基于差分进化的因子分解机算法

喻 飞[1] 赵志勇[2] 魏 波[3]

(闽南师范大学物理与信息工程学院 漳州 363000)[1] (武汉大学计算机学院 武汉 430072)[2]

(华东交通大学软件学院 南昌 330013)[3]

摘 要 因子分解机(Factorization Machine,FM)算法是一种基于矩阵分解的机器学习算法,可用于求解回归、分类和排序等问题。FM模型中的参数求解使用的是基于梯度的优化方法,然而在样本较少的情况下,该优化方法收敛速度慢,且易陷入局部最优。差分进化算法(Differential Evolution,DE)是一种启发式的全局优化算法,具有收敛速度快等特性。为提高FM模型的训练速度,利用DE计算FM模型参数,提出了DE-FM算法。在数据集Diabetes、Horse-Colic以及音乐分类数据集Music上的实验结果表明,改进后的基于差分进化的因子分解机算法DE-FM在训练速度和准确性上均有所提高。

关键词 因子分解机,差分进化算法,机器学习

中图法分类号 TP18 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.9.054

Factorization Machine Based on Differential Evolution

YU Fei[1] ZHAO Zhi-yong[2] WEI Bo[3]

(School of Physics and Information Engineering, Minnan Normal University, Zhangzhou 363000, China)[1]

(College of Computer Science, Wuhan University, Wuhan 430072, China)[2]

(School of Software, East China Jiaotong University, Nanchang 330013, China)[3]

Abstract Factorization machine(FM) is a new machine learning algorithm based on the matrix factorization. It can be used to deal with the regression problems, classification problems and ranking problems. The solution of parameters in this model is based on the optimization method of gradient. However, under the condition of small amount of samples, the optimization method based on gradient has a slow convergence rate and may stick into local optimum. Differential evolution(DE) is a heuristic global optimization algorithm. It has a fast convergence rate. In order to improve the accuracy of FM, we proposed the DE-FM algorithm, which searches the best parameters of FM model with DE algorithm. We compared DE-FM with FM on the Diabetes dataset, the Horse-Colic dataset and the Music dataset, and the result shows that DE-FM can improve the accuracy.

Keywords Factorization machine, Differential evolution, Machine learning

1 引言

机器学习的发展过程中涌现了大量的优秀模型和算法,如基于前馈神经网络模型的BP算法[1-3]、Logistic回归算法[4-6]及支持向量机(SVM)[7]等,这些算法对机器学习的发展起着重要的作用。新的应用领域的不断扩大,特别是大数据应用问题的出现,对于通用性机器学习算法的需求不断扩大。因子分解机(Factorization Machine,FM)[8,9]算法是由Steffen Rendle等提出的一种新的通用的基于矩阵分解模型的机器学习算法,它将支持向量机的优势融入到了因子分解模型中。在对特征进行建模的过程中,因子分解机模型不仅考虑了单个特征对模型的影响,而且将特征与特征之间的相互关系考虑到模型中,这种特征与特征之间的关系可以是两个特征之间的关系或者是更多特征之间的关系。因子分解机算法的出现,为特征稀疏的问题诸如推荐问题提供了一种有效的解决途径[10,11]。

然而,在因子分解机算法模型的参数求解过程中,主要采用基于梯度的求解方法。使用梯度的求解方法求解模型需要大量的训练时间,而且在参数的求解过程中已陷入局部最优。

差分进化算法(Differential Evolution,DE)是由Storn R和Price K提出的一种新的启发式全局优化算法[12]。DE在求解复杂问题的过程中具有高效性、收敛速度快、鲁棒性高等优点。差分进化算法由于简单的数学结构和较高的效率,被广泛地应用在复杂的函数优化、神经网络的训练以及数据挖掘等问题中[13-15]。随着差分进化算法的发展,出现了一些对基本差分进化算法的改进算法,例如在其中引入了学习机制[16]、动态自适应策略[17]等,它们进一步提高了差分进化算法的性能。

为了更加准确地求出因子分解机算法中的参数,以提高算法的性能,提出了一种基于差分进化的因子分解机算法(Differential Evolution Factorization Machine,DE-FM),其充分利用了差分进化算法的全局搜索能力来求解基本的因子分

到稿日期:2015-06-28 返修日期:2015-09-24 本文受福建省自然科学基金(2015J01270),江西省青年科学基金(GJJ14396)资助。

喻 飞(1981-),男,博士生,讲师,主要研究方向为演化计算、机器学习等,E-mail:phonex@mnnu.edu.cn;赵志勇(1989-),男,硕士生,主要研究方向为机器学习、数据挖掘;魏 波(1983-),男,博士,主要研究方向为智能计算。

解机算法中的模型参数。在医疗诊断数据集 Diabetes、从疝气病预测病马的死亡率数据集 Horse-Colic 以及音乐分类数据集 Music 上的仿真实验表明，DE-FM 比基本的因子分解机算法 FM 在二分类问题上准确性更高且训练时间更短。

2 因子分解机

2.1 模型描述

文献[8]中提出了因子分解机模型，其模型基于矩阵分解，又同时吸纳了支持向量机与因子分解的优点。为便于描述，下面给出度为 2 的因子分解机模型，度为 2 即仅考虑两个特征之间的相互关系，其模型可以用式(1)描述为：

$$\hat{y}:=\omega_0+\sum_{i=1}^{n}\omega_i x_i+\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\langle v_i,v_j\rangle x_i y_j \tag{1}$$

其中，$\omega_0\in R$，$w\in R^n$，$V\in R^n$。$\langle v_i,v_j\rangle$表示两个大小为 k 的向量 v_i 和向量 v_j 的点积：

$$\langle v_i,v_j\rangle:=\sum_{f=1}^{k}v_{i,f}\cdot v_{j,f} \tag{2}$$

其中，v_i 表示系数矩阵 V 的第 i 维向量，且该向量可表示为 $v_i=(v_{i,1},v_{i,2},\cdots,v_{i,k})$，$k(k\in N^+)$称为超参数。系数矩阵 V 表示如下：

$$V=\begin{pmatrix}v_1\\v_2\\\vdots\\v_n\end{pmatrix}=\begin{pmatrix}v_{1,1}&v_{1,2}&\cdots&v_{1,k}\\v_{2,1}&v_{2,2}&\cdots&v_{2,k}\\\vdots&\vdots&&\vdots\\v_{n,1}&v_{n,2}&\cdots&v_{n,k}\end{pmatrix} \tag{3}$$

同样，因子分解机模型可以推广到多阶的形式，即在模型中考虑多个特征之间的相互关系，本文使用的是度为 2 的因子分解机模型。相对于基于矩阵分解的其他机器学习算法(如 SVD++)，因子分解机算法是一种更加通用的机器学习算法，可以处理回归、二分类等多种问题。

在回归问题中，可以直接使用$\hat{y}$作为回归问题的预测结果，那么其损失函数定义为：

$$loss^R(\hat{y},y)=\sum_{i=1}^{m}(\hat{y}^{(i)}-y^{(i)}) \tag{4}$$

其中，m 表示样本个数。

对于二分类问题，可以使用 Sigmoid 函数将$\hat{y}$映射到两个不同的类别。那么，其损失函数则可以定义为：

$$loss^C(\hat{y},y)=\sum_{i=1}^{m}-\ln\sigma(\hat{y}^{(i)}y^{(i)}) \tag{5}$$

其中，σ 为 Sigmoid 函数，具体形式为：

$$\sigma(x)=\frac{1}{1+e^{-x}} \tag{6}$$

2.2 模型求解过程

因子分解机算法中的参数主要包括 $\omega_0\in R$，$w\in R^n$ 和 $v\in R^{n\times k}$，可以使用基于梯度的方法对这些参数进行求解。为了降低算法的复杂度，$\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\langle v_i,v_j\rangle x_i x_j$ 可以按如下过程进行化简：

$$\begin{aligned}&\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\langle v_i,v_j\rangle x_i x_j\\&=\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\langle v_i,v_j\rangle x_i x_j-\frac{1}{2}\sum_{i=1}^{n}\langle v_i,v_i\rangle x_i x_i\\&=\frac{1}{2}\left(\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{f=1}^{k}v_{i,f}v_{j,f}x_i x_j-\sum_{i=1}^{n}\sum_{f=1}^{k}v_{i,f}v_{i,f}x_i x_i\right)\\&=\frac{1}{2}\sum_{f=1}^{n}\left(\left(\sum_{i=1}^{n}v_{i,f}x_i\right)\left(\sum_{j=1}^{n}v_{j,f}x_j\right)-\sum_{i=1}^{n}v_{i,f}^2x_i^2\right)\\&=\frac{1}{2}\sum_{f=1}^{k}\left(\left(\sum_{i=1}^{n}v_{i,f}x_i\right)^2-\sum_{i=1}^{n}v_{i,f}^2x_i^2\right)\end{aligned} \tag{7}$$

那么，对于回归问题和二分类问题来说，基于梯度的优化方法的更新公式为：

$$\begin{cases}\omega_0=\omega_0-\eta\left(\dfrac{\partial loss(\hat{y},y)}{\partial\omega_0}\right)\\\omega_i=\omega_i-\eta\left(\dfrac{\partial loss(\hat{y},y)}{\partial\omega_i}\right)\\v_{i,j}=v_{i,j}-\eta\left(\dfrac{\partial loss(\hat{y},y)}{\partial v_{i,j}}\right)\end{cases} \tag{8}$$

其中，η 表示学习率，$\frac{\partial loss(\hat{y},y)}{\partial\theta}$则表示损失函数对参数的偏导数。

因此，对于回归问题，其偏导数可以表示为：

$$\frac{\partial loss^R(\hat{y},y)}{\partial\theta}=2(\hat{y}-y)\cdot\frac{\partial\hat{y}}{\partial\theta} \tag{9}$$

而对于二分类问题，其偏导数为：

$$\frac{\partial loss^C(\hat{y},y)}{\partial\theta}=[\sigma(\hat{y}y)-1]\cdot\frac{\partial\hat{y}}{\partial\theta} \tag{10}$$

在利用梯度的方法求解模型中的参数时，需要对$\hat{y}$求偏导数，则因子分解机中模型的梯度表示为：

$$\frac{\partial}{\partial\theta}\hat{y}(X)=\begin{cases}1, & \text{if } \theta=\omega_0\\x_i, & \text{if } \theta=\omega_i\\x_i\sum_{j=1}^{n}v_{j,f}x_j-v_{i,f}x_i^2, & \text{if } \theta=v_{i,f}\end{cases} \tag{11}$$

3 差分进化算法

差分进化算法在求解问题的整个过程中主要包括 3 个步骤：变异、杂交和选择。根据不同的杂交策略，DE 有不同的表现形式，文中主要采用的是 *DE/rand/1* 策略。对于一个 D 维空间中的搜索问题，种群中包含 NP 个参数向量 $X_{i,G}$，$i=1,2,\cdots,NP$，其中，G 表示算法运行的代数，NP 表明了种群中个体的数量。一般将目标函数作为适应值函数，表示为 $F(X_{i,G})$。新的个体 $X'_{i,G}$是通过变异和杂交的过程产生的，最后通过选择算子在原始的个体 $X_{i,G}$和新产生的个体 $X'_{i,G}$之间选择较好的个体。

3.1 变异

基于不同的变异策略就会有差分进化算法的不同形式，使用 *DE/rand/1* 策略时，变异操作产生新的个体 $X'_{i,G+1}$的形式如下：

$$X'_{i,G+1}=X_{r1,G}+F*(X_{r2,G}-X_{r3,G}) \tag{12}$$

其中，r_1,r_2,r_3 是从区间上[1,NP]上随机选取的 3 个不相等的数，即 $r_1\neq r_2\neq r_3$，F 为缩放因子，是一个常数。

3.2 杂交

为了防止算法陷入局部最优，需要在计算的过程中增加种群的多样性。杂交操作的目的是增加种群的多样性，防止 DE 算法陷入局部最优。杂交操作根据杂交概率 CR 产生新的个体，新产生的个体如式(13)所示：

$$U_{i,G+1}=(u^1_{i,G+1},u^2_{i,G+1},\cdots,u^D_{i,G+1}) \tag{13}$$

其中，$u^j_{i,G+1}$通过下式产生：

$$u_{i,G+1}^{j}=\begin{cases}x_{i,G+1}^{'j}, & \text{if}(rand_j[0,1]\leqslant CR)\\ x_{i,G}^{j}, & \text{otherwise}\end{cases} \tag{14}$$

其中，$j=1,2,\cdots,D$；CR 是区间[0,1]上的一个随机数。

3.3 选择

在完成了变异操作和杂交操作之后，DE 算法要更新种群的结构，选择出较优的个体进行下一代的操作，选择操作主要是根据适应值函数进行贪婪选择，在原始的个体 $X_{i,G}$ 和交叉操作后形成的个体 $U_{i,G+1}$ 之间选择较好的个体，并将这个较好的个体放到下一代的个体中。选择操作的过程用式(5)表示：

$$X_{i,G+1}=\begin{cases}U_{i,G+1}, & \text{if } (f(U_{i,G+1})\leqslant f(X_{i,G}))\\ X_{i,G}, & \text{otherwise}\end{cases} \tag{15}$$

其中，$f(U_{i,G+1})$ 表示的是个体 $U_{i,G+1}$ 的适应值，$f(X_{i,G})$ 表示的是个体 $X_{i,G}$ 的适应值。

4 基于差分进化的因子分解机算法

在利用梯度下降的方法求解因子分解机 FM 算法模型参数的过程中，迭代求解的过程需要大量的训练时间，而且基于梯度的算法不易求出模型中参数的全局最优解。为解决此问题，本文提出基于差分进化算法的因子分解机算法 DE-FM，其利用差分进化算法的全局搜索能力求解因子分解机模型参数。

4.1 DE-FM 模型

假设有 N 个任意的样本 $(X^{(i)},y^{(i)})$，其中 $X^{(i)}=[x_1^{(i)},x_2^{(i)},\cdots,x_n^{(i)}]\in R^n$，$y^{(i)}$ 表示的是期望输出。那么，对于回归问题，有 $y^{(i)}\in R$，而对于二分类问题，则有 $y^{(i)}\in\{0,1\}$。

对于样本中的 n 个特征，基于差分进化的因子分解机 DE-FM 模型可以表示为：

$$\hat{y}:=\omega_0+\omega_1x_1+\omega_2x_2+\cdots+\omega_nx_n+\langle v_1,v_2\rangle x_1x_2+\cdots+\langle v_{n-1},v_n\rangle x_{n-1}x_n \tag{16}$$

其中，$\langle v_i,v_j\rangle$ 表示的是两个大小为 k 的向量 v_i 和 v_j 的点积：

$$\langle v_i,v_j\rangle:=\sum_{f=1}^{k}v_{i,f}\cdot v_{j,f} \tag{17}$$

对于回归问题，则直接使用 $\hat{y}$ 作为最终的回归结果。在二分类问题中，需要添加 Sigmoid 函数进行映射。

4.2 DE-FM 参数求解

在本文中，主要利用基于差分进化的因子分解机 DE-FM 处理二分类问题。对于 N 个任意的样本 $(X^{(i)},y^{(i)})$，其中 $X^{(i)}=[x_1^{(i)},x_2^{(i)},\cdots,x_n^{(i)}]\in R^n$，$y^{(i)}\in\{0,1\}$。在 DE-FM 中，参数求解过程通过差分进化算法完成。利用 DE 求解时，首先构造种群 $P=\{p_1,p_2,\cdots,p_{NP}\}$，种群大小为 NP，每个个体可以表示为：

$$p_i=\{\omega_0,\cdots,\omega_n,v_{1,1},\cdots,v_{1,k},\cdots,v_{n,1},\cdots,v_{n,k}\} \tag{18}$$

个体 p_i 即为 FM 模型中的参数向量，代表一组待求解的参数。显然，该向量的维度为 $D=n\times k+n+1$。

在利用差分进化算法求解参数的过程中，除了个体的编码问题，还需要考虑适应值函数的设计。基本的因子分解模型 FM 中使用的是损失函数 $loss^C(\hat{y},y)$。在基于差分进化的因子分解机 DE-FM 中，使用错误率作为适应值函数，用以下形式表示：

$$errRate=\frac{\#error}{\#all} \tag{19}$$

其中，$\#error$ 表示预测错误的样本的个数，$\#all$ 表示所有样本的个数。在对样本进行二分类的过程中，利用 Sigmoid 函数将 $\hat{y}$ 映射到两个不同的类别，若 $\sigma(\hat{y})>0.5$，则最终的类别为 1，否则类别为 0。

4.3 算法步骤

由前面的介绍可得知，使用差分进化算法的目的就是获得式(16)中的众多参数的最优值。因此，可以将基于差分进化的因子分解机 DE-FM 的算法步骤描述如下。

步骤 1　初始化种群。根据参数设置(种群规模 NP、向量维数 D)随机生成种群 P_G，此时 $G=0$。

步骤 2　重复执行以下过程，直到满足终止条件(迭代次数达到指定值)：

1)计算种群中个体的适应值。将每个个体 $X_{i,G}$ 代表的参数向量转换为 FM 模型对应的参数，使用 FM 算法对训练样本进行训练，将式(19)计算的错误率作为个体的适应值。保存当前全局最优解 X_{best} 和 f_{best}。

2)变异。根据式(12)对种群中的个体 $X_{i,G}$ 进行变异操作，产生新的个体 $X'_{i,G+1}$。

3)杂交。根据式(13)和式(14)对种群中的个体进行交叉操作，得到实验个体 $U_{i,G+1}$。

4)选择。根据式(15)获得新的个体 $X_{i,G+1}$，从而构成新的种群 P_{G+1}。

步骤 3　根据最优解 X_{best} 转换所获得的参数向量构建 FM 模型，用于对待测试数据的预测。

5 实验与结果分析

5.1 实验数据

为了测试本文提出的基于差分进化的因子分解机算法 DE-FM，选取了医疗诊断数据集 Diabetes[1)]、从疝气病预测病马的死亡率数据集 Horse-Colic[2)] 以及文献[18]中使用的音乐分类数据集 Music，将所提方法与传统的因子分解机学习算法 FM 进行对比。其中，Diabetes 数据集中包含 576 个训练数据和 192 个测试数据，是皮马印第安人糖尿病数据集，特征个数为 8，分类的结果为是否有糖尿病；Horse-Colic 数据集中包含了 299 个训练样本和 140 个测试样本，样本中的特征个数为 21，主要用来预测患有疝气病的马的存活问题；Music 数据集中包含 740 个训练样本和 300 个测试样本，样本中的特征个数为 24，主要根据音乐的不同属性判断其为民歌、古筝、摇滚和流行 4 种类型中的哪一类，本文从其中任选了两类作为二分类问题来测试提出的算法。

5.2 参数设置

实验中采用 Python 2.7 编写算法代码，在 i7-3.4G 处理器的运行环境下运行。对于上文所述的 3 个测试数据集，本文设置了不同的运行参数，表 1 列出了具体的参数设置情况。

1) http://archive.ics.uci.edu/ml/datasets/Diabetes

2) http://archive.ics.uci.edu/ml/datasets/Horse+Colic

表 1　FM 和 DE-FM 的运行参数

数据集	η	G_{FM}	$G_{DE\text{-}FM}$	k
Diabetes	0.01	200	30	20
HorseColic	0.01	200	30	30
Music	0.01	200	30	24

表 1 中，η 表示基于梯度的方法的学习率，G 表示学习的代数，k 表示矩阵分解中的维度。在 3 个数据集上，DE-FM 算法均使用相同的参数设置：种群大小 $NP=5$，变异概率 $F=0.4$，杂交概率 $CR=0.8$。

5.3　结果及分析

为了验证 DE-FM 算法是否能在较短的演化代数内获得满意的精度。图 1—图 3 分别示出了 DE-FM 算法在 3 个数据集的训练集上的训练错误率的变化情况。

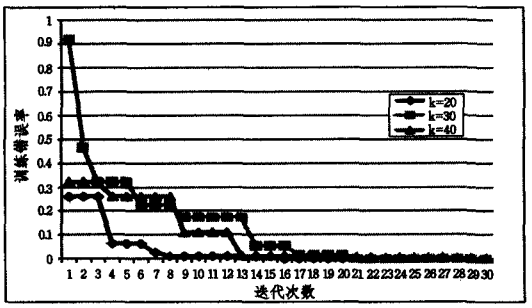

图 1　DE-FM 算法在数据集 Diabetes 上的表现

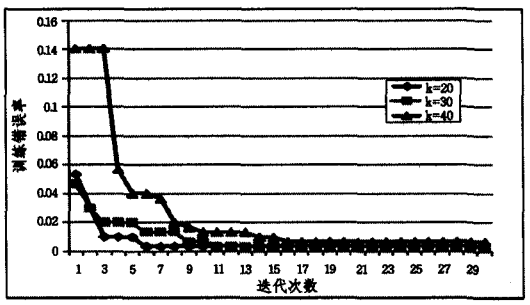

图 2　DE-FM 算法在数据集 HorseColic 上的表现

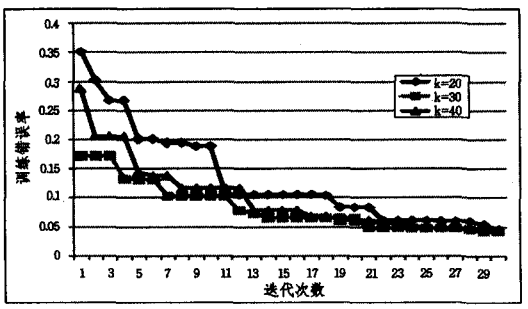

图 3　DE-FM 算法在数据集 Music 上的表现

从图 1—图 3 中不难看出，除了数据集 Music 外，算法均能在有限的迭代次数(30 代以内)内达到较高的训练精度(数据集 Music 的误差率也达到了可以接受的范围)；并且，DE-FM 算法的测试精度不受矩阵分解系数 k 值大小的影响，通用性强。

算法的性能测试分别在训练数据集和测试数据集上进行，本文采用训练时间(TrainingTime，TT)、训练的准确性(TrainingAccuracy，TRA)和测试的准确性(TestingAccuracy，TEA)作为算法性能测试和比较的指标。表 2 列出了算法 FM 和 DE-FM 在数据集 Diabetes、HorseColic 以及 Music 上的运行结果。

表 2　算法在各数据集上的运行结果

对比指标	Diatetes		HorseColic		Music	
	FM	DE-FM	FM	DE-FM	FM	DE-FM
TT(mm:ss)	01:57	00:28	02:54	02:18	08:50	06:24
TRA	0.769	0.997	0.712	0.989	0.595	0.968
TEA	0.776	0.995	0.679	0.979	0.577	0.970

DE 算法属于全局性优化算法，实验中对 DE 中的参数(如种群大小 N、优化的代数 G)都设置得很小，以求在速度上能够与 FM 在一个量级上。从表 2 中的对比结果可以看出，DE-FM 算法在训练时间、训练集和测试集上的准确性都要优于基本的 FM 算法，也就是说改进后的基于差分进化的因子分解机算法 DE-FM 能够在保证训练速度的前提下获得比传统的因子分解机 FM 更高的训练准确性和测试准确性。

因子分解机算法主要是通过基于梯度的优化算法调整算法模型中的参数，然而在参数学习的过程中，传统的基于梯度的学习算法需要通过迭代的方式调整参数，计算时间成本高，而且容易陷入局部最优。对于一些数据集，如本文中所提到的数据集，训练样本量少，而所需要训练的模型参数很多，基于梯度的学习算法不容易学习出所有的最优参数。改进后的基于差分进化的因子分解机算法 DE-FM 则具有良好的性能，这要归功于 DE 的全局搜索能力。相较于基于梯度的方法，DE 算法的鲁棒性更好，对于复杂的参数寻优问题表现出更强的搜索能力。

结束语　本文将 DE 算法引入到因子分解机 FM 的参数求解过程中，摒弃了传统基于梯度的参数求解方法。通过在不同数据集上的测试结果表明，改进后的基于差分进化的因子分解机算法 DE-FM 的性能明显优于传统的因子分解机算法 FM；而且由于差分进化算法 DE 结构简单，可以通过引入新的策略提高 DE 本身的全局搜索能力，从而进一步提升 FM 算法以适应不同规模的数据集，扩展 FM 算法的应用。

FM 算法是应大规模数据的应用而产生的，但是当数据集规模以及训练样本数量受到限制时其性能也会受到限制，DE-FM 算法则在训练样本少的情况下仍能取得较好的性能。本文中 DE 算法使用了统一的参数设置，种群规模和演化代数都设置得比较小，这在某种程度上限制了 DE 的性能。对于不同规模数据集的应用，DE-FM 则需要使用不同的参数设置以及新的策略，例如大规模数据集则需要扩大 DE 算法中的种群规模和增加演化代数。然而，伴随着 DE 算法种群的扩大，演化代数的增加等参数的变化必然会造成 FM 模型的参数寻优过程的时间开销增大，从而延长训练时间。为了突破该限制，后续的研究工作中需要提高 DE 算法在 FM 模型参数训练中的使用效率，以适应不同规模数据集下的应用需求；此外，本文只讨论了二分类问题，DE-FM 算法在多分类问题的应用还需要作进一步探讨。

参 考 文 献

[1] Chauvin Y, Rumelhart D E. Backpropagation: theory, architectures, and applications[M]. Psychology Press, 1995

[2] Deng W J. Chen W C, Pei W. Back-propagation neural network based importance-performance analysis for determing critical service attributes[J]. Expert Systems with Applications, 2008, 34(2): 1115-1125

[3] Sridevi K, Sivaraman E, Mullai P. Back propagation neural network modelling of biodegradation and fermentative biohydrogen production using distillery wastewater in a hybrid upflow anae-

robic sludge blanket reactor[J]. Bioresource Technology, 2014, 165(8): 233-240

[4] Hosmer D W, Lemeshow S. Introduction to the logistic regression model[M]// Applied Logistic Regression(Second Edition). John Wiely & Sons, Inc. 2005: 1-30

[5] Ng A Y, Jordan M I. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes [J]. Advances in Neural Information Processing Systems, 2002, 2 (3): 169-187

[6] Lemeshow S, Hosmer D W. A review of goodness of fit statistics for use in the development of logistic regression models[J]. American Journal of Epidemiology, 1982, 115(1): 92-106

[7] Cortes C, Vapnik V. Support-vector networks [J]. Machine Learning, 1995, 20(3): 273-297

[8] Rendle S. Factorization machines[C]// 2010 IEEE 10th International Conference on Data Mining (ICDM). 2010: 995-1000

[9] Rendle S. Factorization machines with libfm[J]. ACM Transactions on Intelligent Systems and Technology (TIST), 2012, 3 (3): 57

[10] Rendle S. Scaling factorization machines to relational data[C]// Proceedings of the VLDB Endowment. 2013: 337-348

[11] Rendle S, Gantner Z, Freudenthaler C, et al. Fast context-aware recommendations with factorization machines[C]// Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2011: 635-644

[12] Storn R, Price K. Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces [J]. Journal of Global Optimization, 1997, 11(4): 341-359

[13] Ilonen J, Kamarainen J K, Lampinen J. Differential evolution training algorithm for feed-forward neural networks[J]. Neural Processing Letters, 2003, 17(1): 93-105

[14] Kotha S R, Vij S, Sahoo S K. A study on strategies and Mutant factor in differential evolution algorithm for FIR filter design[C]// 2014 International Conference on Signal Processing and Integrated Networks (SPIN). 2014: 50-55

[15] Wang G-G, Gandomi A H, Alavi A H, et al. Hybrid krill herd algorithm with differential evolution for global numerical optimization[J]. Neural Computing and Applications, 2014, 25(2): 297-308

[16] Zhou Y L, Zhu Y H, Zhang J. Discrete Differential Evolution with Learning Mechanism[J]. Computer Science, 2011, 38(7): 225-227(in Chinese)
周雅兰，朱耀辉，张军. 具有学习机制的离散差分演化算法[J]. 计算机科学，2011，38(7)：225-227

[17] Wang C J, Wang X H, Xiao J M. Discrete Differential Evolution with Parameter Adaptive Mechanism [J]. Computer Science, 2014, 41(1): 279-282(in Chinese)
王丛佼，王锡淮，肖建梅. 具有参数自适应机制的改进离散差分进化算法[J]. 计算机科学，2014，41(1)：279-282

[18] Chun L Y, Song H, Yang J. Research on Music Classification Based on MFCC and BP Neural Network[C]// 2nd International Conference on Information, Electronics and Computer. 2014: 57-68

(上接第 249 页)

根据新的评分公式对文档集合重新排序。实验结果表明，与 Lucene 基础排序算法相比，使用改进后的排序算法时，检索系统的准确率、召回率和 F 值均有不同程度的提高，用户有更加良好的垂直搜索体验。但其还存在一些缺点，例如，在朴素贝叶斯分类模型中，分类的效率、准确率和召回率可能对检索系统的效率、准确率和召回率有所影响。下一步的研究工作是考虑用户兴趣和行为特征等因素，优化分类和检索过程，从而提高检索系统的性能。

参 考 文 献

[1] Liu J X, Sheng Y. The differences and case analysis of vertical and general search engines[J]. Modern Information, 2009, 29 (3): 143-149(in Chinese)
刘俊熙，盛宇. 垂直和通用搜索引擎的差异和案例分析[J]. 现代情报，2009，29(3)：143-149

[2] 牛长流，尚宇. Lucene 实战(第 2 版)[M]. 北京：人民邮电出版社，2011

[3] Bai K, Geng G H. Research and Application of vertical search engines based on Lucene/Heritrix[J]. Computer Applications and Software, 2009, 26(1): 212-215(in Chinese)
白坤，耿国华. 基于 Lucene/Heritrix 的垂直搜索引擎的研究与应用[J]. 计算机应用与软件，2009，26(1)：212-215

[4] Zhang X, Liu X F. Design and implementation of full-text search engine based on Lucene and Heritrix[J]. Modern Computer, 2013(22): 74-77(in Chinese)
张宣，刘晓飞. 基于 Lucene 和 Heritrix 的全文搜索引擎的设计与实现[J]. 现代计算机，2013(22)：74-77

[5] Cai F. Research and improvement of Lucene sorting algorithm [J]. New Technology and New Products of China, 2011(4): 15-16(in Chinese)
蔡峰. Lucene 排序算法的研究和改进[J]. 中国新技术新产品，2011(4)：15-16

[6] Chen J X, Huang R, Ma Z B. Optimization and implementation of Lucene sorting algorithm based on PageRank[J]. Computer Engineering and Science, 2012, 34(10): 123-127(in Chinese)
陈建峡，黄日，马忠宝. 基于 PageRank 的 Lucene 排序算法优化与实现[J]. 计算机工程与科学，2012，34(10)：123-127

[7] Mohd M. Development of Search Engines using Lucene: An Experience[J]. Procedia-Social and Behavioral Sciences, 2011, 18: 282-286

[8] Milosavljevic, Branko, Boberic, et al. Retrieval of bibliographic records using Apache Lucene[J]. The Electronic Library, 2010, 28(4): 525-539

[9] Rong G, Zhang H X. Application of text classification in the search engine[J]. Guide of Scitech Magazine, 2008, 12(2): 14-15 (in Chinese)
荣光，张化祥. 文本分类在搜索引擎性能中的应用[J]. 科技致富向导，2008，12(2)：14-15

[10] Lewis D D. Representation and learning in information retrieval [D]. Graduate School of the University of Maassachusetts, 1992

[11] Zhang X F. Analysis and evaluation of several common information retrieval model[J]. Journal of Intelligence, 2008, 27(3): 121-123(in Chinese)
张小芳. 几种常见信息检索模型的分析与评价[J]. 情报杂志，2008，27(3)：121-123

[12] Croft W B, Metzler D, Strohman T. Search Engine: Information Retrieval in Practice[M]. Pearson, 2010