

基于分解极限学习机的手写字符识别方法

何玉林, 李旭, 金一, 黄哲学

引用本文

何玉林, 李旭, 金一, 黄哲学. 基于分解极限学习机的手写字符识别方法[J]. 计算机科学, 2022, 49(11): 148-155.

HE Yu-lin, LI Xu, JIN Yi, HUANG Zhe-xue. [Handwritten Character Recognition Based on Decomposition Extreme Learning Machine](#)[J]. Computer Science, 2022, 49(11): 148-155.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[时序知识图谱表示学习](#)

Temporal Knowledge Graph Representation Learning

计算机科学, 2022, 49(9): 162-171. <https://doi.org/10.11896/jsjcx.220500204>

[基于 Key-Value 关联记忆网络的知识图谱问答方法](#)

Key-Value Relational Memory Networks for Question Answering over Knowledge Graph

计算机科学, 2022, 49(9): 202-207. <https://doi.org/10.11896/jsjcx.220300277>

[基于安全多方计算和差分隐私的联邦学习方案](#)

Federated Learning Scheme Based on Secure Multi-party Computation and Differential Privacy

计算机科学, 2022, 49(9): 297-305. <https://doi.org/10.11896/jsjcx.210800108>

[基于非局部注意力生成对抗网络的视频异常事件检测方法](#)

Non-local Attention Based Generative Adversarial Network for Video Abnormal Event Detection

计算机科学, 2022, 49(8): 172-177. <https://doi.org/10.11896/jsjcx.210600061>

[基于深度学习的社交网络舆情信息抽取方法综述](#)

Survey of Social Network Public Opinion Information Extraction Based on Deep Learning

计算机科学, 2022, 49(8): 279-293. <https://doi.org/10.11896/jsjcx.220300099>

基于分解极限学习机的手写字符识别方法

何玉林^{1,2} 李旭^{1,2} 金一³ 黄哲学^{1,2}

1 深圳大学计算机与软件学院 广东 深圳 518060

2 人工智能与数字经济广东省实验室(深圳) 广东 深圳 518107

3 中国刑事警察学院痕迹检验技术系 沈阳 110854

摘要 手写字符识别是图像识别的一个重要分支,是基于数据挖掘和机器学习技术对数字、字母和文字等的手写体进行识别。当前手写字符识别方法主要集中在对不同深度学习模型的完善和改进上,其中多层极限学习机由于其快于深度信念网络和深度玻尔兹曼机的训练速度以及更高的识别精度引起了学术界和工业界的广泛关注。但是,多层极限学习机的预测表现极易受随机权重的影响,层数越多影响就越明显。文中在深入分析浅层极限学习机训练模式的基础上,提出了一种基于隐含层输出矩阵分解的浅层极限学习机模型,并将其应用于对手写字符的识别。分解极限学习机不需要对手写字符图像进行特征提取,而是通过对大规模隐含层输出矩阵的分解来获得极限学习机的输出层权重。相比深层极限学习机,分解极限学习机降低了基于极限学习机的手写字符识别模型训练的随机性。同时,在 MNIST 类数据集(即 MNIST, EMNIST, KMNIST 和 K49-MNIST)上的比较结果表明,在相同的训练时间下,分解极限学习机能够获得优于多层极限学习机的识别精度;在相同的识别精度下,分解极限学习机的训练时间明显短于多层极限学习机。实验结果证实了解析极限学习的可行性以及在处理手写字符识别问题上的有效性。

关键词: 手写字符识别;极限学习机;多层极限学习机;深度学习;特征提取

中图法分类号 TP391

Handwritten Character Recognition Based on Decomposition Extreme Learning Machine

HE Yu-lin^{1,2}, LI Xu^{1,2}, JIN Yi³ and HUANG Zhe-xue^{1,2}

1 College of Computer Science & Software Engineering, Shenzhen University, Shenzhen, Guangdong 518060, China

2 Guangdong Laboratory of Artificial Intelligence and Digital Economy(SZ), Shenzhen, Guangdong 518107, China

3 Department of Criminal Science and Technology, Criminal Investigation Police University of China, Shenyang 110854, China

Abstract Handwritten character recognition(HCR) is an important branch of image recognition, which recognizes the handwritten characters with the data mining and machine learning technologies. Currently, the HCR methods mainly focus on the improvements of different deep learning models, where the multiple-layer extreme learning machine(ML-ELM) has attracted the wide attention from the academia and industry due to its faster training speed and better recognition performance than deep belief net(DBN) and deep Boltzmann machine(DBM). However, the recognition performance of ML-ELM is severely influenced by the random weights when determining the input weights for each hidden-layer. This paper first proposes a decomposition ELM(DE-ELM) which is a shallow ELM training scheme based on the hidden-layer output matrix decomposition and then applies DE-ELM to deal with HCR problems, i. e., handwritten digits in MNIST, handwritten digits and English letters in EMNIST, handwritten Japanese characters in KMNIST and K49-MNIST. In comparison with ML-ELM, DE-ELM reduces the randomness of ELM-based HCR model. Meanwhile, DE-ELM can obtain higher recognition accuracy than ML-ELM with the same training time and faster training speed than ML-ELM with the equal recognition accuracy. Experimental results demonstrate the feasibility and effectiveness of the proposed DE-ELM when dealing with HCR problems.

Keywords Handwritten character recognition, Extreme learning machine, Multiple layer extreme learning machine, Deep learning, Feature extraction

1 引言

手写字符识别(Handwritten Character Recognition,

HCR)是2018年全国科学技术名词审定委员会公布的计算机科学技术名词之一^[1],它的实质是利用数据挖掘和机器学习技术,如特征提取^[2]、 k -近邻^[3]、支持向量机^[4]、神经网络^[5]

到稿日期:2021-12-23 返修日期:2022-05-09

基金项目:国家自然科学基金(61972261);深圳市基础研究面上项目(JCYJ20210324093609026)

This work was supported by the National Natural Science Foundation of China(61972261) and Basic Research Foundation of Shenzhen(JCYJ20210324093609026).

通信作者:何玉林(yulinhe@gml.ac.cn)

等,对数字、字母、文字等的手写体图像进行识别。HCR 有着广泛的应用场景,例如实现对邮政编码、统计报表、财务报表、银行票据中手写字符的自动识别,这不仅能够提高相关部门的工作效率,同时能够降低人为误操作的风险。由于拍照扫描设备受到光照、分辨率、纸张材质等的影响,使得手写字符图像在采集过程中包含噪声数据^[6];另外,手写字符往往包含手写人较强的主观意志,例如不规范的书写习惯和不易辨别的连笔字体等,这些都会在很大程度上增加 HCR 的难度。

1985年,Shridhar等^[6]提出了基于拓扑特征与句法分类器相结合的手写数字识别方法,该方法通过树语法来最少化测试图案的预处理,以达到提高识别速率的目的。1989年,LeCun等^[7]提出了LeNet的原始模型,通过反向传播算法训练的卷积神经网络来实现对手写数字的识别。之后,LeCun等^[8]于1998年提出了LeNet5手写数字识别算法,该算法使用带有可学习参数的卷积神经网络来提取图像上的相似特征,然而其在处理手写数字识别任务时计算代价过大,网络训练速度慢,不能处理复杂手写字符数据。21世纪初,随着计算机硬件技术的飞速发展和图像采集技术的不断进步,使用深度学习对手写字符进行识别成为HCR研究的主流,这期间的代表性工作以Hinton等提出的深度置信网络(Deep Belief Net, DBN)^[9]和深度玻尔兹曼机(Deep Boltzmann Machine, DBM)^[10]最为著名。经典的DBN网络是由若干层受限玻尔兹曼机(Restricted Boltzmann Machine, RBM)和一层误差反传网络组成的一种深层神经网络,其中误差反传接收RBM的输出特征向量作为输入特征向量,进行有监督学习训练,构建针对HCR的深度学习模型。DBM与DBN十分类似,也是由若干RBM堆叠而成,但两者存在着本质的差别:整体来看,DBN是有向的(RBM与贝叶斯信念网的搭配),而DBM是完全无向的(全部是RBM)。从实际应用的效果来看,DBM比DBN具有更高的HCR稳定性,但是DBM的计算代价相比DBN要高很多。

2006年,Huang等^[11-12]提出了一种基于隐含层输出矩阵广义逆求解式的全链接前馈神经网络训练范式——极限学习机(Extreme Learning Machine, ELM)。不同于误差反传式的神经网络训练,ELM随机初始化输入层权重,解析求解输出层权重,从而获得了极快的网络训练速度。同时,通过理论证明^[13]了ELM的一致收敛性,这使得ELM备受学术界和产业界从业者的关注,其在有监督学习^[14]、无监督学习^[15]、半监督学习^[16]、增强学习^[17]、图像识别^[18]等领域获得了广泛的应用。在此,我们需要强调一点,ELM是一种特殊形式的随机权重网络。对随机权重网络的代表性研究有Schmidt等^[19]于1992年提出的具有随机权重的前馈神经网络(Feedforward Neural Network with Random Weights, FNNRW)和Igelnik等^[20]于1995年提出的随机向量功能链接网(Random Vector Functional Link Network, RVFLN)。关于ELM, FNNRW和RVFLN的相似性与差异性的争辩,超出了本文讨论的范围,感兴趣的读者可参见近期关于ELM研究的文献^[21-23],此处不再赘述。

2013年,Kasun等^[24]尝试基于ELM构建用于HCR的深度学习模型——多层极限学习机(Multi-Layer Extreme Learning Machine, ML-ELM),实验结果表明ML-ELM能够

获得优于DBN和DBM的手写字符识别表现。ML-ELM开创了快速深度神经网络训练的先河,给出了一种解析解式深度学习的训练模式。之后,Ding等^[25]和Tang等^[26]于2015年分别提出了将ML-ELM与核ELM相结合的深度ELM和基于 l_1 范数规则化项与迭代阈值收缩法权重求解的层叠式ELM,这两种不同类型的ELM深度学习算法在一定程度上都对ML-ELM的学习效率有所改进。

通过对ML-ELM进行深入分析发现,尽管它在HCR方面具有训练速度快、识别精度好的优势,但是随机初始化的存在使得ML-ELM的预测表现极易受随机权重的影响,并且层数越多,这种影响越明显,从而导致ML-ELM的预测稳定性相对单层ELM很差。因此,本文在深入分析ELM训练模式的基础上,提出了一种基于隐含层输出矩阵分解的极限学习机(Decomposition Extreme Learning Machine, DE-ELM)模型,并将其应用于对手写字符的识别上。与ML-ELM不同,DE-ELM不需要对手写字符图像进行特征提取,而是通过对大规模隐含层输出矩阵的分解来逼近ELM的输出层权重。相比ML-ELM,DE-ELM降低了基于ELM的手写字符识别模型训练的随机性。同时,在MNIST类数据集(即MNIST^[27], EMNIST^[28], KMNIST^[29]和K49-MNIST^[29])上的比较结果表明,在相同的训练时间下,DE-ELM能够获得优于ML-ELM的HCR识别精度;在相同的识别精度下,DE-ELM又具有明显短于ML-ELM的训练时间。实验结果证实了DE-ELM的可行性以及在处理手写字符识别问题上的有效性。

2 预备知识

2.1 手写字符识别

在执行手写字符识别HCR任务时,我们需要将给定的分辨率为 $a \times b$ 的手写字符图片转化为 $\mathcal{D} = a \times b$ 维的向量。HCR的目的是基于给定的含有 \mathcal{N} 幅属于 \mathcal{M} 个不同类别的手写字符图像数据集。

$$\mathbb{D} = \left\{ \begin{array}{l} (\vec{x}_n, \vec{y}_n) \mid \vec{x}_n = (x_{n1}, x_{n2}, \dots, x_{n\mathcal{D}}) \\ \vec{y}_n = (y_{n1}, y_{n2}, \dots, y_{n\mathcal{M}}) \\ x_{ni} \in \{0, 1, 2, \dots, 255\} \\ y_{nm} \in \{0, 1\} \\ n = 1, 2, \dots, \mathcal{N} \\ d = 1, 2, \dots, \mathcal{D} \\ m = 1, 2, \dots, \mathcal{M} \end{array} \right. \quad (1)$$

训练识别函数 $F(\cdot)$,使其能够预测新的手写字符图像样本 $\vec{x} = (x_1, x_2, \dots, x_{\mathcal{D}})$ 的类别输出。

$$\vec{y} = F(\vec{x}) = (y_1, y_2, \dots, y_{\mathcal{M}}) \quad (2)$$

其中,

$$y_{mm} = \begin{cases} 0, & C(\vec{x}_n) \neq m \\ 1, & C(\vec{x}_n) = m \end{cases} \quad (3)$$

其中, $C(\vec{x}_n)$ 表示手写字符图像 \vec{x}_n 的类别标签。基于预测输出 \vec{y} ,我们可以判断 \vec{x} 的类别标签为:

$$C(\vec{x}) = m = \arg \max_{i=1,2,\dots,\mathcal{M}} \{y_i\} \quad (4)$$

2.2 极限学习机

极限学习机ELM是一种快速的全链接前馈神经网络训练范式,本文基于2.1节给定的训练数据集 \mathbb{D} 对应的输入

矩阵 $\mathbf{X}=(x_{nl})_{A \times \mathcal{Q}}$ 和输出矩阵 $\mathbf{Y}=(y_{ml})_{A \times \mathcal{U}}$ 来介绍 ELM 的工作原理。

首先,随机初始化输入层权重矩阵 $\mathbf{W}=(w_{dl})_{\mathcal{Q} \times \mathcal{L}}$, \mathcal{L} 为隐含层节点的个数。根据式(5)得到隐含层输入矩阵。

$$\mathbf{S}=(s_{nl})_{A \times \mathcal{L}}=\mathbf{XW} \quad (5)$$

在此,为简化讨论,我们不考虑隐含层偏置对网络训练的影响。对隐含层输入矩阵进行激活操作得隐含层输出矩阵

$$\mathbf{H}=(h_{nl})_{A \times \mathcal{L}} \quad (6)$$

$$h_{nl}=G(s_{nl})$$

其中, $G(\cdot)$ 为激活函数,进而可得神经网络的输出矩阵。

$$\mathbf{Y}=\mathbf{HV} \quad (7)$$

其中, $\mathbf{V}=(v_{ml})_{\mathcal{L} \times \mathcal{U}}$ 为输出层权重矩阵,计算式如下:

$$\mathbf{V}=\mathbf{H}^{-1}\mathbf{Y} \quad (8)$$

其中, \mathbf{H}^{-1} 为隐含层输出矩阵 \mathbf{H} 的广义逆矩阵。对于新样本 $\tilde{\mathbf{x}}=(x_1, x_2, \dots, x_{\mathcal{Q}})$, ELM 根据式(9)预测其输出 $\tilde{\mathbf{y}}=(y_1, y_2, \dots, y_{\mathcal{U}})$:

$$\begin{aligned} \tilde{\mathbf{y}} &= \mathbf{G}(\tilde{\mathbf{x}}\mathbf{W})\mathbf{V} \\ &= \mathbf{G}(\tilde{\mathbf{x}}\mathbf{W})\mathbf{H}^{-1}\mathbf{Y} \\ &= \mathbf{G}(\tilde{\mathbf{x}}\mathbf{W})[\mathbf{G}(\mathbf{XW})]^{-1}\mathbf{Y} \end{aligned} \quad (9)$$

至此,我们完成了基于训练数据 \mathbb{D} 训练 ELM 并对未知样本 $\tilde{\mathbf{x}}$ 的输出进行预测的全过程。我们在该过程中发现,ELM 没有使用误差反传式的权重训练策略,而是采用了随机初始化权重、解析求解输出权重的策略来完成对神经网络权重的确定,因此获得了极快的网络训练速度。

2.3 多层极限学习机

多层极限学习机 ML-ELM 是一种将极限自动编码器 (Extreme Learning Machine as an Auto Encoder, ELM-AE)^[24] 和 ELM 相结合的多层神经网络训练范式,ELM-AE 用于对原始输入进行特征提取,ELM 基于提取特征进行有监督学习。含有 K 个隐含层的 ML-ELM 如图 1 所示。

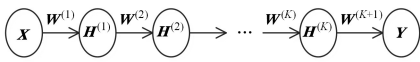


图 1 ML-ELM 网络结构图

Fig. 1 Structure of ML-ELM

图 1 中, $\mathbf{H}^{(k)}$ 是第 $k(k=1, 2, \dots, \mathcal{K})$ 个隐含层的输出矩阵,该隐含层包含 \mathcal{L}_k 个节点, $\mathbf{S}^{(k)}$ 为第 k 个隐含层的输入矩阵,令 $\mathbf{H}^{(0)}=\mathbf{X}$ 和 $\mathcal{L}_0=\mathcal{Q}$ 。ML-ELM 训练的主要目的就是确定 \mathcal{K} 个 ELM-AE 的隐含层权重矩阵 $\mathbf{W}^{(k)}=(w_{dl}^{(k)})_{\mathcal{L}_{k-1} \times \mathcal{L}_k} (k=1, 2, \dots, \mathcal{K})$ 和 1 个 ELM 的隐含层权重矩阵 $\mathbf{W}^{(K+1)}=(w_{dl}^{(K+1)})_{\mathcal{L}_K \times \mathcal{U}}$ 。

$$\mathbf{H}^{(k)}=(h_{nl}^{(k)})_{N \times \mathcal{L}_k}=\mathbf{G}(\mathbf{S}^{(k)}) \quad (10)$$

$$\mathbf{S}^{(k)}=\mathbf{H}^{(k-1)}\mathbf{W}^{(k)} \quad (11)$$

对于第 $k(k=1, 2, \dots, \mathcal{K})$ 个隐含层权重矩阵的确定,构建如式(12)所示的 ELM-AE。

$$\begin{aligned} \mathbf{H}^{(k-1)}\mathbf{W}^{(k)} &= \mathbf{S}^{(k)} \xrightarrow{G(\mathbf{S}^{(k)})} \mathbf{H}^{(k)}\mathbf{V}^{(k)} \\ &= \mathbf{H}^{(k-1)} \end{aligned} \quad (12)$$

进而求解得出:

$$\mathbf{V}^{(k)}=[\mathbf{H}^{(k)}]^{-1}\mathbf{H}^{(k-1)}=[\mathbf{G}(\mathbf{H}^{(k-1)}\mathbf{W}^{(k)})]^{-1}\mathbf{H}^{(k-1)} \quad (13)$$

令:

$$\mathbf{W}^{(k)}=[\mathbf{V}^{(k)}]^T \quad (14)$$

从而求出 ML-ELM 的第 $k(k=1, 2, \dots, \mathcal{K})$ 个隐含层的权重矩阵。

对于隐含层权重矩阵 $\mathbf{W}^{(k+1)}$ 的确定,构建如式(15)所示的 ELM。

$$\begin{aligned} \mathbf{H}^{(k-1)}\mathbf{W}^{(k)} &= \mathbf{S}^{(k)} \xrightarrow{G(\mathbf{S}^{(k)})} \mathbf{H}^{(k)}\mathbf{W}^{(k+1)} \\ &= \mathbf{Y} \end{aligned} \quad (15)$$

进而求解得出:

$$\begin{aligned} \mathbf{W}^{(k+1)} &= [\mathbf{H}^{(k)}]^{-1}\mathbf{Y} \\ &= [\mathbf{G}(\mathbf{H}^{(k-1)}\mathbf{W}^{(k)})]^{-1}\mathbf{H}^{(k-1)} \end{aligned} \quad (16)$$

对于新样本 $\tilde{\mathbf{x}}=(x_1, x_2, \dots, x_{\mathcal{Q}})$, ML-ELM 根据式(17)预测其输出 $\tilde{\mathbf{y}}=(y_1, y_2, \dots, y_{\mathcal{U}})$ 。

$$\tilde{\mathbf{y}}=\mathbf{G}(\mathbf{G}(\mathbf{G}(\tilde{\mathbf{x}}\mathbf{W}^{(1)})\mathbf{W}^{(2)})\dots\mathbf{W}^{(K)})\mathbf{W}^{(K+1)} \quad (17)$$

同样,与 ELM 的训练和预测过程类似,在 ML-ELM 的权重确定过程中,每个隐含层的权重矩阵均采用解析求解的方式确定,从而获得了极快的深度学习网络训练速度。

3 分解极限学习机

3.1 ML-ELM 的随机性分析

尽管现有的研究^[24-26, 30-31] 报告了 ML-ELM 在处理手写字识别以及图像识别方面具有良好的识别效果,但是对于 ML-ELM 的训练而言,由于随机权重的不断加入,随机性在某种程度上严重影响了 ML-ELM 的预测效果,而且这种影响会随着 ML-ELM 层数的增加越来越明显。在此,我们给出了一个数值分析去直观地展示 ML-ELM 随机性是如何随层数的增加而变化的。

不失一般性, $\mathcal{Q}=\mathcal{L}_1=\mathcal{L}_2=\dots=\mathcal{L}_K$ 。

随机生成 \mathcal{K} 个含有 $\mathcal{Q}=\mathcal{Q} \times \mathcal{Q}$ 个一维样本点的数据集 R_1, R_2, \dots, R_K , 其中:

$$R_k=\{r_1^{(k)}, r_2^{(k)}, \dots, r_{\mathcal{Q}}^{(k)}\}, k=1, 2, \dots, \mathcal{K} \quad (18)$$

利用代入熵^[30] 计算 R_k 的随机性(Randomness):

$$E(R_k)=-\frac{1}{\mathcal{Q}} \sum_{p=1}^{\mathcal{Q}} \sum_{q \neq p}^{\mathcal{Q}} \ln f_{-p}(r_p^{(k)}) \quad (19)$$

其中:

$$f_{-p}(r_p^{(k)})=\frac{1}{\mathcal{Q}-1} \times \sum_{\substack{q=1 \\ q \neq p}}^{\mathcal{Q}} \frac{1}{\sqrt{2\pi}h} \exp\left[-\frac{1}{2} \left(\frac{r_p^{(k)}-r_q^{(k)}}{h}\right)^2\right] \quad (20)$$

式(20)为样本 $r_p^{(k)}$ 对应的概率密度函数估计值, $h>0$ 为窗口宽度参数。那么,含有 \mathcal{K} 个隐含层的 ML-ELM 的权重随机性为:

$$E=\sum_{k=1}^{\mathcal{K}} E(R_k) \quad (21)$$

在本验证中,我们分别生成了服从 $[0, 2]$ 均匀分布和标准正态分布的两种类型的随机数,简便起见,令 $h=0.5$ 。图 2 和图 3 分别给出了在两种不同分布的随机数上,不同的 \mathcal{Q} 下,随着 \mathcal{K} 的增加随机性的变化趋势。从图 2 和图 3 中可以发现,对于包含了 \mathcal{K} 个隐含层的 ML-ELM 而言,无论是均匀分布的随机权重还是正态分布的随机权重,随着 \mathcal{K} 值的不断增加,ML-ELM 含有的随机性逐渐增大。现有的研究成果^[32] 表明,学习系统的不确定性在一定程度上影响了其泛化能力。一般而言,学习系统的不确定性越大,其泛化能力越弱;不确定性

越小,其泛化能力越强。随机性作为不确定性的一种表现形式,若能有效对其进行降低,将有助于提升学习系统的泛化性能。

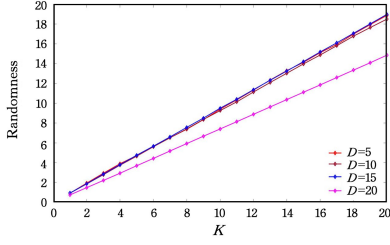


图2 ML-ELM 均匀分布权重的随机性随层数的变化

Fig. 2 Variation tendency of randomness corresponding to uniform weights

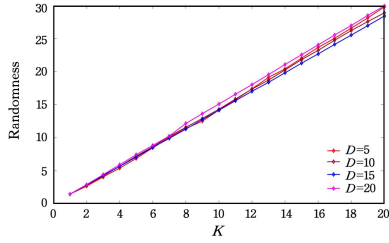


图3 ML-ELM 正态分布权重的随机性随层数的变化

Fig. 3 Variation tendency of randomness corresponding to normal weights

3.2 DE-ELM 的实现原理

本文提出的分解极限学习机 DE-ELM 的工作原理如下。在式(8)中,我们通过求解隐层输出矩阵的伪逆得到了 ELM 的输出层权重矩阵,在提出的 DE-ELM 中,我们利用式(22)计算隐层权重矩阵。

$$\mathbf{V} = \sum_{t=1}^{\mathcal{T}} \alpha_t \mathbf{V}_t \quad (22)$$

其中,矩阵系数 α_t ($t=1, 2, \dots, \mathcal{T}$) 满足:

$$\sum_{t=1}^{\mathcal{T}} \alpha_t = 1 \quad (23)$$

其中, $\mathbf{V}_t = (\mathbf{v}_{im}^{(t)})_{\mathcal{L} \times \mathcal{M}}$ 为基于原始数据集 $\mathbf{X} = (\mathbf{x}_{nd}^{(i)})_{\mathcal{N} \times \mathcal{Q}}$ 的第 t 个分解矩阵 $\mathbf{X}_t = (\mathbf{x}_{nd}^{(t)})_{\mathcal{N}_t \times \mathcal{Q}}$ 训练的 ELM 的输出层权重矩阵。

对于分解矩阵 \mathbf{X}_t , 其满足如下两个条件: 1) $\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_{\mathcal{T}} \end{bmatrix}$;

2) 对于任意 $n \in \{1, 2, \dots, \mathcal{N}\}$, 存在唯一 $t \in \{1, 2, \dots, \mathcal{T}\}$, 使得 $\mathbf{x}_n \in \mathbf{X}_t$ 。另外, 我们要求 \mathcal{N}_t ($t=1, 2, \dots, \mathcal{T}$) 满足 $\mathcal{N}_t = \mathcal{Q}$ 。从 DE-ELM 的工作原理可知, 当 $\mathcal{N} \gg \mathcal{Q}$ 且 \mathcal{Q} 的值也较大时, DE-ELM 的计算效率远高于单层 ELM 对输出层权重矩阵的直接求解方式。同时, 相比 ML-ELM, 由于 DE-ELM 采用的随机权重数量少于 ML-ELM, 因此其随机性更小。为保证上述工作原理的正确性, 我们给出如下的矩阵广义逆分解定理。

定理 1 (广义逆分解定理) 假设矩阵 $\mathbf{H} = (h_m)_{\mathcal{N} \times \mathcal{L}}$ 按行切分成 \mathcal{T} 个子矩阵 $\mathbf{H}_t = (h_m^{(t)})_{\mathcal{N}_t \times \mathcal{L}}$, $t=1, 2, \dots, \mathcal{T}$, 其中 $\mathcal{N}_t = \mathcal{L}$ 且 \mathbf{H}_t 可逆, 那么对于任意的 α_t 满足条件 $\sum_{t=1}^{\mathcal{T}} \alpha_t = 1$, 矩阵 \mathbf{H}^* 为 \mathbf{H} 的广义逆矩阵。

$$\mathbf{H}^* = [\alpha_1 \mathbf{H}_1^{-1} \quad \alpha_2 \mathbf{H}_2^{-1} \quad \dots \quad \alpha_{\mathcal{T}} \mathbf{H}_{\mathcal{T}}^{-1}] \quad (24)$$

证明: 由于 \mathbf{H}^* 满足:

$$\begin{aligned} \mathbf{H}^* \mathbf{H} &= [\alpha_1 \mathbf{H}_1^{-1} \quad \alpha_2 \mathbf{H}_2^{-1} \quad \dots \quad \alpha_{\mathcal{T}} \mathbf{H}_{\mathcal{T}}^{-1}] \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \\ \vdots \\ \mathbf{H}_{\mathcal{T}} \end{bmatrix} \\ &= \alpha_1 \mathbf{H}_1^{-1} \mathbf{H}_1 + \alpha_2 \mathbf{H}_2^{-1} \mathbf{H}_2 + \dots + \alpha_{\mathcal{T}} \mathbf{H}_{\mathcal{T}}^{-1} \mathbf{H}_{\mathcal{T}} \\ &= \left(\sum_{t=1}^{\mathcal{T}} \alpha_t \right) \mathbf{I}_{\mathcal{L} \times \mathcal{L}} \\ &= \mathbf{I}_{\mathcal{L} \times \mathcal{L}} \end{aligned} \quad (25)$$

因此, $\mathbf{H} \mathbf{H}^* \mathbf{H} = \mathbf{H}$ 与 $\mathbf{H}^* \mathbf{H} \mathbf{H}^* = \mathbf{H}^*$ 成立, \mathbf{H}^* 为 \mathbf{H} 的广义逆矩阵。证毕。

对于我们将要进行的手写字符识别任务, HCR 数据集具有样本多和维度高的特点。如果我们直接使用单隐层 ELM, 由于样本量巨大, 隐层输出矩阵广义逆的计算非常耗时; 如果我们使用 ML-ELM 对原始的 HCR 数据集进行特征提取, 由于多次随机权重的使用, 导致 ML-ELM 的随机性不断增加, 高质量特征的提取很不稳定。而我们提出的 DE-ELM 既避免了浅层 ELM 计算量大的缺点, 又未如 ML-ELM 一样增加了学习系统的随机性, DE-ELM 的算法流程如算法 1 所示。

算法 1 DE-ELM 算法

输入: HCR 数据集 \mathbb{D} 对应的输入矩阵 $\mathbf{X} = (\mathbf{x}_{nd})_{\mathcal{N} \times \mathcal{Q}}$ 和输出矩阵 $\mathbf{Y} = (\mathbf{y}_{nm})_{\mathcal{N} \times \mathcal{M}}$, ELM 隐层节点个数 \mathcal{L}

输出: ELM 隐层权重矩阵 $\mathbf{V} = (\mathbf{v}_{im})_{\mathcal{L} \times \mathcal{M}}$

Step 1 将 \mathbf{X} 按行分割成 \mathcal{T} 个子矩阵 $\mathbf{X}_t = (\mathbf{x}_{nd}^{(t)})_{\mathcal{N}_t \times \mathcal{Q}}$, $t=1, 2, \dots, \mathcal{T}$, 其中 $\mathcal{N}_t = \mathcal{L}$;

Step 2 与 \mathbf{X}_t 对应地将 $\mathbf{Y} = (\mathbf{y}_{nm})_{\mathcal{N} \times \mathcal{M}}$ 按行分割成 \mathcal{T} 个子矩阵 $\mathbf{Y}_t = (\mathbf{y}_{nm}^{(t)})_{\mathcal{N}_t \times \mathcal{M}}$, $t=1, 2, \dots, \mathcal{T}$;

Step 3 初始化随机权重矩阵 $\mathbf{W} = (\mathbf{w}_{ai})_{\mathcal{Q} \times \mathcal{L}}$;

Step 4 对于每一个分解输入矩阵 \mathbf{X}_t 和分解输出矩阵 \mathbf{Y}_t , \mathbf{W} 训练 ELM, 得输出层权重矩阵 $\mathbf{V}_t = (\mathbf{v}_{im}^{(t)})_{\mathcal{L} \times \mathcal{M}}$;

Step 5 按照式(22)计算 ELM 隐层权重矩阵 \mathbf{V} 。

对于算法 1 中的关键步骤, 我们进行如下的解释说明:

(1) 由式(8)和定义 1 可得:

$$\mathbf{V} = \mathbf{H}^{-1} \mathbf{Y}$$

$$\begin{aligned} &= [\alpha_1 \mathbf{H}_1^{-1} \quad \alpha_2 \mathbf{H}_2^{-1} \quad \dots \quad \alpha_{\mathcal{T}} \mathbf{H}_{\mathcal{T}}^{-1}] \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_{\mathcal{T}} \end{bmatrix} \\ &= \alpha_1 \mathbf{H}_1^{-1} \mathbf{Y}_1 + \alpha_2 \mathbf{H}_2^{-1} \mathbf{Y}_2 + \dots + \alpha_{\mathcal{T}} \mathbf{H}_{\mathcal{T}}^{-1} \mathbf{Y}_{\mathcal{T}} \\ &= \sum_{t=1}^{\mathcal{T}} \alpha_t \mathbf{V}_t \end{aligned} \quad (26)$$

从而证实了 DE-ELM 的收敛性;

(2) 对于 $\mathbf{V}_t = (\mathbf{v}_{im}^{(t)})_{\mathcal{L} \times \mathcal{M}}$, $t=1, 2, \dots, \mathcal{T}$ 的求解, 我们仅初始化一次随机权重, 从而保证了随机性不随着矩阵的分解而增加;

(3) 为保证 $\mathcal{N}_t = \mathcal{L}$, 即 $\mathcal{N} = \mathcal{L} \times \mathcal{T}$, 我们可以删除训练集中的 $\mathcal{N} - \left\lfloor \frac{\mathcal{N}}{\mathcal{T}} \right\rfloor \times \mathcal{T}$ 个样本, 由于 HCR 数据集规模很大, 增加或者删除若干样本对识别模型预测精度的影响不大;

(4) 对于 \mathbf{X}_t 对应的 ELM 的隐层输出矩阵 \mathbf{H}_t , 在具体

的计算过程中,当 H_i 不可逆时,令:

$$H_i = H_i + \lambda I_{l \times l} \quad (27)$$

以保证 H_i 的可逆性,其中可逆参数 λ 为一个极小正实数。

4 实验验证

4.1 实验数据集和运行环境

为了验证 DE-ELM 的可行性和有效性,本文基于 4 个经典的手写字符识别 HCR 数据集 (MNIST^[27], EMNIST^[28], KMNIST^[29] 和 K49-MNIST^[29]) 测试了分解矩阵个数对 DE-ELM 训练时间和识别准确率的影响以及比较了 DE-ELM, ELM^[28] 与 ML-ELM 的 HCR 准确率。表 1 列出了 4 个 HCR 数据集的详细信息。

(1) MNIST 数据集来自美国国家标准与技术研究所,训练集和测试集均由 250 个不同的人手写的数字构成。

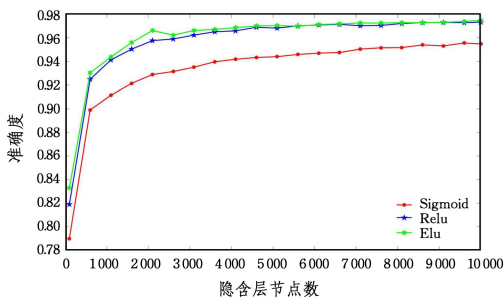
(2) EMNIST 数据集是一个拓展的 MNIST,在手写数字的基础上加入了手写的大小写英文字母,对难以分辨的字母手写体进行合并,共计 47 类。

(3) KMNIST 和 K49-MNIST 数据集都是由日文的平假名字符构成, KMNIST 包括 10 个平假名字符的手写体,共 10 类;而 K49-MNIST 中包括了 48 个平假名字符和一个平假名迭代标记的手写体,共 49 类。

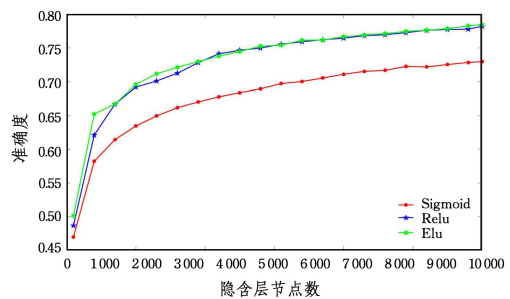
表 1 4 个手写字符识别数据集

Table 1 Details of 4 HCR data sets

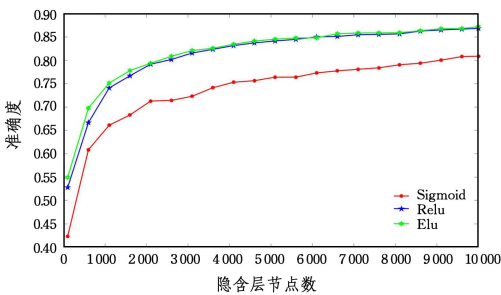
数据集	样本总数	训练集 规模	测试集 规模	图片 分辨率	类别 个数
MNIST	70 000	60 000	10 000	28×28	10
EMNIST	131 600	112 800	18 800	28×28	47
KMNIST	70 000	60 000	10 000	28×28	10
K49-MNIST	270 912	232 365	38 547	28×28	49



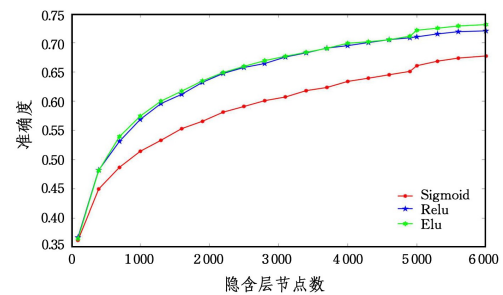
(a) MNIST 数据集上的准确率



(b) EMNIST 数据集上的准确率



(c) KMNIST 数据集上的准确率



(d) K49-MNIST 数据集上的准确率

图 4 不同激活函数对 DE-ELM 手写字符识别准确率的影响

Fig. 4 Impact of activation function on HCR accuracy of DE-ELM

所有的实验均运行在配置了 Linux16.04 操作系统、拥有 Intel(R) 2.40 GHz E5-2680 v4 CPU(14 核、28 线程)、128 GB 内存的服务器上。ELM, ML-ELM 以及 DE-ELM 算法均采用 Python 编程语言实现,其中 DE-ELM 算法的代码可通过百度网盘¹⁾下载。简便起见,算法 1 中式(22)的矩阵系数 $\alpha_1 = \alpha_2 = \dots = \alpha_j = \frac{1}{j}$ 。

4.2 实验结果及分析

在此,我们进行了两部分实验验证:1) DE-ELM 的可行性验证;2) DE-ELM 的有效性验证。在第一部分的实验中,我们首先验证了不同的隐含层激活函数对 DE-ELM 手写字符识别准确率的影响。在此,我们考虑了 3 种不同的激活函数,分别如下。

(1) Sigmoid 激活函数

$$G(x) = (1 + e^{-x})^{-1}, x \in (-\infty, +\infty) \quad (28)$$

(2) Relu 激活函数

$$G(x) = \max\{0, x\}, x \in (-\infty, +\infty) \quad (29)$$

(3) Elu 激活函数

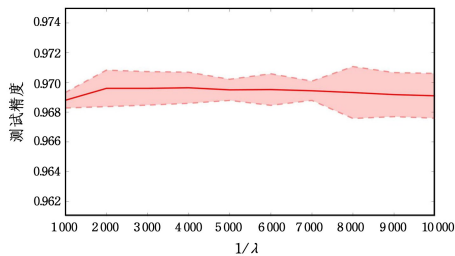
$$G(x) = \max\{0, x\} + \min\{0, a(e^x - 1)\}, x \in (-\infty, +\infty) \quad (30)$$

图 4 给出了随着隐含层节点个数的增加,对应着不同激活函数的 DE-ELM 在 4 个 HCR 数据集上的识别准确率的变化情况。从图 4 中可以发现,采用了 Relu 激活函数和 Elu 激活函数的 DE-ELM 的表现明显优于采用了 Sigmoid 激活函数的 DE-ELM;同时,采用了 Elu 激活函数的 DE-ELM 的表现略优于采用了 Relu 激活函数的 DE-ELM。因此,在后续的实验中所提 DE-ELM 均采用 Elu 激活函数。

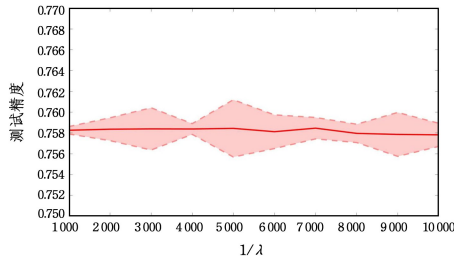
¹⁾ 手写字符识别 DE-ELM 算法(提取码:p48c); <https://pan.baidu.com/s/1pjRvkci1xQc9RFNWBwd84Q>

在采用了 Elu 激活函数后,我们在 1 00010 000 之间以步长 1 000 取可逆参数 $1/\lambda$ 来对 DE-ELM 的性能进行研究。从

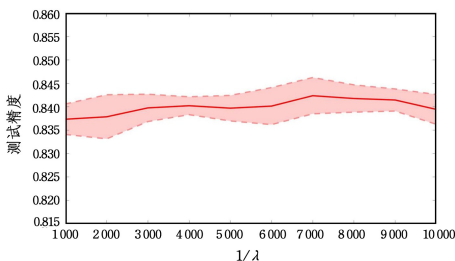
图 5 可知,在 4 个数据集下,可逆参数 λ 的取值对 DE-ELM 性能的影响并不大,因此在后期实验中我们均取 $1/\lambda=1 000$ 。



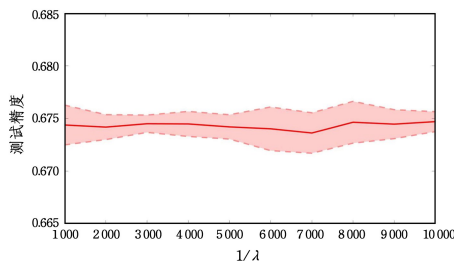
(a)MNIST 数据集上的实验结果



(b)EMNIST 数据集上的实验结果



(c)KMNIST 数据集上的实验结果

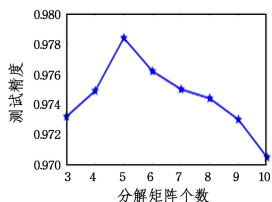


(d)K49-MNIST 数据集上的实验结果

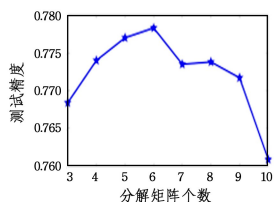
图 5 可逆参数 λ 对 DE-ELM 手写字符识别准确率的影响Fig. 5 Impact of invertible parameter λ on HCR accuracy of DE-ELM

接着,我们验证了不同的隐含层输出矩阵分解矩阵个数 \mathcal{T} 对 DE-ELM 训练时间和 HCR 准确率的影响,实验结果如图 6 所示。由于隐含层节点数 \mathcal{L} 与 \mathcal{T} 之间存在关系,即 $\mathcal{N}=\mathcal{L}\times\mathcal{T}$,因此在本实验中对于给定的 \mathcal{T} ,令 $\mathcal{L}=\lfloor\frac{\mathcal{N}}{\mathcal{T}}\rfloor$ 。从图 6 中可以发现,随着 \mathcal{T} 的增加,DE-ELM 的 HCR 准确率呈现先提高

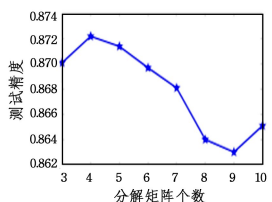
后下降的趋势。这说明并不是对原始输入矩阵或者隐含层输出矩阵分解得越细越好,当超过 DE-ELM 支持的最优分解之后,越细的分解反而会导致 DE-ELM 识别准确率的降低。基于图 6 的实验结果,在后续的实验,针对 MNIST,EMNIST,KMNIST 和 K49-MNIST 这 4 个 HCR 数据集,DE-ELM 的最优分解矩阵个数分别为 5,6,4 和 6。



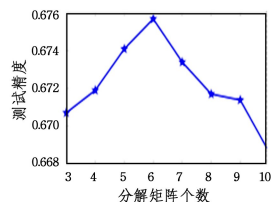
(a)MNIST 数据集上的实验结果



(b)EMNIST 数据集上的实验结果



(c)KMNIST 数据集上的实验结果

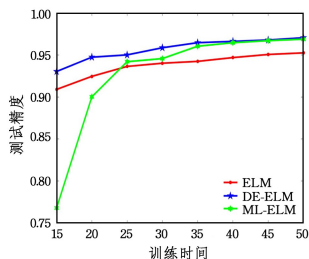


(d)K49-MNIST 数据集上的实验结果

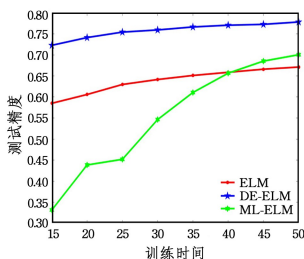
图 6 分解矩阵个数 \mathcal{T} 对 DE-ELM 的 HCR 准确率的影响Fig. 6 Impact of \mathcal{T} on HCR accuracy of DE-ELM

在第二部分的实验中,我们比较了 ELM,ML-ELM 和 DE-ELM 在 MNIST,EMNIST,KMNIST 和 K49-MNIST 这 4 个 HCR 数据集上的识别准确率。为了体现比较的公平性,我们对比了 3 种方法在相同训练时间下的测试精度以及在相同测

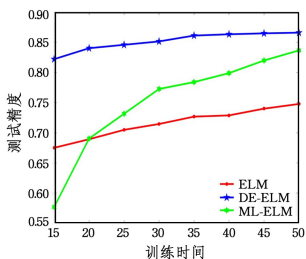
试精度下的训练时间,其中 ML-ELM 的网络结构为 \mathcal{Q} -700-500- \mathcal{L} - \mathcal{M} ,实验统计时仅考虑 \mathcal{L} 的变化对 ML-ELM 识别准确率的影响。相同训练时间下测试精度的对比结果如图 7 和表 2 所示,相同测试精度下训练时间的对比结果如图 8 和表 3 所示。



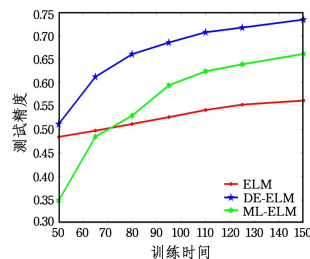
(a)MNIST 数据集上的实验结果



(b)EMNIST 数据集上的实验结果



(c)KMNIST 数据集上的实验结果



(d)K49-MNIST 数据集上的实验结果

图 7 相同训练时间下 ELM,ML-ELM 和 DE-ELM 的 HCR 准确率比较

Fig. 7 Comparison of HCR accuracies among ELM,ML-ELM and DE-ELM with the same training time

表 2 50s 训练时间下 ELM, ML-ELM 和 DE-ELM 的 HCR 准确率比较

Table 2 Comparison of HCR accuracies among ELM, ML-ELM and DE-ELM with the training time of 50 s

	MNIST(50s±1s)			EMNIST(50s±1s)			KMNIST(50s±1s)			K49-MNIST(200s±2s)		
	隐层节点	训练精度/%	测试精度/%	隐层节点	训练精度/%	测试精度/%	隐层节点	训练精度/%	测试精度/%	隐层节点	训练精度/%	测试精度/%
ELM	1720	95.49	95.24	950	69.71	68.00	1500	88.92	74.77	1550	72.58	58.76
ML-ELM	2900	97.91	96.87	850	70.64	69.41	2300	94.48	83.95	3200	83.91	71.28
DE-ELM	4850	98.12	97.04	3000	78.17	75.82	4600	95.14	84.13	6300	85.57	73.14

从图 7 中可以发现,随着训练时间的增加(即隐含层节点个数的不断增加),3 种 HCR 模型均呈现收敛的趋势:手写字符识别准确率逐渐提高而后趋于平稳。同时还可以发现,在相同的训练时间下,本文提出的 DE-ELM 获得了最高的 HCR 准确率,这表明相比单个大规模隐含层输出矩阵求解广义逆的 ELM 和多个隐含层广义逆求解的 ML-ELM, DE-ELM 具有更好的手写字符识别效果。表 2 列出了在训练时间 50s 下测试精度的具体量化比较结果,从中可以看到,DE-ELM 在这 4 个 HCR 数据集上比 ELM 和 ML-ELM 使用了更多的隐含层节点,也就是说 DE-ELM 能够支持更多隐含层节点的网络模型构建,从而获得了更高的 HCR 准确率。

从图 8 中可以发现,随着测试精度的提高,3 种 HCR 模型所需要的训练时间逐渐增加。但是,相比 ELM 和 ML-ELM,本文提出的 DE-ELM 在同等测试精度下所需要的训练时间是最短的。随着测试精度的提高,ELM 的训练时间增幅最大,这是因为 ELM 需要更多的隐含层节点才能达到和 DE-ELM 相等的测试精度;ML-ELM 训练时间的增幅虽然没有 ELM 明显,但仍然高于 DE-ELM。表 3 列出了针对 4 个 HCR 数据集在给定的测试精度下训练时间的具体量化比较结果。综上所述可以发现,本文提出的 DE-ELM 手写字符识别模型是可行的,同时也是有效的。

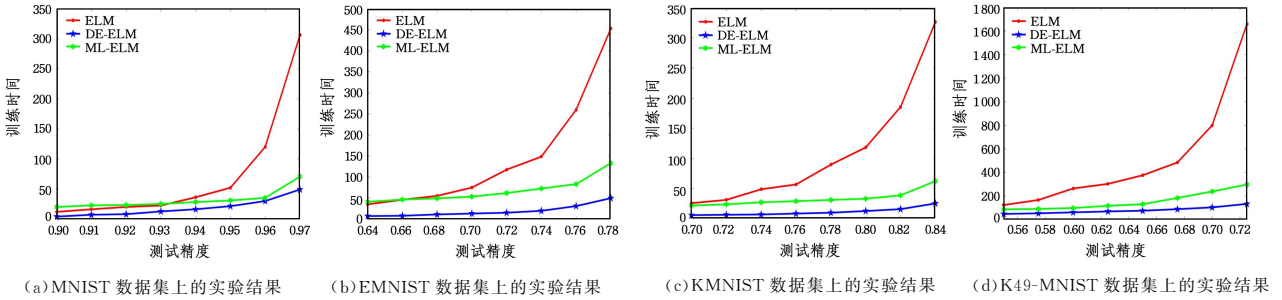


图 8 相同测试精度下 ELM, ML-ELM 和 DE-ELM 的训练时间比较

Fig. 8 Comparison of training times among ELM, ML-ELM and DE-ELM with the same HCR accuracies

表 3 给定的相同测试精度下 ELM, ML-ELM 和 DE-ELM 的训练时间比较

Table 3 Comparison of training times among ELM, ML-ELM and DE-ELM with the given HCR accuracies

	MNIST(97%±0.2%)			EMNIST(78%±0.2%)			KMNIST(86%±0.2%)			K49-MNIST(72.5%±0.2%)		
	隐层节点	训练时间/%	测试时间/%	隐层节点	训练时间/%	测试时间/%	隐层节点	训练时间/%	测试时间/%	隐层节点	训练时间/%	测试时间/%
ELM	6100	306.085	4.228	4800	452.982	6.333	8600	660.768	5.873	7000	1658.65	18.318
ML-ELM	4000	78.648	3.287	3500	106.367	5.550	5000	95.287	3.940	3700	227.685	11.930
DE-ELM	4800	52.159	3.163	5200	83.982	6.761	8100	83.776	5.351	6100	192.351	15.282

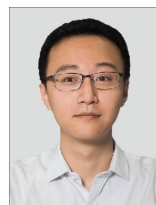
结束语 利用随机权的思想构建深度神经网络进行图像识别,是当前深度学习领域研究的一个热点,本文在对一种特殊的随机权深度神经网络,即多层极限学习机进行深入分析的基础上,提出了一种用于手写字符识别的分解极限学习机。DE-ELM 是一种浅层的 ELM 模型,它并未像 ML-ELM 一样通过构建深层的学习系统对手写字符数据集进行特征提取进而进行识别,而是通过将 ELM 的隐含层输出矩阵进行分解,利用小规模可逆矩阵的逆的加权和逼近大规模隐含层输出矩阵的广义逆,从而起到降低广义逆求解的时间复杂度。这样做的好处在于,相比 ML-ELM, DE-ELM 的训练过程有更少的随机性引入,从而能够提升手写字符识别的准确率和稳定性。在 4 个标准的手写字符识别数据集上,我们对比了 DE-ELM 与 ELM 和 ML-ELM 的识别准确率,实验结果表明,所提 DE-ELM 在相同训练时间下能够获得更高的识别精度;在相同的识别精度下具有明显短于另外两者的训练时间,从而

证实了 DE-ELM 的可行性以及在处理手写字符识别问题上的有效性。

参考文献

- [1] 全国科学技术名词审定委员会术语在线 [EB/OL]. [2020-12-20]. <https://www.termonline.cn/search?k=手写字符识别&r=1608471572108>.
- [2] MUHAMMAD'ARIFMOHAMAD D N, HASSAN H, HARON H. A review on feature extraction and feature selection for handwritten character recognition [J]. International Journal of Advanced Computer Science and Applications, 2015, 6(2): 204-212.
- [3] KUMAR M, JINDAL M K, SHARMA R K. k-nearest neighbor based offline handwritten Gurmukhi character recognition [C]//Proceedings of International Conference on Image Infor-

- mation Processing. IEEE,2011;1-4.
- [4] SHANTHI N,DURAIWAMY K. A novel SVM-based handwritten Tamil character recognition system [J]. Pattern Analysis and Applications,2010,13(2):173-180.
- [5] PATEL C I,PATEL R,PATEL P. Handwritten character recognition using neural network [J]. International Journal of Scientific & Engineering Research,2011,2(5):1-6.
- [6] SHRIDHAR M,BADRELDIN A. A high-accuracy syntactic recognition algorithm for handwritten numerals [J]. IEEE Transactions on Systems, Man, and Cybernetics, 1985, 1: 152-158.
- [7] LECUN Y,BOSER B,DENKER J S,et al. Backpropagation applied to handwritten zip code recognition [J]. Neural Computation,1989,1(4):541-551.
- [8] LECUN Y,BOTTOU L,BENGIO Y,et al. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE,1998,86(11):2278-2324.
- [9] HINTON G E,OSINDERO S,TEH Y W. A fast learning algorithm for deep belief nets [J]. Neural Computation, 2006, 18(7):1527-1554.
- [10] SALAKHUTDINOV R,LAROCHELLE H. Efficient learning of deep Boltzmann machines [C]// In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, 2010:693-700.
- [11] HUANG G B,ZHU Q Y,SIEW C K. Extreme learning machine: theory and applications [J]. Neurocomputing, 2006, 70(1/2/3):489-501.
- [12] HUANG G B,ZHOU H,DING X,et al. Extreme learning machine for regression and multiclass classification [J]. IEEE Transactions on Systems,Man, and Cybernetics, Part B(Cybernetics),2011,42(2):513-529.
- [13] HUANG G B,CHEN L,SIEW C K. Universal approximation using incremental constructive feedforward networks with random hidden nodes [J]. IEEE Transactions on Neural Networks, 2006,17(4):879-892.
- [14] HUANG G B,DING X,ZHOU H. Optimization method based extreme learning machine for classification [J]. Neurocomputing,2010,74(1/2/3):155-163.
- [15] LEKAMALAGE C K,LIU T,YANG Y,et al. Extreme learning machine for clustering [C]// Proceedings of ELM-2014. Springer,2015:435-444.
- [16] HUANG G,SONG S,GUPTA J N,et al. Semi-supervised and unsupervised extreme learning machines [J]. IEEE Transactions on Cybernetics,2014,44(12):2405-2417.
- [17] PAN J,WANG X,CHENG Y,et al. Reinforcement learning based on extreme learning machine [C]// Proceedings of International Conference on Intelligent Computing. Springer,2012:80-86.
- [18] CAO F,LIU B,PARK D S. Image classification based on effective extreme learning machine [J/OL]. Neurocomputing, 2013, 102:90-97. <https://doi.org/10.1016/j.neucom.2012.02.042>.
- [19] SCHMIDT W F,KRAAIJVELD M A,DUIN R P. Feed forward neural networks with random weights [C] // Proceedings of International Conference on Pattern Recognition. IEEE,1992:1-4.
- [20] IGELNIK B,PAO Y H. Stochastic choice of basis functions in adaptive function approximation and the functional-link net [J]. IEEE transactions on Neural Networks,1995,6(6):1320-1329.
- [21] HUANG G, HUANG G B, SONG S, et al. Trends in extreme learning machines: A review [J]. Neural Networks, 2015, 61(1):32-48.
- [22] DING S,ZHAO H,ZHANG Y,et al. Extreme learning machine:algorithm, theory and applications [J]. Artificial Intelligence Review,2015,44(1):103-115.
- [23] CAO W,WANG X,MING Z,et al. A review on neural networks with random weights [J/OL]. Neurocomputing,2018,275:278-287. <https://doi.org/10.1016/j.neucom.2017.08.040>.
- [24] KASUN L L,ZHOU H,HUANG G B,et al. Representational learning with extreme learning machine for big data [J]. IEEE Intelligent Systems,2013,28(6):31-34.
- [25] DING S,ZHANG N,XU X,et al. Deep extreme learning machine and its application in EEG classification [J/OL]. Mathematical Problems in Engineering,2015,2015(129021). <https://www.hindawi.com/journals/mpe/2015/129021>.
- [26] TANG J X,DENG C,HUANG G B. Extreme learning machine for multilayer perceptron [J]. IEEE Transactions on Neural Networks and Learning Systems,2015,27(4):809-821.
- [27] KUSSUL E,BAIDYK T. Improved method of handwritten digit recognition tested on MNIST database [J]. Image and Vision Computing,2004,22(12):971-981.
- [28] COHEN G,AFSHAR S,TAPSON J,et al. EMNIST:Extending MNIST to handwritten letters [C]// Proceedings of International Joint Conference on Neural Networks. IEEE,2017:2921-2926.
- [29] CLANUWAT T,BOBER-IRIZAR M,KITAMOTO A,et al. Deep learning for classical Japanese literature [J]. arXiv:1812.01718,2018.
- [30] ZHAO Z T,ZHENG X D. Feature Incremental Extreme Learning Machine[J]. Computer Science,2019,46(11A):112-116.
- [31] XU R,LIANG X,MA R F,et al. Adaptive Orthogonal Search for Network Structure Learning of ELM[J]. Chinese Journal of Computer,2021,44(9):1888-1906.
- [32] HE Y L,LIU J N,WANG X Z,et al. Optimal bandwidth selection for re-substitution entropy estimation [J]. Applied Mathematics and Computation,2012,219(8):3425-3260.



HE Yu-lin, born in 1982, Ph.D, research associate, is a member of China Computer Federation. His main research interests include big data approximate computing technologies, multi-sample statistics theories and methods, data

mining and machine algorithms and applications.