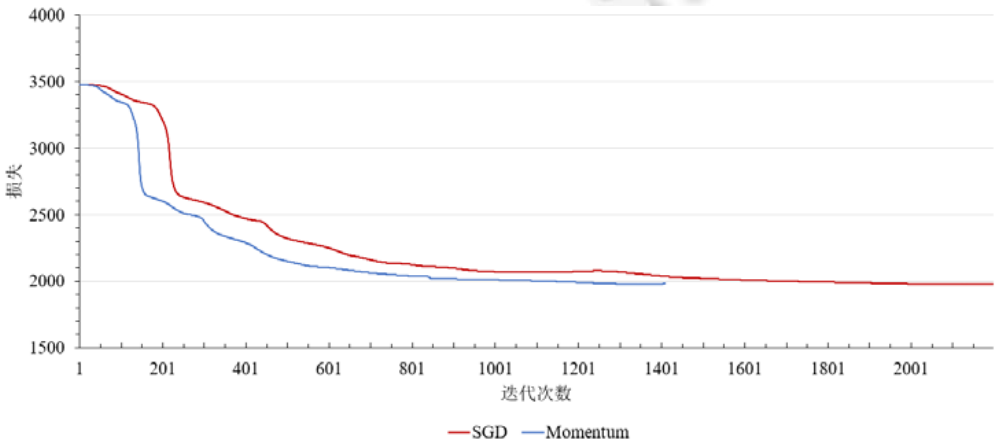


Fig.9 Curves of the loss while training DBN with SGD and Momentum

图 9 用 SGD 和 Momentum 训练 DBN 的损失曲线

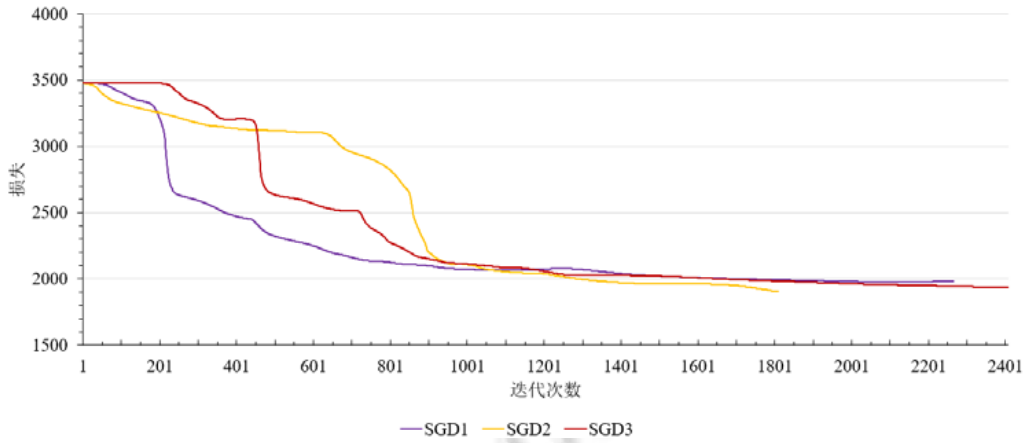


Fig.10 Curves of the loss while training DBN with SGD in different initial parameters

图 10 用 SGD 在不同初值条件下训练 DBN 的损失曲线

Table 5 Results while training DBN with SGD and Momentum

表 5 用 SGD 和 Momentum 训练 DBN 的结果

算法	迭代次数	时间(s)	Loss	MSE	AUC	LogLoss
SGD	2 194	3 330	1974.20	0.1352	0.7126	0.4312
Momentum	1 409	2 327	1978.00	0.1351	0.7123	0.4313

Table 6 Results while training DBN with SGD in different initial parameters

表 6 用 SGD 在不同初值条件下训练 DBN 的结果

算法	迭代次数	时间(s)	Loss	MSE	AUC	LogLoss
SGD 初值 1	2 194	3 330	1 974.20	0.135 2	0.712 6	0.431 2
SGD 初值 2	1 808	2 775	1 906.83	0.135 1	0.712 6	0.431 2
SGD 初值 3	2 404	3 553	1 977.70	0.135 1	0.712 5	0.431 2

由表 5、表 6 可知:不论迭代次数和训练时间如何,在最终收敛时其 MSE、AUC 和 LogLoss 指标相差均小于 1%.由此可得出结论:高维网络最终能够到达的极值点之间差异不大,只要能够找到一个可接受的极值点,网络训练都可以认为是成功的.

如图 11、图 12 所示的是用 SGD 和 Momentum 训练的 loss 曲线,图中字母处的小图代表着各字母点处步长的变化趋势.先减小后增加的步长趋势代表着训练陷入平缓平面并最终跳出.

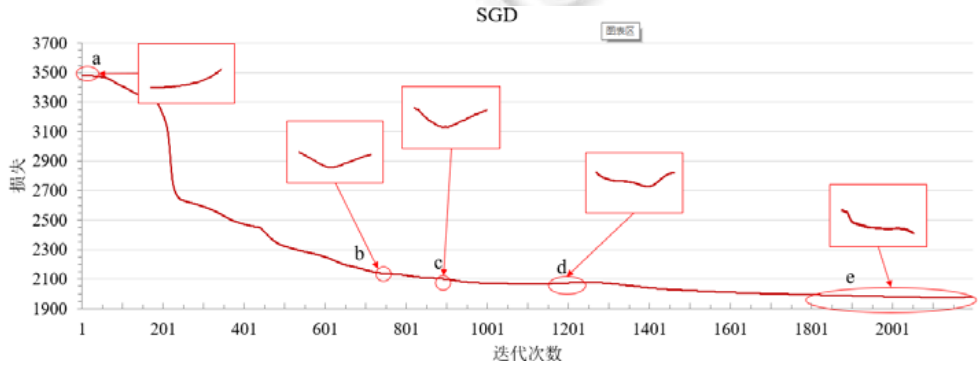


Fig.11 Curve of the loss while training DBN with SGD

图 11 用 SGD 训练 DBN 的损失曲线

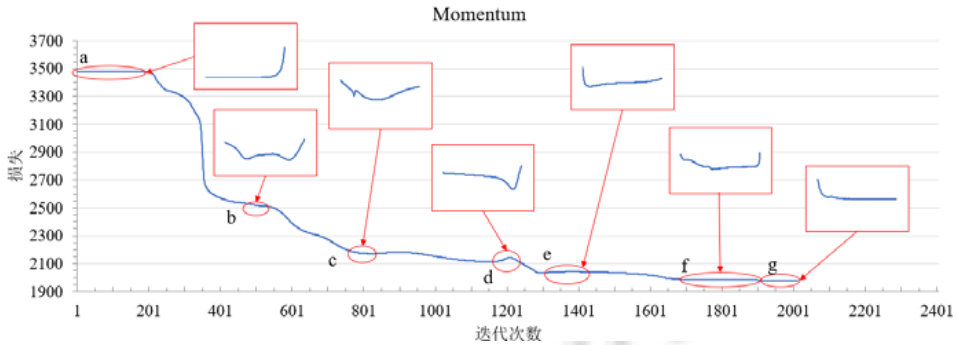


Fig.12 Curve of the loss while training DBN with Momentum

图 12 用 Momentum 训练 DBN 的损失曲线

两次实验中训练过程在极值点周围的平缓平面上停留的迭代次数分别达到总次数的37.95%和5%.如图11点 *e*、图 12 点 *g* 所示,证明了极值点对训练效率的影响.

4.1.3 鞍点

鞍点也是各方向梯度等于 0 的点,但它并非是邻域内的取值最小值点,即在某些方向上,鞍点可以继续训练下降.实验测试了使用 SGD 算法和 Momentum 算法时的损失函数值下降过程.实验结果表明:在使用 SGD 算法和 Momentum 算法进行训练的过程中,损失函数值的下降过程存在较明显的阶梯形态,即过程中会遇到许多鞍点区域.损失函数停留在这些区域内浪费了大量的时间[23−25].除图 11 点 *e*、图 12 点 *g* 外,其他点均为鞍点平面.

在使用SGD算法作为优化算法的实验中,在不算最后一次属于极值点平面的情况下,迭代次数中有24.18%在鞍点平面内,Momentum 算法中为 54.34%.

4.1.4 分析与结论

一方面,在CTR预估问题的高维网络中,收敛到的不同极值点对最终结果的影响不大,即不需要太过在意最终收敛解的质量.另一方面,驻点,即极值点与鞍点对网络训练的效率影响非常大,总体上,在使用 SGD 算法时有62.13%、使用 Momentum 算法时有 59.34%的迭代是在驻点平面内进行,这极大地拖累了网络训练的效率.

4.2 基于随机梯度下降和改进型粒子群的训练优化算法

4.2.1 融合方法流程

跳出驻点的判断标准依靠判断其 Hessian 矩阵的正定性,计算复杂度为 $O(n^3)$,且在驻点附近的驻点平台上训练同样会被拖慢.根据第 3 节的实验结果,当迭代步长小于 0.3 时,极有可能陷入驻点平台.为简化计算,以判断步长为激活条件,融合方法使用单次迭代效率较高的 SGD 算法作为基础算法,并在陷入驻点平台时激活 PSO 算法进行跳出.融合方法的伪代码如图 13 所示.算法流程图如图 14 所示.

初始化:Set *last_loss*=∞; *loss*=∞;
输入:特征向量;
输出:训练好的 DBN 模型和训练记录.
迭代:

```
while !StopOptimization(·) do
    NormalAlgorithm(·)
    loss←CalcLoss(·)
    if last_loss−loss<threshold then
        PSOAlgorithm(loss)
        loss←CalcLoss(·)
    end if
    last_loss←loss
end while
```

Fig.13 Pseudocode of fusion method

图 13 融合算法伪代码

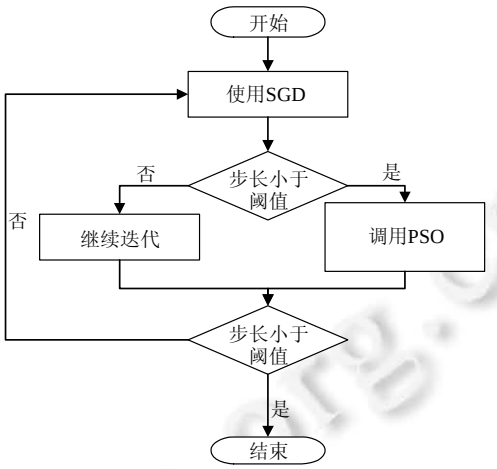


Fig.14 Flowchart of the fusion method

图 14 融合算法流程图

4.2.2 随机梯度下降算法

SGD 算法是最常用的 DNN 模型优化算法之一,发展成熟,存在较为完善的实现方式,且训练速度快,故融合算法使用 SGD 作为基础优化算法.与梯度下降算法(gradient descent)和批量梯度下降(batch gradient descent)算法不同的是:SGD 算法计算损失函数的每个参数的偏导数,并在每次迭代时使用单一样本进行梯度下降.SGD 的优化函数公式为

$$\theta_i = \theta_i - \alpha \frac{\partial}{\partial \theta_i} J(\theta) \tag{9}$$

其中,θ代表被优化的参数,$J(\theta)$为损失函数,α为学习率.当损失函数是方差时,其形式为

$$J(\theta) = \frac{1}{2}(h_\theta(x^{(j)}) - y^{(j)})^2 \tag{10}$$

对于θ_i有:

$$\theta_i = \theta_i - \alpha(h_\theta(x^{(j)}) - y^{(j)})x_i^{(j)} \tag{11}$$

其中,$x^{(j)}$代表数据集中第 j 个样本.

4.2.3 改进的粒子群算法

粒子群算法参数少,易于调整,并具有易实现、收敛快、应用灵活等优点,在函数优化、神经网络训练和模糊系统控制中均有良好的应用效果.本文根据驻点性质,改进了标准版本的 PSO 算法.引入 PSO 的目的不在于寻找局部极值点,而是跳出驻点平面.改进的 PSO 在初始化、迭代和退出判断等方面均与标准版本不同.

(1) 初始化

改进的 PSO 算法首先初始化初始粒子 P_0,其值等于 DBN 中需要被训练的参数值.接着,该算法引入冲量[11]的思想初始化一系列如标准 PSO 中的粒子,使用初始粒子 P_0 的值作为基准值,根据上一次迭代的方向和距离,将每个粒子值的方向设置为在各个维度内与上次迭代方向偏角不超过 45°的随机值,粒子与初始粒子的距离大小为与上一次迭代前进步长相关的随机值,公式如下:

$$Add_{ki}=rand(M)\times LastStep_{0i} \tag{12}$$

$$P_{ki}=P_{0i}+Add_{ki}+M\times\max(Add_{ki}) \tag{13}$$

其中,Add_{ki}是第 k 个粒子的第 i 个参数的增量,$rand(M)$是值域为$\lceil -M,M \rceil$的随机函数,$LastStep_{0i}$为上次迭代时第 i 个参数的步长,$\max(Add_{ki})$是属于第 k 个粒子的所有参数的最大值.

(2) 迭代

迭代过程中,与标准 PSO 算法相同,记录每个粒子的历史最优值和所有粒子的历史最优值.算法使用公式

(14)和公式(15)计算粒子下一个位置:

$$speed_{ki}=(w\times speed_{ki-1}+c1\times rand\times(SelfBest_{ki-1}-P_{ki-1})+c2\times rand\times(OveralBest-P_{ki-1})) \tag{14}$$

$$P_{ki}=P_{ki}+speed_{ki} \tag{15}$$

其中,$speed_{ki}$指第 k 个粒子第 i 个参数此次的迭代步长;w 即惯性因子,即此次迭代步长中保留了上一次迭代的惯性;$c1$,$c2$ 分别为个体学习率和全局学习率,分别乘上个体历史最优与当前位置的差,和全局最优与当前位置的差;$rand$ 是 0~1 的随机值.由于希望向外搜索而非向内搜索,本文将学习率设置为大于 1 的值,此处 $c1$,$c2$ 均设置为 2.

(3) 退出判断

由于本文中 PSO 算法并不需要收敛到一个极值点,而是向外找到一个驻点平面以外的点,考虑两种情况.

a) PSO 算法与梯度算法相比效率较低,为了确保融合算法的整体效率,需要为 PSO 算法的迭代次数设置上限,即如果在一定次数的迭代后算法仍然没有找到驻点平面外的点,算法被强行退出.根据实验经验,本模型中迭代次数上限为 10 次.

b) 如果在迭代次数上限之内算法找到了当前驻点平面以外的点,即算法找到的最优粒子的损失值与初始粒子 P_0 的损失值相差大于某个阈值,则算法成功运行并退出.根据实验经验,此处阈值设为 30.

4.2.4 Momentum 优化算法

Momentum 算法是基于动量的优化算法,核心思路在于在当前梯度上加入上一次迭代的动量,即

$$\Delta x_t=-\eta g_t+\rho\Delta x_{t-1} \tag{16}$$

其中,Δx_t为本次迭代步长,Δx_{t-1}为上次迭代步长,η为学习率,g_t为此次迭代梯度,ρ为上一次迭代的动量参数.本文使用 Momentum 优化算法作为对照组,考察融合算法的优化效果.

4.3 训练优化算法的实验与结果

学习率是一个重要的超参数,它控制着基于损失梯度调整神经网络权值的速度,大多数优化算法对它都有涉及.学习率越小,沿着损失梯度下降的速度越慢.从长远来看,这种谨慎慢行的选择比较保守,但可以避免错过任何局部最优解.然而,过小的学习率也意味着收敛速度变慢.本节将讨论不同学习率下训练优化算法的效果.

4.3.1 较小学习率实验

如图 15 所示为学习率 0.008(较小学习率)时,网络在使用 SGD 算法,Momentum 算法和新融合算法作为优化函数时损失函数值的下降过程.

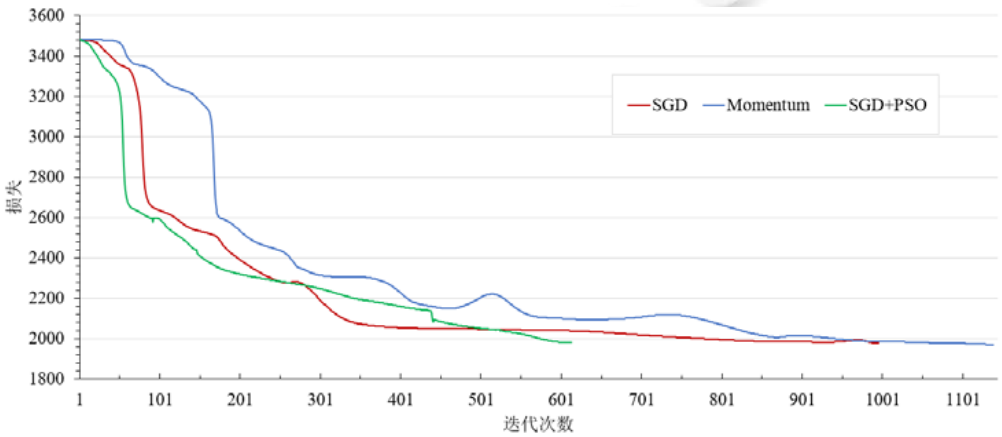


Fig.15 Curves of the loss while training DBN with SGD, Momentum and SGD+PSO (learning rate 0.008)

图 15 用 SGD,Momentum 及 SGD+PSO 训练 DBN 的损失曲线(学习率为 0.008)

从图中可知,不同算法训练 Loss 下降速度不同.SGD 算法与 SGD+PSO 算法 Loss 曲线存在两次交点,在迭

代次数小于第 1 次交点时,融合算法下降速度快;此后,单一 SGD 算法下降速度反超.但在 Loss 值逼近 2 000 时,单一 SGD 算法下降趋于平缓,陷入了驻点平面;而融合算法通过改进型 PSO 算法跳出平缓平面,从而继续正常下降.3 种算法的详细表现情况见表 7.

Table 7 Detailed result of training DBN with SGD, Momentum and SGD+PSO (learning rate 0.008)

表 7 用 SGD,Momentum 及 SGD+PSO 训练 DBN 的详细结果(学习率为 0.008)

算法	迭代次数	时间(s)	Loss	MSE	AUC	LogLoss
SGD	985	1 558	1 982.23	0.135 2	0.712 5	0.431 2
Momentum	1 139	2 043	1 969.85	0.135 1	0.712 6	0.431 1
SGD+PSO	614	1 182	1 981.29	0.135 1	0.712 5	0.431 2

如图 16 所示是在使用这 3 种算法时每次迭代损失函数下降幅度的 log 值.在相同的参数下,使用新算法进行优化时每次迭代下降幅度都可以控制在阈值以上.如图 8 所示,此处阈值为 0.3,则新算法下降幅度的 log 值均大于–0.5.这证明融合算法可以极大地减少网络需要迭代的次数和时间.具体地,使用了融合算法的优化过程迭代次数是仅使用 SGD 算法的 62.34%,迭代用时是仅使用 SGD 算法的 75.87%.

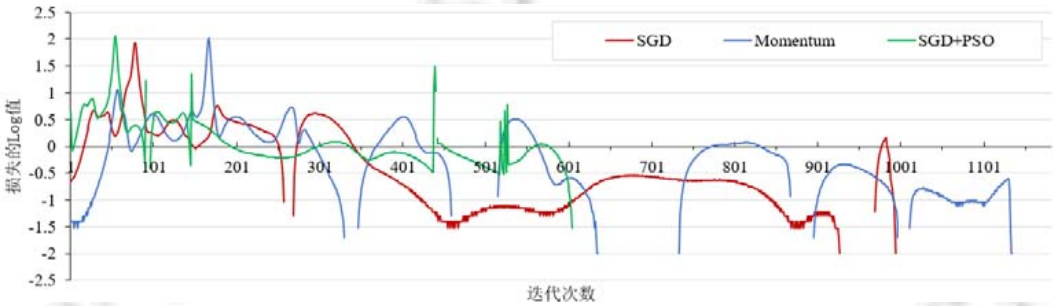


Fig.16 Curves of log(Δloss) while training DBN with SGD, Momentum and SGD+PSO (learning rate 0.008)

图 16 用 SGD,Momentum 及 SGD+PSO 训练 DBN 时每次迭代 loss 的 log 值(学习率为 0.008)

4.3.2 较大学习率实验

在学习率为 0.02(较大学习率)时,SGD,Momentum 和 SGD+PSO 算法的训练过程和结果如图 17 所示.

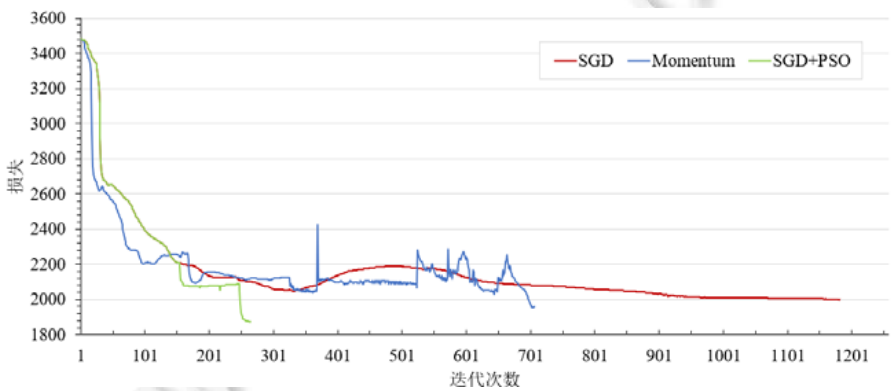


Fig.17 Curves of the loss while training DBN with SGD,Momentum and SGD+PSO (learning rate 0.02)

图 17 用 SGD,Momentum 及 SGD+PSO 训练 DBN 的损失曲线(学习率为 0.02)

如表 8 所示为 3 种算法的详细表现情况.在学习率较大的情况下,单一 SGD 算法和 Momentum 算法在迭代过程中都出现了 Loss 值升高的情况,而融合算法保持持续下降态势,证明了融合算法更适应激进的学习策略.融合算法的总迭代次数是单独使用 SGD 算法的 22.35%,运行总时间是 SGD 算法的 35.09%.同时,融合算法迭代

次数是单独使用 Momentum 算法的 37.15%,运行时间则缩短为 44.85%.

Table 8 Detailed result of training DBN with SGD, Momentum and SGD+PSO (learning rate 0.02)

表 8 用 SGD,Momentum 及 SGD+PSO 训练 DBN 的详细结果(学习率为 0.02)

算法	迭代次数	时间(s)	Loss	*MSE*	*AUC*	LogLoss
SGD	1 181	1 750	1 998.90	0.135 2	0.712 5	0.431 2
Momentum	705	1 369	1 955.61	0.135 1	0.712 6	0.431 1
SGD+PSO	264	614	1 871.80	0.135 1	0.712 5	0.431 2

如图 18 所示是 3 种算法每次迭代 loss 的 log 值,对比图 17 可以看出,明显出现了振荡现象.

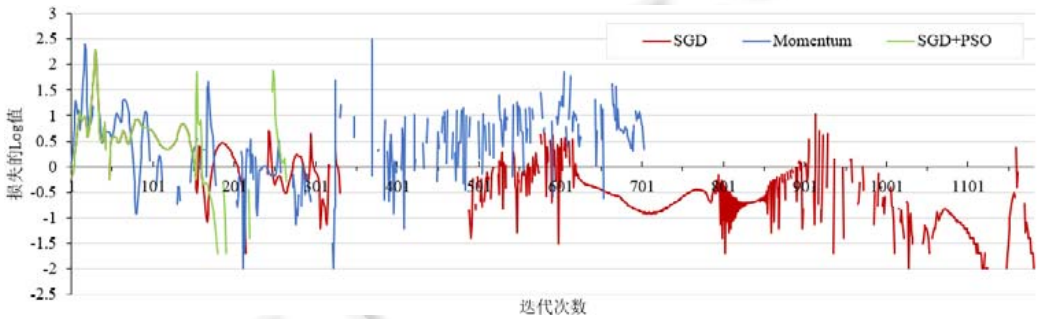


Fig.18 Curves of log(Δloss) while training DBN with SGD,Momentum and SGD+PSO (learning rate 0.02)

图 18 用 SGD,Momentum 及 SGD+PSO 训练 DBN 时每次迭代 loss 的 log 值(学习率为 0.02)

4.3.3 分析与结论

本文提出的 DBN 模型训练优化融合算法使用 SGD 作为基础优化算法,该算法是最常用的 DNN 模型优化算法之一,发展成熟,存在较为完善的实现方式,训练速度快.而作为跳出算法的 PSO 算法参数少,易于调整,并具有易实现、收敛快、应用灵活等优点,在函数优化、神经网络训练和模糊系统控制中均有良好的应用效果.针对驻点特性,在将 PSO 算法的初始化、迭代和退出判断进行改进之后,本文提出的融合算法在迭代步长小于阈值时激活改进型 PSO 算法跳出驻点平面,减少训练过程中的震荡,提升整体训练效率.

在较小学习率和较大学习率实验中,实验结果均证明了本文融合算法的有效性,算法效率在迭代次数和迭代时间方面均有较大提升.从图 17 中可以得到结论:当学习率较小,每次迭代时 loss 的下降幅度较小.相对地,图 18 中显示出较大的学习率会激化训练时的震荡.

学习率越大,改进型 PSO 算法的采用便越多,融合算法对训练的影响便越大.大的学习率也意味着 SGD 算法部分更加激进,在地形复杂程度较小的前中期,SGD 算法的优化效率自身就可以更高;而当到达地形较为复杂的空间,网络损失函数值的下降幅度一旦有减小的趋势,即训练效率有恶化的迹象时,较大的学习率会使得损失值下降幅度在更少的迭代次数内下降得足够小,甚至产生震荡,而这会导致改进型 PSO 算法的调用.相反,在学习率较小时,相邻迭代的损失值下降幅度变化较小,从较大下降幅度下降到调用改进型 PSO 算法的阈值需要的迭代次数较多,这一定程度上降低了融合算法的效率.

通过加大学习率来提高融合算法的优化效率并非无限制的,更高的学习率会导致传统算法震荡现象的大量出现,因此,过大的学习率本身没有任何实用和对比意义.另外,学习率的提高会使得改进型 PSO 算法调用频率的增加,而改进型 PSO 算法的迭代效率低于 SGD 算法,这会导致网络整体迭代效率的下降.

5 总结与展望

本文引入 DBN 进行 CTR 预估,给出了其结构及训练方法,通过实验探讨了不同的隐藏层层数,隐含节点数目以及迭代周期对预测结果的影响,并与其他模型的预估结果进行对比分析,实验证明了使用 DBN 作为构造模型的融合模型相比现有的 CTR 预估常见算法具有更好的 CTR 预估效果,预估精度在 MSE、AUC 和 LogLoss

指标上优于 GBDT+LR 模型的融合模型 2.39%,9.70%和 2.46%,优于 FDNN 模型 1.24%,7.61%和 1.30%.

优化策略方面,通过实验证明了在 CTR 预估问题的 DBN 模型中,驻点对网络训练效率和结果有很大的影响.接着,本文从发掘 DBN 损失函数特性入手,针对驻点特征,提出了一种结合了 SGD 和 PSO 的融合算法.该融合算法在迭代步长小于阈值时可以跳出驻点平面,继续正常迭代.最终实验结果表明,融合算法能够很好地结合 SGD 的高效与 PSO 的梯度无关性,在不影响网络训练结果的前提下,提高了网络训练的效率 30%~70%.

本文提出的融合算法仍有一些后续工作值得扩展:(1) 在 DNN 的训练中,如何系统科学地设置学习率已是研究人员的研究重点[26,27],也是本文提出的融合算法的关键参数之一;(2) 本文使用阈值方法判断驻点和驻点平面,下一步本文考虑引入自适应方法,进行该阈值的动态调节;(3) 引入更多的应用场景,考察本文提出的融合算法对于其他应用场景中的 DBN 乃至 DNN 是否存在普适性,以及本文研究结论的一般性.

References:

[1] Zhou AY, Zhou MQ, Gong XQ. Computational advertising: A data-centric comprehensive Web application. Chinese Journal of Computers, 2011,34(10):1805−1819 (in Chinese with English abstract).

[2] Atkinson G. Search engine advertisement design effects on click-through rates. Journal of Interactive Advertising, 2014,14(1): 24−30. [doi: 10.1080/15252019.2014.890394]

[3] He XR, Pan JF, Jin Q, Xu TB, Liu B, Xu T, Shi YX, Atallah A, Herbrich R, Bowers A, Candela JQ. Practical lessons from predicting clicks on ads at facebook. In: Proc. of the 8th Int'l Workshop on Data Mining for Online Advertising (ADKDD 2014). 2014. [doi: 10.1145/2648584.2648589]

[4] Kennedy J, Eberhart R. Particle swarm optimization. In: Proc. of the IEEE Int'l Conf. on Neural Networks, Vol.4. 1995. 1942−1948.

[5] Piao HG, Wang ZX, Zhang HQ. Nonlinear control system of PID neural network based on cooperated particle swarm optimization (PSO). Control Theory & Applications, 2009,26(12):1317−1324 (in Chinese with English abstract).

[6] Chapelle O. Modeling delayed feedback in display advertising. In: Proc. of the ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2014. 1097−1105. [doi: 10.1145/2623330.2623634]

[7] Rumelhart DE, Hinton GE, Williams RJ. Learning Internal Representations by Error Propagation. MIT Press, 1988.

[8] Shan LL. Predicting ad click-through rates via feature-based fully coupled interaction tensor factorization. Electronic Commerce Research and Applications, 2016,16(C):30−42. [doi: 10.1016/j.elerap.2016.01.004]

[9] Zhang Y, Dai H, Xu C, Feng J, Wang T, Bian J, Wang B, Liu T. Sequential click prediction for sponsored search with recurrent neural networks. In: Proc. of the 28th AAAI Conf. on Artificial Intelligence (AAAI 2014). 2014. 1369−1375.

[10] Chen QH, Yu SM, Guo ZX, Jia YB. Estimating ads' click through rate with recurrent neural network. ITM Web of Conferences, 2016,7:Article No.04001. [doi: 10.1051/ itmconf/20160704001]

[11] Jiang Z, Gao S, Li M. An improved advertising CTR prediction approach based on the fuzzy deep neural network. Plos One, 2018, 13(5):Article No.e0190831. [doi: 10.1371/journal.pone.0190831]

[12] Mao Y, Shen J, Gui X. A study on deep belief net for branch prediction. IEEE Access, 2017. [doi: 10.1109/ACCESS.2017.2772334]

[13] Rumelhart DE, Hinton GE, Williams RJ. Learning Internal Representations by Error Propagation. MIT Press, 1988.

[14] Sutskever I, Martens J, Dahl G, *et al*. On the importance of initialization and momentum in deep learning. In: Proc. of the Int'l Conf. on Machine Learning. 2013.

[15] Krummenacher G, Mcwilliams B, Kilcher Y, Buhmann JM, Meinshausen N. Scalable adaptive stochastic optimization using random projections. In: Proc. of the 30th Conf. on Neural Information Processing Systems (NIPS 2016). 2016.

[16] Wang L. Damped Newton Method—An Ann Learning Algorithm. 1995.

[17] Lo TH, Gui Y, Peng Y. Overcoming the local-minimum problem in training multilayer perceptrons with the NRAE training method. In: Proc. of the Int'l Conf. on Advances in Neural Networks. Springer-Verlag, 2012. 440−447. [doi: 10.1007/978-3-642-31346-2_50]

[18] Chen YJ, Huang TC, Hwang RC. An effective learning of neural network by using RFBP learning algorithm. Information Sciences, 2004,167(1-4):77−86.

[19] Hamey LG. XOR has no local minima: A case study in neural network error surface analysis. Neural Networks the Official Journal of the Int'l Neural Network Society, 1998,11(11):669−681. [doi: 10.1016/S0893-6080(97)00134-2]

[20] Im DJ, Tao M, Branson K. An Empirical Analysis of Deep Network Loss Surfaces. 2016. https://arxiv.org/pdf/1612.04010v1.pdf

[21] Bishop CM. Neural Networks for Pattern Recognition. Oxford University Press, 1995.

[22] Dan C, Meier U, Masci J, Schmidhuber J. Multi-column deep neural network for traffic sign classification. Neural Networks: The Official Journal of the Int'l Neural Network Society, 2012,32(1):333−338. [doi: 10.1016/j.neunet.2012.02.023]

[23] Dauphin YN, Pascanu R, Gulcehre C, *et al*. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In: Advances in Neural Information Processing Systems. 2014. 2933−2941.

[24] Schenk O, Wächter A, Hagemann M. Matching-Based preprocessing algorithms to the solution of saddle-point problems in large-scale nonconvex interior-point optimization. Computational Optimization and Applications, 2007,36(2):321−341. [doi: 10.1007/s10589-006-9003-y]

[25] Jin HH, Chen J, Tang Z, Zheng GQ. Learning algorithm for solving local minimum problems based on Hopfield network. Journal of Tsinghua University (Science and Technology), 2002,42(6):731−734 (in Chinese with English abstract).

[26] Zhang S, Fu Q, Xiao W. Advertisement click-through rate prediction based on the weighted-ELM and adaboost algorithm. Scientific Programming, 2017,2017:Article ID 2938369. [doi: 10.1155/2017/2938369]

[27] Song G, Zhang J, Sun Z. The research of dynamic change learning rate strategy in BP neural network and application in network intrusion detection. In: Proc. of the Int'l Conf. on Innovative Computing Information and Control. IEEE Computer Society, 2008. [doi: 10.1109/ICICIC.2008.668]

附中文参考文献:

[1] 周傲英,周敏奇,宫学庆.计算广告:以数据为核心的 Web 综合应用.计算机学报,2011,34(10):1805−1819.

[5] 朴海国,王志新,张华强.基于合作粒子群算法的 PID 神经网络非线性控制系统.控制理论与应用,2009,26(12):1317−1324.

[25] 金海和,陈剑,唐政,郑国旗.基于 Hopfield 网络的极小值问题学习算法.清华大学学报(自然科学版),2002,42(6):731−734.

陈杰浩(1984－),男,广东潮州市人,博士,高级实验师,主要研究领域为复杂信息系统,大数据应用.

史继筠(1991－),女,硕士,主要研究领域为计算广告.

张钦(1994－),男,硕士,主要研究领域为计算广告.

赵子芊(1993－),男,硕士,主要研究领域为计算广告.

王树良(1974－),男,博士后,教授,博士生导师,主要研究领域为空间数据挖掘.