计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210100001

基于自注意力的自监督深度聚类算法

韩 洁[1] 陈俊芬[1] 李 艳[2] 湛泽聪[1]
1 河北大学数学与信息科学学院河北省机器学习与计算智能重点实验室 河北 保定 071002
2 北京师范大学珠海分校应用数学学院 广东 珠海 519087
(lzyhj0124@163.com)

摘 要 近年来,基于联合训练的深度聚类方法,如DEC(Deep Embedding Clustering)和DDC(Deep Denoising Clustering)算法,使基于特征提取的图像聚类取得了很多新进展,带来了聚类性能的突破,而且特征提取环节对后续聚类任务有直接影响。但是,这些方法的泛化能力较差,在不同数据集使用不同的网络结构,聚类性能相比分类性能仍有很大的提升空间。为此,文中提出了一种基于自注意力的自监督深度聚类方法(Self-attention Based Self-supervised Deep Clustering,SADC)。首先设计一个深度卷积自编码器用于提取特征,并且用带噪声的输入数据训练该网络来增强模型的鲁棒性;其次引入自注意力机制,辅助网络捕获对聚类有用的信息;最后编码器部分结合K-means算法形成一个深度聚类器,用于进行特征表示和聚类分配,通过迭代更新网络参数来提高聚类精度和网络的泛化能力。在6个图像数据集上验证所提聚类算法的性能,并与深度聚类算法DEC,DDC等进行比较。实验结果表明,SADC能提供令人满意的聚类结果,而且聚类性能与DEC和DDC相当。总之,统一的网络结构在保证聚类精度的同时降低了深度聚类算法的复杂度。

关键词:深度卷积自编码器;图像聚类;特征表示;自注意力;计算复杂度
中图法分类号 TP181

Self-supervised Deep Clustering Algorithm Based on Self-attention

HAN Jie[1],CHEN Jun-fen[1],LI Yan[2] and ZHAN Ze-cong[1]
1 Hebei Key Laboratory of Machine Learning and Computational Intelligence,College of Mathematics and Information Sciences,Hebei University,Baoding,Hebei 071002,China
2 School of Applied Mathematics,Beijing Normal University Zhuhai,Zhuhai,Guangdong 519087,China

Abstract In recent years,deep clustering methods using joint optimization strategy,such as DEC (deep embedding clustering) and DDC (deep denoising clustering) algorithms,have made great progress in image clustering that heavily related to features representation ability of deep networks,and brought certain degree breakthroughs in clustering performances. The quality of feature extraction directly affects the subsequent clustering tasks. However,the generalization abilities of these methods are not satisfied,exactly as different network structures are used in different datasets to guarantee the clustering performance. In addition,there is a quite larger space to enhance clustering performances compared to classification performances. To this end,a self-supervised deep clustering (SADC) method based on self-attention is proposed. Firstly,a deep convolutional autoencoder is designed to extract features,and noisy images are employed to enhance the robustness of the network. Secondly,self-attention mechanism is combined with the proposed network to capture useful features for clustering. At last,the trained encoder combines with K-means algorithm to form a deep clustering model for feature representation and clustering assignment,and iteratively updates parameters to improve the clustering accuracy and generalization ability of the proposed network. The proposed clustering method is verified on 6 traditional image datasets and compared with the deep clustering algorithms DEC and DDC. Experimental results show that the proposed SADC can provide better clustering results,and is comparable to the state-of-the-art clustering algorithms. Overall,the unified network structure ensures the clustering accuracy and simultaneously reducing computational complexity of the deep clustering algorithms.

Keywords Deep convolutional autoencoder,Image clustering,Features representation,Self-attention,Computational complexity

到稿日期:2021-01-03 返修日期:2021-07-08 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:河北省引进留学人员资助项目(C20200302);河北省自然科学基金(F2018201096);广东省自然科学基金(2018A0303130026);河北省社会科学基金项目(HB20TQ005)
This work was supported by the Hebei Province Introduction of Studying Abroad Talent Funded Project(C20200302),Natural Science Foundation of Hebei Province(F2018201096),Natural Science Foundation of Guangdong Province(2018A0303130026) and Social Science Foundation of Hebei Province(HB20TQ005).
通信作者:陈俊芬(chenjunfen2010@126.com)

1　引言

监督学习任务对带标记的样本的数量和质量要求较高，然而现实中无法获得大量带标记的样本去训练模型。当带标记的样本数量有限而未标记的样本非常丰富时，可以选择半监督学习。如果没有带标记的样本，则只能用无监督学习和自监督学习完成学习任务。自监督学习定义对目标任务有帮助的辅助任务时，可以直接根据原始数据来构建，这样还可以保持良好的可移植性[1]。图像聚类分析是在无任何先验信息的条件下，根据预先定义好的相似性度量，在数据中发现样本及样本之间的关系，将样本划分成不相交的若干类簇[2]，最终使得同一类簇中样本的相似度较高，不同类簇之间的样本的相似度较低。

2016 年，AlphaGo 战胜李世石的消息再度掀起了人工智能和深度学习的热潮，很多人开始重新认识深度神经网络这个概念。神经网络最早于 1943 年由 Mcculloch 等[3]提出，卷积神经网络于 1998 年由 Lecun 等[4]发明，通过区域内权值共享来减少网络训练的参数，降低网络结构的复杂性。虽然卷积网络是图像处理和机器视觉的基础，但是缺乏旋转不变性[5]。注意力机制[6]可一定程度地弥补小卷积核的不足，自注意力机制[7]是一种可微的软注意力[8]，可通过神经网络算出梯度，并且利用前向传播和后向反馈来学习得到注意力的权重，更加擅长捕捉数据或特征内部的相关性。

早期的深度聚类方法如 DEN[9]（Deep Embedding Network），DCN[10]（Deep Clustering Network）等将特征提取与聚类分析分开执行，无法很好地处理图像数据的聚类问题。后来，学者提出了基于联合训练的聚类方法，如 DEC[11]，DBC[12]（Discriminatively Boosted Clustering）和 DDC[13]等，进一步提升了深度网络的聚类性能。在 DEC 中采用全连接层的自编码器进行特征提取，没有使用卷积层，使得图像原本的位置信息丢失。DBC 聚类算法对此做出了改进，将全连接层替换为卷积层，且每个卷积层中添加了批归一化层，首次使用端到端的方式来训练自编码器，针对不同数据集设计不同的网络结构进行训练。DDC 聚类算法使用了添加自注意力机制的降噪卷积自编码器，采用了不同时数据集对应不同的网络结构，且添加自注意力层的位置也不相同。另外，IDEC_DCNN 算法[14]（Improved Deep Embedded Clustering Based on Deep Convolutional Neural Network）将原 DEC 聚类算法的全连接层替换为卷积层，并添加了两个全连接层，还将损失函数调整为重构损失和聚类损失之和，使得聚类性能得到提升。联合学习节点嵌入和聚类分配的 DCN[15]聚类算法是一种新的非属性图的无监督深度节点聚类方法，该方法通过图模型联合学习嵌入和节点聚类来完成深度聚类。

深度神经网络可被看作一个连续且几乎处处可微的多元函数，它将高维输入非线性映射成抽象低维表示，得到对下游任务有用的特征表示。然而，网络结构的灵活性和黑箱性使得网络抽取特征的普适性受到极大的挑战。如何捕获图像数据的非线性关系是关键问题[16]。另外在工业和商业场景的实际应用中，工程师和开发人员经常面临时间预算限制的要求[17]。为此，本文使用深度卷积自编码器结合自注意力机制的统一网络结构对图像数据进行深度聚类分析，在保证模型效果的同时降低网络的时间和空间复杂度。在预训练时利用自编码器的重构损失优化网络参数，在联合训练时使用迭代优化相对熵（KL 散度）损失函数来调整网络参数和 K-means 聚类结果，同时对数据进行特征学习和聚类指派。本文的主要贡献如下：

（1）使用带卷积层和全连接层的自编码器，保留图片中的拓扑信息，又能进行全局特征提取。

（2）利用自注意力机制提高网络特征的提取能力，更加关注有用信息。

（3）统一网络结构，提升了网络的泛化能力，避免针对不同数据集使用不同网络结构的弊端。

（4）使用小卷积核，并改变步长的 Stride 和 Padding 层，降低网络的时间复杂度。

2　相关工作

使用无监督方法时，模型网络的初始化对后面的研究起到了重要的作用，网络初始化较好可以提高网络学习的准确率，减少迭代的次数。无监督离散表示学习的任务是获得一个函数并将相似的数据映射成相似的离散表示[18]，选用无监督预训练对网络进行初始化时，用所有的数据训练自编码器[19]（Autoencoder，AE），将自编码器网络的参数作为网络的初始参数。自编码器是嵌入方法的主要解决方案，并且在深度学习中已经得到广泛应用，推动了深度学习在无监督学习领域的发展，如聚类等无监督的学习任务。

传统的自编码器要求输入和输出相同，一般用于降维和特征提取。后来，Vincent 等[20]提出的降噪自编码器进一步提升了编码器网络的稳定性，在降噪自编码器训练的过程中，输入添加噪声的数据进行训练，最终使得解码器的输出和未添加噪声的数据相同，从而得到更好的抗噪能力以提高聚类精度。

根据特征提取和聚类过程可以把当前的深度聚类算法分为两类。一类是先进行特征提取，再进行聚类分析，如 Huang 等[9]于 2014 年提出的深度嵌入聚类网络 DEN 和 Yang 等[10]于 2016 年提出的深度聚类网络 DCN。DEN 利用深度自编码器，通过最小化数据重构误差，从原始数据中获得良好的表示；为了从已有的表示中揭示内在的流形，应用了保持原始数据局部结构性质的局部保持约束；为了进一步促进聚类并使表示包含聚类信息，还使用了一组稀疏约束来对角化表示的相关性，但是其忽略了提取的特征是否有利于聚类分析。DCN 同时进行网络参数学习和无监督聚类，通过自编码器进行降维和特征提取，将特征输入 K-means 模型进行聚类，而解码器则对特征进行还原，使得特征重构成原始数据，从而使网络更加关注自编码器的优化，忽略了编码器部分提取的特征是否真正有利于聚类。这两种方法的特征提取

部分与聚类分析部分分开，因此编码器提取出来的特征不一定能产生良好的聚类结果。

另一类是特征提取和聚类同时进行。Xie 等[11]于 2016 年提出深度嵌入聚类算法 DEC。DEC 算法使用堆叠自编码器[20]进行预训练，并使用聚类分配强化损失作为规范。首先通过重构损失训练自编码器，丢弃解码器部分。编码器网络提取的特征用于聚类模块的输入。之后，使用聚类分配强化损失对网络进行微调。同时，通过最小化软标签的分布和辅助目标分布之间的 KL 散度来迭代优化聚类。但 DEC 没有尊重数据样本的规定，没有考虑微调会扭曲嵌入式的空间，因此削弱了嵌入式特征的代表性，从而影响了聚类效果[14]。Dizaji 等[21]于 2017 年提出深度嵌入正则聚类模型 DEPICT (Deep Embedded Regularized Clustering)。该模型采用类似于 DEC 算法的整体思路，使用多项逻辑回归函数(即 Softmax 函数)来计算特征点与聚类中心点的相似性；加入编码器的重构损失正则约束 KL 散度来避免深度聚类模型过拟合。Li 等[12]于 2018 年提出判别增强图像聚类方法 DBC。DBC 网络对 DEC 进行了改进，DEC 中使用的是堆叠自编码器，而 DBC 中使用的是全卷积自编码器，聚类结果有了一定的改善，但是网络泛化能力较差，针对不同数据集采用了不同的网络结构。Chen 等[13]于 2019 年提出 DDC 聚类算法，DDC 采用卷积降噪自编结合自注意力机制的方法，增强了网络提取特征的能力，但是同样存在针对不同数据集采用不同网络结构进行训练的问题。

Lin 等[22]于 2017 年首次提出自注意力机制，用于解决自然语言处理的嵌入问题，与先前的注意力机制[6]不同的是，自注意力机制不需要外部信息，仅通过自身的信息来更新学习参数。Wang 等[23]于 2018 年对自注意力机制进行了应用，卷积是经典的局部特征提取操作方法，因此提出了这种非局部的特征提取，在某个位置进行计算时，考虑了所有位置特征的加权，其在自然语言处理、图片识别和视频分类任务上都有很好的效果。Zhang 等[24]于 2018 年将自注意力机制应用到生成对抗网络(GAN)中，实验证明，在网络中层或者低层添加自注意力层能取得较好的效果。现有的大多自注意力机制是辅助卷积层，局部自注意力相比全局自注意力极大地减少了计算量。本文在缩短网络训练时间的同时选用了较小的卷积核，以应用自注意力弥补小卷积核获取信息不足的缺陷。

自注意力机制的计算过程如图 1 所示。

具体为：1)对输入的图像特征分别进行卷积操作，其中 $f(x)$，$g(x)$和 $h(x)$是 3 个函数，代表 1×1 的卷积核，但每个函数对应的输出通道数不同，分别得到 ***query***，***key*** 和 ***value*** 矩阵；2)将 ***query*** 与 ***key*** 进行点积计算得到权重矩阵，即得到特征图中每个像素与其他所有位置像素相关性的矩阵；3)利用 softmax 函数对权重进行归一化处理，以得到权重系数；4)将权重系数和 ***value*** 相乘，将通道注意力应用到所有通道的每张特征图的对应位置上，最终得到注意力特征 Z，其中 H，W，C 分别代表输入图片的高度、宽度以及输入的通道数。

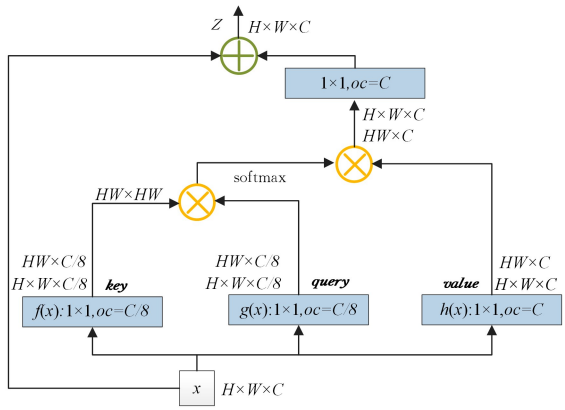


图 1 自注意力机制

Fig. 1 Self-attention

输入数据的多源性以及数据类型的丰富性，使得特定网络结构的泛化能力受到限制，如文献[12-13]中，在不同数据集上采用的网络深度、每层神经元的个数、卷积核的大小等均不相同。为了进一步降低数据类型对网络结构的依赖性，本文拟探索网络结构的表征能力并将其应用于自监督图像聚类任务中。

3 基于自注意力机制的自监督深度聚类算法

本文采用带自注意力层的卷积自编码器来抽取特征，即将自注意力层固定添加在第一个卷积层后，使用统一的网络结构进行训练特征表达后再进行聚类分析。

3.1 SADC 网络结构

为了提升所提网络特征提取的性能，使用带自注意力层的卷积自编码器(Self-attention Convolutional Autoencoder，SACAE)网络进行预训练。编码器和解码器各有 4 个卷积层，编码器卷积层后添加了 flatten 操作来拉平特征向量，并使用两个全连接层来获得低维特征表示，解码器与编码器部分的卷积层和全连接层完全对称，保证了输入输出维度相同。将带高斯噪声的数据和原始数据都作为自编码器网络的输入，将重构图像作为输出，计算网络的重构损失，迭代优化自编码器的网络参数。

首先，前两个卷积层的卷积核大小为 3×3，卷积核的个数分别为 30 和 50，步长为 2；第三个卷积层选用 3×3 的卷积核，卷积核个数为 50，步长为 1；第四个卷积层的卷积核大小为 2×2，卷积核个数为 50，步长为 1。通过选用小的卷积核和变步长来增强网络特征提取的能力。

其次，基于自注意力机制的自监督深度聚类算法由两部分组成，如图 2 所示。第一部分为添加了自注意力层的卷积自编码器(SACAE)，为了捕获更多对聚类有用的特征信息，将自注意力层添加在第一个卷积层之后。第二部分由 SACAE 的编码器部分加上聚类层(*K*-means 聚类算法)组成，可由 3.2 节中的式(2)计算出聚类软分配，再通过辅助分布式(3)得到判别增强的聚类软分配。迭代优化 KL 散度损失来调整网络参数和聚类中心，同时对数据进行聚类。设网络

输入数据为 $X=\{x_1,x_2,\cdots,x_N\}\in R^{N\times D}$，定义非线性映射函数 $f_{\theta_1}:x_i\rightarrow z_i$，$g_{\theta_2}:z_i\rightarrow\tilde{x}_i$，$z_i$ 是 x_i 在特征空间的低维特征且 $Z=\{z_1,z_2,\cdots,z_N\}$，$z_i\in R^d$，d 是特征空间的维度；$\tilde{x}_i$ 是样本 x_i 的重构样本，f_{θ_1} 和 g_{θ_2} 分别代表深度卷积自编码器从原始数据到低维特征空间的映射和从特征空间到重构数据的映射，$\theta=\{\theta_1,\theta_2\}$ 代表网络参数。第一部分网络在预训练过程中优化整个网络的参数 θ，第二部分网络在联合训练过程中优化编码器的网络参数 θ_1、聚类中心点和类簇指派。聚类中心表示为 $U=\{u_1,u_2,\cdots,u_K\}$，其中 K 为聚类中心的个数。聚类后输出的是一个 k 维向量，即软聚类指派，每个元素的范围是[0,1]，通过将其转换成硬指派类标来确定聚类类别。

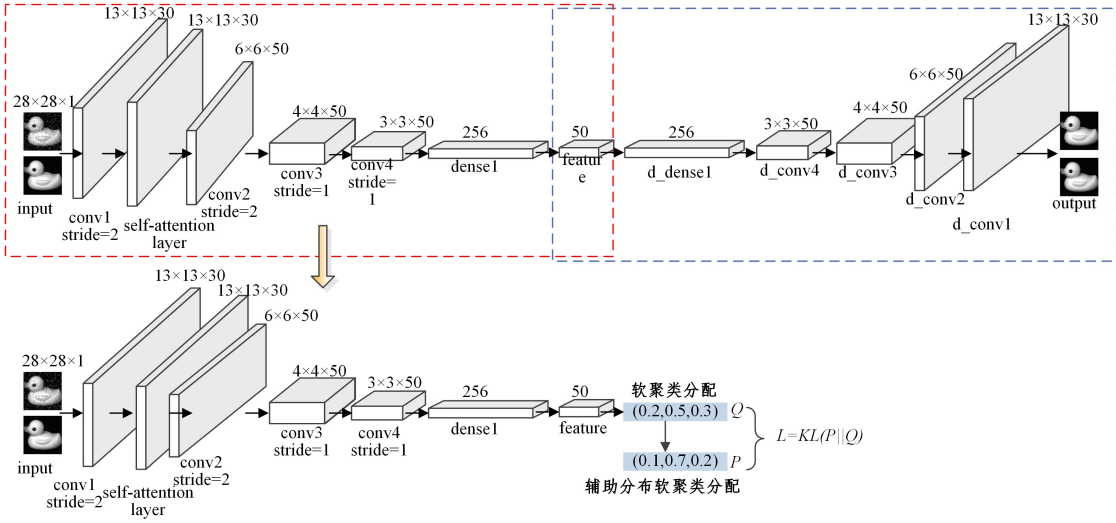


图 2　SADC 网络的框架

Fig. 2　Framework of SADC network

3.2　SADC 算法

第一部分中，损失函数采用重构误差：

$$L_{\mathrm{SACAE}}=\frac{1}{n}\sum_{i=1}^{n}\|x_i-\tilde{x}_i\|_2^2+\lambda\sum\|\theta\|_2^2 \tag{1}$$

其中，第一项是网络的重构误差，第二项是 L2 正则化项，防止网络过拟合，平衡参数 $\lambda>0$。用随机梯度下降法[25-26]进行优化，完成自编码器的预训练，得到低维特征表示 z_i。

第二部分在带网络初始参数的编码器后接经典的 K-means 算法构成 SADC 深度聚类方法，对第一部分用 K-means 对低维特征表示 z_i 进行聚类，得到初始聚类中心和聚类软分配，通过迭代损失函数来进一步优化网络参数和聚类划分结果。

首先采用基于 t 分布的软指派来计算低维特征表示 z_i 与聚类中心点 u_j 的相似性 s_{ij}。计算公式为 $s_{ij}=\left(1+\frac{w_j\times d_{ij}}{\alpha}\right)^{-\frac{\alpha+1}{2}}$，其中，$d_{ij}=\|z_i-u_j\|_2^2$ 是特征表示 z_i 与聚类中心 u_j 的欧氏距离，$\alpha>0$ 是 t 分布的自由度；另外超参数 $w_j=\begin{cases}1, & d_{ij}=\min\{d_{i1},d_{i2},\cdots,d_{ik}\}\\ 2, & \text{others}\end{cases}$ 增加了相距最近的特征 z_i 与聚类中心点 u_j 的相似性。由 s_{ij} 归一化后得到的概率值作为聚类的软指派，如式(2)所示：

$$\boldsymbol{q}_{ij}=\frac{s_{ij}}{\sum_{k=1}^{K}s_{ik}} \tag{2}$$

其中，$\boldsymbol{q}_{ij}$ 满足约束条件 $\sum_{j=1}^{K}\boldsymbol{q}_{ij}=1$。

其次，由软分布 $\boldsymbol{q}_{ij}$ 构造辅助分布 p_{ij}，使得辅助分布 p_{ij} 具有更高的可信度，如式(3)所示：

$$p_{ij}=\frac{q_{ij}^{\beta}}{\sum_{k=1}^{K}q_{ik}^{\beta}},\beta>1 \tag{3}$$

其中，β 为指数幂。

由式(2)和式(3)得到两个概率分布 q_i 和 p_i。当 q_i 中某个类簇的概率值 q_{ic} 大于 $1/K$ 时，对应的 p_{ic} 的概率值一定大于 q_{ic}，同时 p_i 其余位置上对应的概率值均小于 q_i。这说明辅助分布 p_i 提高了属于某一类簇的概率，同时降低了属于其他类簇的概率，提高了聚类划分的可信度。

定义 SADC 算法的损失函数 L_{SADC} 为概率分布 q_{ij} 和辅助目标分布 p_{ij} 之间的距离，同时加入正则化项来防止模型过拟合。

$$L_{\mathrm{SADC}}=\sum_{i=1}^{n}\sum_{j=1}^{K}p_{ij}\ln\frac{p_{ij}}{\boldsymbol{q}_{ij}}+\frac{\lambda}{2}\sum\|\theta_1\|_2^2 \tag{4}$$

优化函数在网络传播的过程中更新聚类中心 u_j 和网络参数，损失函数 L_{SADC} 关于 z_i 和 u_j 的梯度计算如式(5)和式(6)所示：

$$\frac{\partial L_{\mathrm{SADC}}}{\partial z_i}=\frac{\alpha+1}{\alpha}\sum_j\left(1+\frac{\|z_i-u_j\|^2}{\alpha}\right)^{-1}(p_{ij}-\boldsymbol{q}_{ij})(z_i-u_j) \tag{5}$$

$$\frac{\partial L_{\mathrm{SADC}}}{\partial u_j}=-\frac{\alpha+1}{\alpha}\sum_i\left(1+\frac{\|z_i-u_j\|^2}{\alpha}\right)^{-1}(p_{ij}-\boldsymbol{q}_{ij})(z_i-u_j) \tag{6}$$

根据式(5)所示的反向传播[27]来更新网络权重，用式(6)来更新聚类中心点，其中 α 设定为 1。

基于自注意力的深度聚类(SADC)算法的详细步骤如算法 1 所示。

算法 1 SADC 算法

输入：原始图像数据、添加高斯噪声的数据、聚类个数 K

输出：聚类指派即类标向量 $\mathbf{q}_{ij}$

1. 预处理：
   对数据集中的原始数据进行归一化处理，加入一定比例的高斯噪声，初始化网络参数。
2. 使用端到端的方法训练自编码器，使用随机梯度下降算法优化网络参数。
3. 保存并输出编码器部分的参数值及低维特征表示 $Z=\{z_1, z_2, \cdots, z_n\}$。
4. 利用低维特征表示和聚类中心，由式(2)计算输出概率分布。
5. 由式(3)计算辅助概率分布。
6. 由式(5)和式(6)迭代更新网络参数和聚类中心点，直至收敛。
7. 计算数据的聚类软指派，输出聚类硬指派，并计算聚类精度。

3.3 网络复杂度分析

3.3.1 卷积层

卷积网络中，单卷积层的时间复杂度为 $O(M^2K^2C_{in}C_{out})$，M 是该层卷积后输出特征图的边长，K 是卷积核的边长，C_{in} 是该卷积层的输入通道数，C_{out} 是本卷积层的输出通道数。因此，所有卷积层的时间复杂度为 $O_{Time}(\sum_{l=1}^{D} M_l^2 K_l^2 C_{l-1} C_l)$，其中 D 是卷积层的个数；SACAE 网络卷积层的时间复杂度为 $O_{Time}(\sum_{l=1}^{8} M_l^2 K_l^2 C_{l-1} C_l)$，反向传播阶段的时间复杂度为 $O_{Time}(\sum_{l=1}^{8} C_l C_{l-1} K^2 X^2)$。

3.3.2 全连接层

前向传播单层全连接层的时间复杂度为 $O_{Time}(n_{in} n_{out})$，n_{in} 是输入神经元个数，n_{out} 是输出神经元个数。误差反向传播与前向传播的时间复杂度相同，故全连接层的时间复杂度为 $O_{Time}(\sum_{i=1}^{D} n_{in} n_{out})$，SADC 网络结构的 SACAE 部分共有 3 个全连接层，故全连接层的时间杂度为 $O_{Time}(2\sum_{i=1}^{3} n_{in} n_{out})$。

3.3.3 空间复杂度

空间复杂度分两部分：1)总参数量，即网络所有带参数的权重参数总量；2)各层输出的特征图，即网络每层计算输出特征的大小。卷积层空间复杂度为 $O_{Space}(\sum_{l=1}^{D} K_l^2 C_{l-1} C_l + \sum_{l=1}^{D} M^2 C_l)$，全连接层空间复杂度为 $O_{space}(\sum_{i=1}^{D} n_{in} n_{out})$，SACAE 空间复杂度为 $O_{Space}(\sum_{l=1}^{8} K_l^2 C_{l-1} C_l + \sum_{l=1}^{8} M^2 C_l) + O_{space}(\sum_{i=1}^{3} n_{in} n_{out})$。

4 实验与分析

本文实验使用的硬件平台为 Intel © CoreTMi7-4720HQ 处理器；编程环境为 Anaconda4. 5. 12，使用开源的深度学习框架 tensorflow2. 0 和 Keras 库搭建 SADC 网络。

4.1 数据集描述

本文使用以下 6 个数据集来验证所提聚类算法的性能。

(1)MNIST 数据集是由 Lecun 等[28]提出的一个经典的手写数字数据集，共有 70000 张图片，均为 28×28 像素的灰度图像，数字范围为 0－9。

(2)BioID-FaceDatabase[1] 人脸数据集是由 27 个不同的人脸构成的 162 张 384×286 像素的灰度图像，且每张图片的拍摄角度和表情各不相同。

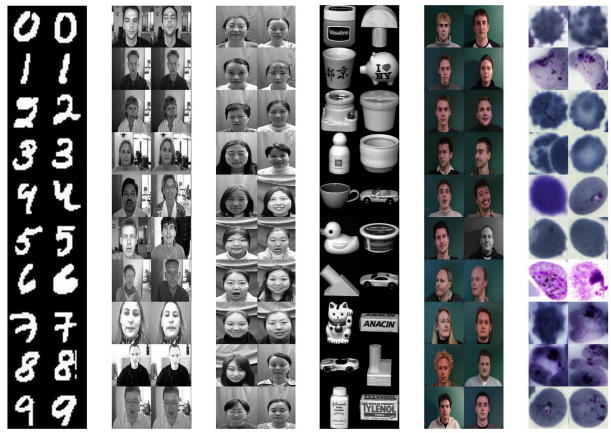
(3)CAS-PEAL-R1[2] 人脸数据集是由 40 个不同的人脸构成的表情各不相同的 200 张 360×480 像素的灰度图像，并且眼睛有闭有开，增加了图像识别的难度。

(4)COIL-20[3] 物品数据集由 20 类不同的物品构成，每个物品在水平轴上旋转 360°，每隔 5°拍摄一张照片，因此每个物品共有 72 幅图，构成 1440 张大小为 128×128 的灰度图片。

(5)IMM[4] 数据集是由 40 个不同的人脸构成的 240 张彩色图片，每张图片的大小为 640×480，人脸的拍摄角度和表情各不相同。

(6)MALARIA[5] 疟疾细胞数据集与前几个分布均匀的数据集不同，数据集共有 7352 张彩色图像，其中感染的细胞图片有 2149 张，未被感染的细胞图片有 5203 张，且每张图片的大小各不相同。

为了使用统一的网络结构对所有数据进行训练，图片大小统一被预处理为 28×28，并且图片被转换为灰度图像。对数量较少的 BioID-Face Database，CAS-PEAL-R1 和 IMM 数据集做简单加倍处理以扩大数据集，从而改善实验效果。图 3 给出了 6 个数据集的部分图片。



(a)MNIST (b)BioID (c)CAS-PEAL-R1 (d)COIL-20 (e)IMM (f)MALARIA

图 3 6 个数据集的部分图片展示

Fig. 3 Several examples from six datasets

4.2 评价指标

本文衡量聚类性能的指标有聚类精度(Accuracy，ACC)和归一化互信息(Normalized Multual Information，NMI)。这两种指标的范围都是[0，1]，取值越大表示聚类效果越好。已知真实聚类划分 $C=\{C_1, C_2, \cdots, C_T\}$，经过聚类算法后得到

1) https://www.bioid.com/About/BioID-Face-Data-base

2) http://www.jdl.ac.cn/peal/index.html

3) http://www.cs.columbia.edu/CAVE/software/soft-lib/coil-20

4) http://www.imm.dtu.dk/~aam/aamexplorer/

5) https://lhncbc.nlm.nih.gov/publication/pub9932

的聚类划分 $\Omega=\{\Omega_1,\Omega_2,\cdots,\Omega_R\}$。

4.2.1　聚类精度

聚类精度就是聚类正确的样本数量 n 与总样本数量 N 的比值：

$$ACC=\frac{n}{N} \tag{7}$$

设 a 表示在 C 和 Ω 中均为相同簇的样本对数量；b 表示在 C 中为相同簇，但在 Ω 中为不同簇的样本对数量；c 表示在 C 中为不同簇，但在 Ω 中为相同簇的样本对数量；d 表示在 C 和 Ω 中均为不同簇的样本对数量。

查全率(Recall)为在 C 中为相同簇且在 Ω 中被预测为相同簇样本对所占的百分比，查准率(Precision)为在 Ω 中为相同簇且在 C 中也为相同簇的样本对所占的百分比。

$$Recall=\frac{a}{a+b} \tag{8}$$

$$Precision=\frac{a}{a+c} \tag{9}$$

4.2.2　归一化互信息

用归一化互信息来衡量聚类的结果，$I(\Omega,C)$ 为互信息，$I(\Omega,C)=H(\Omega)-H(\Omega|C)$，表示已知类簇划分 C 的前提下类簇划分 Ω 的信息增益，Ω 和 C 的分布越相似，互信息就越大。采用 $(H(\Omega)+H(C))/2$ 做分母，保证 $NMI\in[0,1]$。

$$NMI(\Omega,C)=\frac{I(\Omega;C)}{(H(\Omega)+H(C))/2} \tag{10}$$

4.3　实验设置

本文的聚类器(SADC)由 4 个卷积层、2 个全连接层和 K-means 聚类算法构成。前两个卷积层的步长设置为 2，后两个卷积层的步长为 1；每层采用 ReLU[29] 激活函数。每个卷积层后都添加批归一化层(BN 层)，以缓解梯度消失。其次，自注意卷积自编码器网络用随机梯度下降(SGD)优化器，联合训练部分选用自适应距估计(Adam)优化器[30]，学习率分别为 2×10^{-4} 和 1×10^{-3}。Adam 优化器的收敛速度越快，带来的聚类性能就越佳。SACAE 部分迭代优化了 300 次，SADC 联合训练部分迭代优化了 200 次。

首先，通过观察模型收敛速度以及聚类精度，来确定公式中的超参数和优化器的学习率，通过增加全连接层来观察聚类精度的变化，证明网络结构的合理性。其次，验证自注意力机制对模型聚类的影响并展示对比的聚类结果。然后根据 ACC 和 NMI 评价指标将本文的 SADC 算法与 K-means，SACAE+K-means，DEC[11]，DDC[13] 算法进行对比。为了降低 K-means 的聚类结果对随机初始化的依赖性，最终选择 30 次重复实验中的平均值。最后，分析实验中出现的退化解问题。

4.4　实验结果

4.4.1　超参数选择

首先，在 4 个数据集上测试参数 w_j 对聚类性能的影响。

$$w_j=\begin{cases}1, & d_{ij}=\min\{d_{i1},d_{i2},\cdots,d_{ik}\}\\ 2, & \text{others}\end{cases}$$

由表 1 知，参数 w 对模型聚类精度的影响较小，在 MNIST 上精度下降了 0.2%，在 BioID 上精度提高了 0.6%。但参数 w 对网络的收敛速度有一定影响，以 BioID 数据集为例，引入 w 后网络更快收敛，达到最高的精度值，如图 4 所示。

表 1　参数 w 对模型聚类精度的影响

Table 1　Influence of parameter w on clustering accuracy of model

	MNIST	BioID	CAS-PEAL-R1	COIL-20
未引入 w	**0.921**	0.919	0.915	0.729
引入 w	0.919	**0.925**	**0.915**	**0.729**

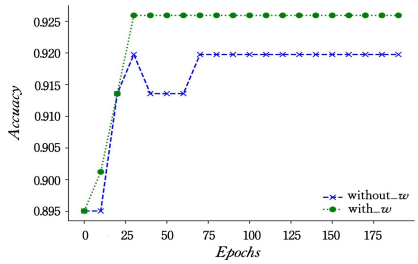


图 4　BioID 数据集上参数 w 对网络聚类性能的影响

Fig. 4　Influence of parameter w on the performance of network clustering on BioID database

其次，计算辅助分布 p_{ij} 时的超参数，式(3)中的分子是指数族函数，超参数 β 对算法的聚类精度及网络收敛速度均有一定的影响。如表 2 所列，记录不同的 β 值对应模型达到精度最高时迭代的次数。β 越大，网络的收敛速度越快，但是聚类精度有所下降。为了保证聚类精度，后续实验中均选取 $\beta=2$。

表 2　参数 β 对聚类精度及网络收敛速度的影响

Table 2　Influence of parameter β on clustering accuracy and network convergence speed

β	MNIST		BioID		CAS-PEAL-R1		COIL20	
	epoches	*ACC*	*epoches*	*ACC*	*epoches*	*ACC*	*epoches*	*ACC*
2	200	**0.921**	200	0.925	91	**0.915**	6	**0.736**
3	105	0.895	158	0.919	60	0.915	1	0.729
4	83	0.891	73	**0.932**	44	0.905	1	0.729
5	62	0.892	71	0.925	35	0.905	1	0.729

然后，在构建网络时用 BioID 数据集对网络的学习率进行测试。由表 3 可以看出，学习率为 2×10^{-4} 网络的重构损失最小，因此选取此学习率进行后续的网络训练。在 Adam 中，学习率为 1×10^{-3} 时聚类效果最好，与通常实验中默认的学习率相同，结果如表 4 所列。该组实验说明，学习率对模型的聚类精度有一定的影响。

表 3　SGD 不同学习率时的重构损失

Table 3　Reconstruction loss of SGD with different learning rates

学习率	2×10^{-3}	$\mathbf{2\times10^{-4}}$	2×10^{-5}	2×10^{-6}	2×10^{-7}
L_{SACAE}	0.122	**0.116**	0.136	0.211	0.847

表 4　Adam 不同学习率时的聚类精度

Table 4　Clustering accuracy of Adam with different learning rates

学习率	$\mathbf{1\times10^{-3}}$	1×10^{-4}	1×10^{-5}	1×10^{-7}
ACC	**0.925**	0.919	0.919	0.895
L_{SADC}	**0.221**	0.252	0.391	0.452

最后，在 BioID，CAS-PEAL-R1 和 IMM 数据集上测试网络结构的聚类性能。SADC+k 表示在原网络基础上添加的全连接层数，聚类精度的柱状对比图如图 5 所示，可以看出，只增加网络深度，聚类精度没有得到提高。全连接将局部特征进行组合，能降低特征位置对分类任务的影响，但没能给特征位置敏感的聚类任务带来好的效果。

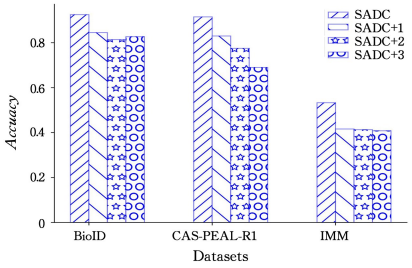


图 5 SADC 中增加全连接层时的聚类精度

Fig. 5 Clustering accuracy of SADC with full connection layer

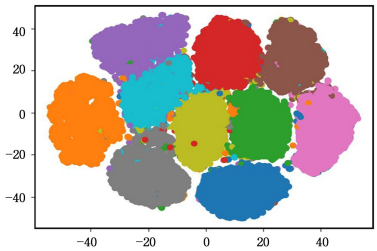
4.4.2 自注意力机制对聚类性能的影响

为使网络有更强的特征提取能力，更加关注有用信息，忽略无关信息，SADC 算法引入了自注意力机制。聚类性能如表 5 所列。自注意力机制大大提高了 SADC 算法的聚类精度，在 MNIST 和 CAS-PEAL-R1 数据集上有小幅度的提升，分别为 2.9%和 3.0%；在 BioID 和 MALARIA 数据集上提升了 10%左右。t-SNE[31] 主要用于高维数据的可视化，当数据嵌入 2 维或 3 维时，效果最好。用 t-SNE 对 MNIST 上的聚类结果进行可视化，结果如图 6 所示。可以看出，添加自注意力层后算法的聚类效果更好，类簇间的区分更加明显。

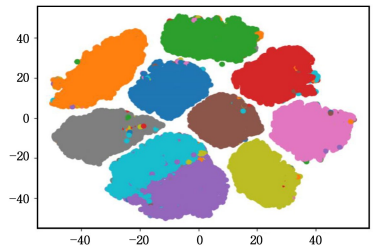
表 5 自注意力层对 SADC 模型性能的影响

Table 5 Effect of self-attention layer on SADC model performance

	MNIST		BioID		CAS-PEAL-R1		MALARIA	
Self-attention	with	without	with	without	with	without	with	without
ACC	**0.921**	0.892	**0.925**	0.814	**0.915**	0.885	**0.712**	0.615
NMI	**0.903**	0.893	**0.944**	0.929	0.929	**0.949**	**0.776**	0.697



(a)未添加自注意力层



(b)添加自注意力层

图 6 MNIST 上 SADC 的聚类结果 t-SNE 可视化

Fig. 6 t-SNE visualization of clustering results of SADC on MNIST dataset

4.4.3 其他聚类算法的对比

SADC 聚类算法与其他深度聚类算法在 6 个数据集上进行了对比，聚类精度如表 6 所列。SACAE+K-means 先用自注意卷积自编码器提取特征，再用 K-means 聚类；DEC 算法使用原文中堆叠自编码器进行预训练，其中自编码器为全连接网络；由于 DDC 算法的网络结构不统一，除 MNIST 和 COIL-20 数据集采用 DDC 中的聚类精度外，其他数据集均采用 DDC 中 MNIST 上的网络结构进行聚类。

表 6 各类算法的聚类精度比较

Table 6 Comparison of clustering accuracy of several clustering algorithms

Methods	Datasets					
	MNIST	BioID	CAS-PEAL-R1	COIL-20	IMM	MALARIA
K-means	0.392	0.846	0.880	0.669	0.533	0.519
SACAE+K-means	0.818	0.599	0.480	0.536	0.283	0.460
DEC[11]	0.843*	0.853	0.903	0.597	0.429	0.701
DDC[13]	0.892*	0.864	0.755	**0.803***	0.495	0.692
SADC	**0.921**	**0.925**	**0.915**	0.736	**0.545**	**0.712**

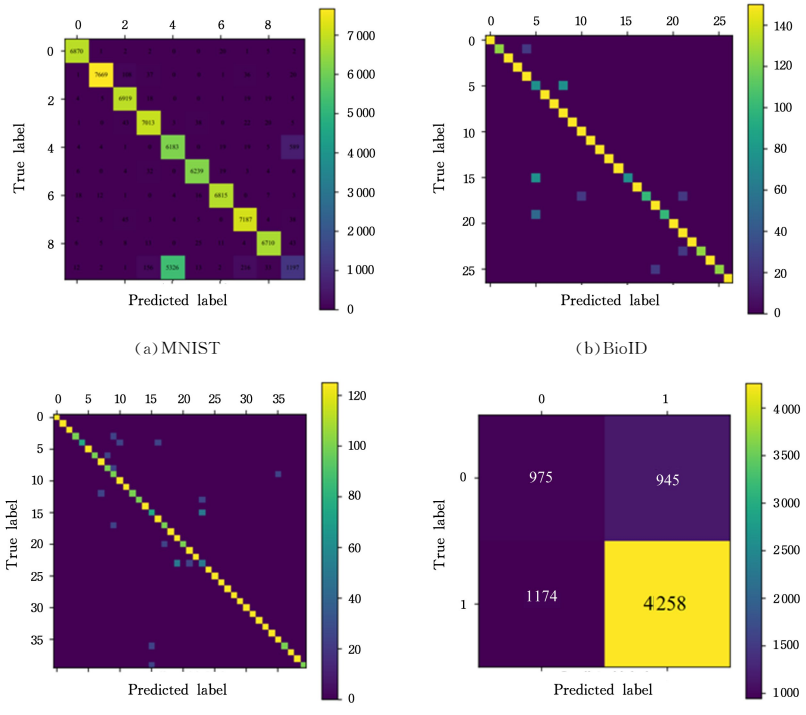
注：* 表示文献[11,13]中给出的实验精度

划分出不容易被区分的图像类，然后更有针对性地设计特征提取策略。本文实验首先根据 SADC 算法在各个类簇上的聚类表现，计算模型对各个类簇的准确率(Precision)和召回率(Recall)，获得混淆矩阵。图 7 给出了 SADC 聚类算法在部分数据集上的混淆矩阵。

由表 6 可知，除了 COIL-20 上的 DDC 算法外，本文提出的 SADC 算法的准确率均优于对比算法，每个数据集上的精度均有提高。与 DEC 相比，6 个数据集分别提高了 7.8%，7.2%，1.2%，13.9%，11.6%和 1.1%，说明本文的 SADC 网络提取特征的能力较好。由图 7(a)所示的结果可明显看到，数字 4 容易被错分成数字 9；从 4.1 节中的图 3 可以看出，这两个数字较为相似。如图 7(b)所示，在 BioID 上，第 5，8，15 类分错的数量相对较多，在数据集中这 3 个人的五官和脸部轮廓非常相似，导致识别效果欠佳。CAS-PEAL-R1 有 40 类，分类难度较大，但是观察混淆矩阵可以发现，每一类分错的数量并不多。由于数据集 MALARIA 是两类且类分布不均衡，因此对角线上聚类正确的样本个数相差较大，其中 0 代表感染的细胞，1 代表未感染的细胞。第 0 类即感染的细胞中，分类正确的个数为 975，错分到第 1 类的个数为 945，该类的聚类正确率为 0.508，稍高于 0.5；第 1 类即未感染的细胞中，分类正确的个数为 4258，错分到第 0 类的个数为 1174，该类的聚类正确率是 0.784，远远高于第 0 类的聚类正确率。说明被感染的细胞的质变特征没被表示出来，与正常细胞的区分度不高。

我们还对比了深度聚类算法 DEC，DDC 和 SADC 收敛时的迭代次数，结果如表 7 所列。由表 7 可知，除 COIL-20 数据集外，SADC 聚类算法的收敛速度均优于 DEC 和 DDC，尤其是 BioID 和 CAS-PEAL-R1 上的收敛迭代次数大大缩减。对于时间复杂度，依据 DEC 中给出的网络结构参数计算其复杂

度，而 DDC 则选择文中给出的 4 个网络中时间复杂度最小的进行计算，它们的编码器部分时间复杂度为 $O(10^7)$，而 SADC 算法的时间复杂度为 $O(10^6)$。显然，本文算法的时间复杂度较低，与其他算法相比差了一个数量级。



(a) MNIST　(b) BioID

(c) CAS-PEAL-R1　(d) MALARIA

图 7　SADC 算法在 4 个数据集上的混淆矩阵

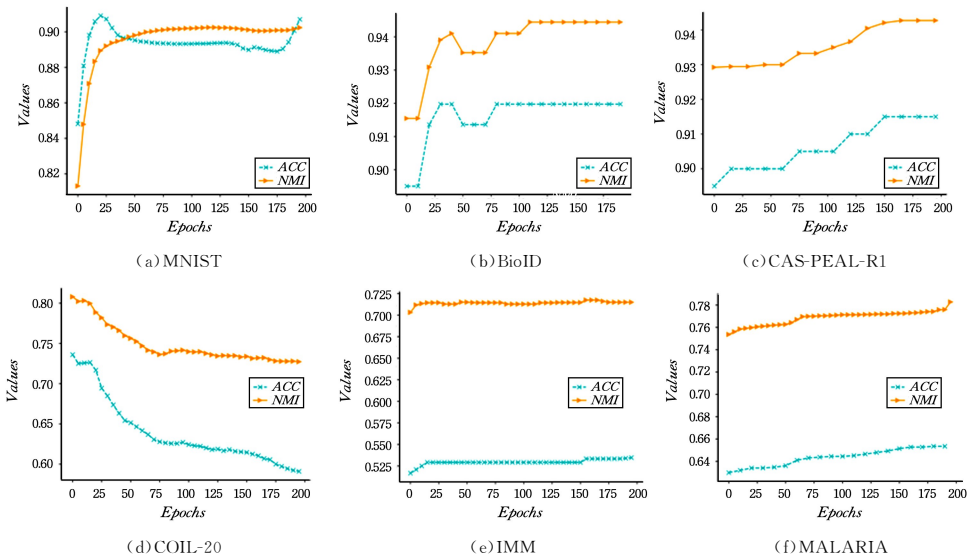
Fig. 7　Confusion matrix of SADC algorithm on four datasets

表 7　对比 3 个深度聚类算法收敛的迭代次数

Table7　Comparison of number of convergence-iterations of three deep clustering algorithms

Methods	Datasets					
	MNIST	BioID	CAS-PEAL-R1	COIL-20	IMM	MALARIA
DEC[11]	152	60	74	173	181	135
DDC[13]	146	42	65	**158**	162	115
SADC	**130**	**28**	**42**	200	**156**	**102**

另外，图 8 给出了 SADC 算法在 6 个数据集上的基准指标 *ACC* 和 *NMI* 曲线，其中横坐标轴为迭代次数，纵坐标轴为聚类指标值。由图 8 可知，在数据集 MNIST，BioID 和 CAS-PEAL-R1 上，随着迭代次数的增加，*ACC* 和 *NMI* 的上升趋势较为明显，最终趋于稳定。在数据集 IMM 和 MALARIA 上，随着迭代次数的增加，*ACC* 和 *NMI* 的提升幅度较小。而 COIL-20 数据集上的精度却随着迭代次数的增加而降低。



(a) MNIST　(b) BioID　(c) CAS-PEAL-R1

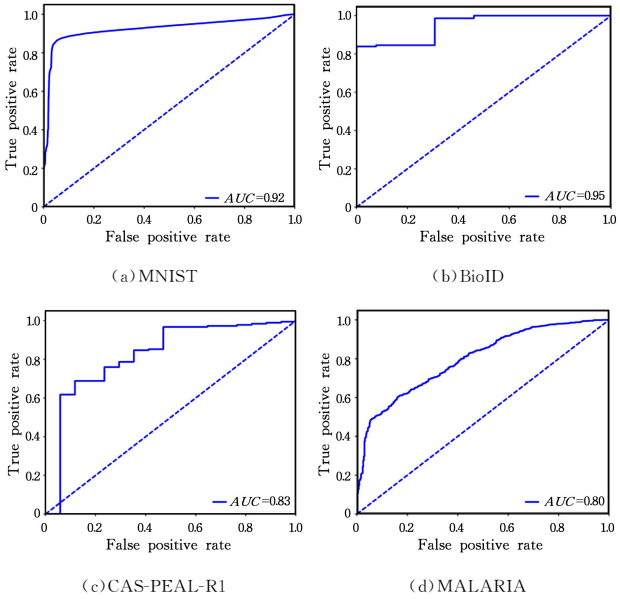
(d) COIL-20　(e) IMM　(f) MALARIA

图 8　在 6 个数据集上 SADC 的 *ACC* 和 *NMI* 曲线

Fig. 8　*ACC* and *NMI* curves of SADC on six datasets

实验中注意到只有第一轮的精度较高，从第二轮开始，网络把所有样本归为一类，即出现了严重的退化解现象。因此，数据增强手段一定程度上缓解了退化解的问题，但是随着迭代次数的增加，聚类精度依然缓慢下降。限于篇幅，解决该数据集上的退化解问题将是未来工作的重点之一。

用 ROC(Receiver Operating Characteristic)曲线和 AUC 面积(Area Under the ROC Curve)进一步衡量 SADC 聚类算法的性能，结果如图 9 所示。*ROC* 曲线越靠近左上方，*AUC* 的值越大(*ROC* 曲线下的面积)，查全率就越高，这说明 SADC 算法的聚类效果越好。首先对概率分布 q 进行硬指派，计算 K 类类簇真正分类正确的数据量和实际分到此类簇但是错分的数据量，从而得到 K 条 *ROC* 曲线。对 K 条 *ROC* 曲线取平均得到最终的 *ROC* 曲线。在 4 个数据集上 *AUC* 的面积分别为 0.92，0.95，0.83 和 0.80。可以看出，SADC 聚类算法在 MNIST 和 BioID 上的性能最好；在平衡类数据集上的聚类性能优于在不平衡类数据集上的聚类性能。由于 BioID 和 CAS-PEAL-R1 的样本个数较少，因此 *ROC* 曲线不够光滑，出现了锯齿状。



(a) MNIST (b) BioID

(c) CAS-PEAL-R1 (d) MALARIA

图 9 ROC 曲线

Fig. 9 Receiver operating characteristic curve

结束语 本文针对以往的经典深度聚类算法网络特征提取能力欠佳和网络结构泛化能力较差等问题，提出了基于自注意力的自监督深度聚类算法 SADC。实验结果证明，使用卷积层和全连接层相结合的卷积自编码器的特征表示能力优于堆叠自编码器；添加自注意力机制后网络能更好地捕获有用的特征信息，通过微调网络来提升聚类性能，增强了网络的泛化能力，进而统一了网络结构。

实验结果表明，本文提出的 SADC 聚类算法在经典的数据集上比以往的经典深度聚类算法有更好的聚类效果。然而，亟待解决的问题如 COIL-20 上出现了退化解，MALARIA 数据集中感染疟疾的细胞质变特征没被很好地刻画出来，均会在实际应用领域中造成严重的损失。这些都是需要进一步研究的重点工作。

参 考 文 献

[1] ASANO Y M, RUPPRECHT C, VEDALDI A. Self-labelling via simultaneous clustering and repre-sentation learning[C]// Proceedings of the International Conference on Learning Representations (ICLR). 2020: 1-22.

[2] WANG C, BAI X, DU J. Diffuse Interface Based Unsupervised Images Clustering Algorithm [J]. Computer Science, 2020, 47(5): 149-153.

[3] MCCULLOCH W S, PITTS W H. A logical calculus of the ideas immanent in nervous activity[J]. The Bulletin of Mathematical Biophysics, 1988, 5: 115-133.

[4] LECUN Y, BOTTOU L. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[5] ZHAO H, JIA J, KOLTN V. Exploring self-attention for image recognition[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 10076-10085.

[6] SHANG L, LU Z, LI H. Neural Responding Machine for Short-Text Conversation[C]// Proceedings of the Meeting of the Association for Computational Linguistics. 2015.

[7] VASWANI A, SHAZEER N, PARMAR N, et al. Attention Is All You Need[C]// Proceedings of the Advances in Neural Information Processing Systems(NeuIPS). 2017: 5998-6008.

[8] XU K, BA J, KIROS R, et al. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention[C]// Proceedings of the International Conference on Machine Learning (ICML). 2015: 2048-2057.

[9] HUANG P, HUANG Y, WANG W, et al. Deep embedding network for clustering[C]// Proceedings of the IEEE International Conference on Pattern Recognition(ICPR). 2014: 1532-1537.

[10] YANG B, FU X. Towards k-means-friendly spaces: Simultaneous deep earning and clustering[C]// Proceedings of the International Conference on Machine Learning. 2017: 3861-3870.

[11] XIE J, GIRSHICK R, FARHADI A. Unsupervised Deep Embedding for Clustering Analysis[C]// Proceedings of the International Conference on Machine Learning(ICML). 2016: 478-487.

[12] LI F, QIAO H, ZHANG B. Discriminatively Boosted Image Clustering with Fully Convolutional Auto-Encoders[J]. Pattern Recognition, 2018, 83: 161-173.

[13] CHEN J, ZHANG M, ZHAO J. A Deep Clustering Algorithm Based on Denoising and Self-attention[J]. Computer Science and Techno-logy, 2020, 15(9): 1117-1727.

[14] XIE J, HOU Q, CAO J. Image Clustering Algorithms by Deep Convolutional Autoencoders[J]. Computer Science and Technology, 2019, 13(4): 586-595.

[15] LI B, PI D, CUI L, et al. DNC: A Deep Neural Network-based Clustering-oriented Network Embedding Algorithm[J]. Journal of Network and Computer Applications, 2020, 173: 102854.

[16] DING Y，WEI H，PAN Z，et al. Survey of Net-work Representation Learning[J]. Computer Science，2020，47(9)：52-59.

[17] HE K，SUN J. Convolutional neural networks at constrained time cost[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015：5353-5360.

[18] HU W，MIYATO T，TOKUI S，et al. Learning Discrete Representations via Information Maximizing Self-Augmented Training [C]// Proceedings of the 34th International Conference on Machine Learning (ICML). 2017：1-15.

[19] BALDI P. Autoencoders，Unsupervised learning and deep architectures[C]// Proceedings of the International Conference on Unsupervised and Transfer Learning Workshop. 2011：37-50.

[20] VICENT P，LAROCHELLE H，LAJOIE I，et al. Stacked Denoising Autoencoders：Learning Useful Representations in a Deep Network with a Local Denoising Criterion[J]. Journal of Machine Learning Research，2010，11(12)：3371-3408.

[21] DIZAJI K G，HERANDI A，HUANG H. Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization[C]// Proceedings of the IEEE International Conference on Computer Vision. 2017：5736-5745.

[22] LIN Z，FENG M，SANTOS C N D，et al. A Structured Self-attentive Sentence Embedding[C]// Proceedings of the International Conference on Learning Representations. 2017：1-15.

[23] WANG X，GIRSHICK R，GUPTA A，et al. Non-local Neural networks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018：7794-7803.

[24] ZHANG H，GOODFELLOW I，METAXAS D，et al. Self-Attention Generative Adversarial Networks[C]// International Conference on Machine Learning. PMLR，2019：7354-7363.

[25] BOTTOU L，CURTIS F E，NOCEDAL J. Optimization methods for large-scale machine learning[J]. Siam Review，2018，60(2)：223-311.

[26] RUDER S. An overview of gradient descent optimization algorithms[J]. arXiv：1609.04747，2016.

[27] RUMELHARTT D，HINTON G，WILLIAMS R. Learning representations by back-propagating errors [J]. Nature，1986，323(6088)：533-536.

[28] LECUN Y，BOTTOU L. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE，1998，86(11)：2278-2324.

[29] NAIR V，HINTON G. Rectified linear units i-mprove restricted boltzmann machines[C]// Proceedings of the 27th International Conference on Machine Learning. Madison：Omni press，2010：807-814.

[30] KINGMA D，BA L. Adam：A method for sto-chastic optimization[C]// Proceedings of the 3rd International Conference on Learning Representations. 2015：1-15.

[31] LAURENS V，HINTON G. Visualizing Data using t-SNE[J]. Journal of Machine Learning Research，2008，9 (2605)：2579-2605.

HAN Jie，born in 1996，postgraduate. Her main research interests include image clustering and machine learning.

CHEN Jun-fen，born in 1976，Ph.D，associate professor，master supervisor，is a member of China Computer Federation. Her main research interests include data mining，machine learning and image processing.

（责任编辑：柯颖）