智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

结合度量融合和地标表示的自编码谱聚类算法

张敏，周治平

引用本文:
张敏, 周治平. 结合度量融合和地标表示的自编码谱聚类算法[J]. 智能系统学报, 2020, 15(4): 687–696.
ZHANG Min, ZHOU Zhiping. An autoencoder-based spectral clustering algorithm combined with metric fusion and landmark representation[J]. *CAAI Transactions on Intelligent Systems*, 2020, 15(4): 687–696.

在线阅读 View online: https://dx.doi.org/10.11992/tis.201911039

DOI: 10.11992/tis.201911039

结合度量融合和地标表示的自编码谱聚类算法

张敏[1], 周治平[1,2]

(1. 江南大学 物联网工程学院, 江苏 无锡 214122; 2. 江南大学 物联网技术应用教育部工程研究中心, 江苏 无锡 214122)

摘 要: 针对大多数现有谱聚类算法处理大规模数据集时面临聚类精度低、大规模相似度矩阵存储开销大的问题, 提出一种结合度量融合和地标表示的自编码谱聚类算法。引入相对质量概念进行节点评估, 选取最具代表性的点作为地标点, 通过稀疏表示近似获得图相似度矩阵, 以降低存储开销。同时考虑到近邻样本的几何分布和拓扑分布的信息, 融合欧氏距离与 Kendall Tau 距离来度量地标点与其他样本之间的相似度, 提高聚类精度; 以栈式自编码器取代拉普拉斯矩阵特征分解, 将所获得的相似度矩阵作为自编码器的输入, 通过联合学习嵌入表示和聚类来进一步提高聚类精度。在 5 个大规模数据集上的实验验证了本文算法的有效性。

关键词: 大规模数据集; 度量融合; 地标表示; 相对质量; 稀疏表示; 栈式自编码器; 联合学习; 嵌入表示
中图分类号: TP18 **文献标志码:** A **文章编号:** 1673−4785(2020)04−0687−10

中文引用格式: 张敏, 周治平. 结合度量融合和地标表示的自编码谱聚类算法 [J]. 智能系统学报, 2020, 15(4): 687–696.
英文引用格式: ZHANG Min, ZHOU Zhiping. An autoencoder-based spectral clustering algorithm combined with metric fusion and landmark representation[J]. CAAI transactions on intelligent systems, 2020, 15(4): 687–696.

An autoencoder-based spectral clustering algorithm combined with metric fusion and landmark representation

ZHANG Min[1], ZHOU Zhiping[1,2]

(1. School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China; 2. Engineering Research Center of Internet of Things Technology Applications Ministry of Education, Jiangnan University, Wuxi 214122, China)

Abstract: Most existing spectral clustering algorithms are faced with low clustering accuracy and costly large-scale similarity matrix storage. Aiming at these problems, this paper proposes an autoencoder-based spectral clustering algorithm combined with metric fusion and landmark representation. First, instead of random sampling, the concept of relative mass is introduced to evaluate node quality. Based on this, the most representative nodes are selected as the landmark points and the graph similarity matrix is approximately obtained by sparse representation. Meanwhile, considering the geometric and topological distribution of the nearest neighbor samples,the Euclidean distance and Kendall Tau distance are integrated to measure the similarity between the landmarks and the other points, so as to increase the clustering precision. A stacked autoencoder is then used to replace the Laplace matrix eigen-decomposition, and the obtained similarity matrix is taken as the autoencoder's input. The clustering accuracy is further improved by joint learning of embedded representation and clustering. Experiments on five large-scale datasets validate the effectiveness of our algorithm.

Keywords: large-scale datasets; metric fusion; landmark representation; relative mass; sparse representation; stacked autoencoder; joint learning; embedded representation

聚类旨在根据数据点之间的相似性将其划分到不同的簇, 使簇内相似度最大, 簇间相似度最小[1]。传统聚类方法如 K-means 算法和模糊聚类算法, 缺乏处理复杂数据结构的能力, 当样本空间非凸时, 易陷入局部最优。近年来, 谱聚类算法因其可在任意形状空间内进行聚类, 并收敛到全局最优, 在非凸数据集表现出良好聚类性能, 在人脸识别、社区检测、图像分割等领域有着广

收稿日期: 2019−12−02.
通信作者: 张敏. E-mail: 15061882373_1@163.com.

泛的应用[2]。

谱聚类将聚类问题转化为图分割问题[3]，通过寻找最优子图对数据点进行划分。但现存的多数谱聚类算法在处理大规模数据集时，涉及相似度矩阵构造和对应的拉普拉斯矩阵分解，往往需要高昂的时空开销，高计算成本成为制约其在大规模场景应用中的瓶颈[4]。为实现对大规模数据的高效聚类分析，提升谱聚类算法的扩展性，大量研究方法被提出。Nyström 扩展作为一种近似低秩矩阵的有效方法，利用抽样技术，减少特征分解复杂度[5]。针对抽样策略的选择，Zhan 等[6]设计了一种自适应的 Nyström 采样方法，通过多次遍历并更新抽样概率，选取更有意义的样本点，以较小的抽样集获得理想的聚类效果。考虑到高斯核参数调整问题，Yang 等[7]提出一种基于层次二部图的谱聚类算法，通过金字塔结构式的多层锚点构造层次二部图降低计算复杂度，此外采用无参数邻接分配策略构造相似度矩阵，避免参数调优过程。蔡登等[8-9]提出基于地标表示的研究，择取具有代表性的数据点作为地标点，通过地标点的稀疏线性组合近似构造图相似度矩阵，从而有效降低谱嵌入的计算复杂度，该算法一定程度上解决了矩阵存储开销问题，但其随机采样确定地标点的方式会造成聚类结果的不稳定，故而地标点的选取为算法的关键。叶茂等[10]利用由近似奇异向量的长度计算的抽样概率来进行抽样，以保证聚类结果的稳定性和精度。Zhang 等[11]提出一种基于增量视角的抽样框架，每一个要抽样的点是由先前选定的地标点决定，在地标点之间建立显式关系。但这些改进的方法在择取地标点时都忽略了节点的拓扑属性，其可有效描述亲和图整体结构特性，对于捕捉高维数据空间的拓扑信息有着重要意义。故邓思雨等[12]将 PageRank 分值作为样本信息量的度量指标，Rafailid 等[13]提出一种基于地标选择的快速谱聚类算法，根据加权 PageRank 算法选择亲和图中最重要的节点作为地标点。Liu 等[14]将数据点视为 web 页面，数据点之间的距离类似为链路间的权重，采用 PageRank 算法评估数据点的重要性，选择代表点。该方法能够区分球状和非球状的簇，且减少噪声点的负面影响。以上这些改进的谱聚类算法虽然降低了计算复杂度，仍然需要对矩阵进行特征分解。Jia 等[15]设计了一种无需特征分解的大规模近似 Ncut 谱聚类算法，一方面通过对数据点的采样推断数据集的全局特征，减少归一化切割的空间需求，另一方面利用近似加权核 k-均值优化 Ncut 的目标函数，避免拉普拉斯矩阵的直接特征分解。Tian 等[16]证明了自编码器和谱聚类之间的相似性，利用栈式自编码器代替特征分解，有效降低计算复杂度。Banijamali 等[17]提出将深层结构与地标表示相结合，实现快速且准确的聚类。但上述提及的谱聚类算法均使用传统基于欧氏距离度量方案，并不能完全反映数据复杂的空间分布特性。光俊叶等[18]提出融合欧氏距离和 Kendall Tau 距离的谱聚类方法，充分利用不同相似性度量从不同角度获取数据信息的优势，全面反映底层结构信息。

本文在地标点采样[8-9]的理论基础上，采用基于网页质量的 PageRank 算法计算节点的重要性，作为抽样策略依据代替随机抽样，选取代表点作为地标点，通过稀疏表示近似获得图相似度矩阵，以降低存储开销。同时为全面反映数据底层结构信息，将欧氏距离与 Kendall Tau 距离两种度量方案融合，用局部标准差代替特定的缩放尺度，构造融合多度量方式的自适应相似度矩阵，以提高聚类精度。此外，用栈式自编码器代替特征分解，在联合学习框架中进行嵌入表示学习和聚类，避免微调环节覆盖之前所得最优参数，从而进一步提高聚类精度。

1　相关算法理论

1.1　基于 QPR 的地标选择

PageRank 算法采用平均分配思想，将节点权重划分给入链节点，忽略节点质量存在差异性，改进后的加权 PageRank 算法通过传入链路权重 $w_{(a,b)}^{\text{in}}$ 及传出链路权重 $w_{(a,b)}^{\text{out}}$ 2 个参数[13]，依据网络结构有区分地进行权值分配，但该参数在迭代过程中是固定存在的，存在局限性。基于网页质量的 PageRank 算法[19](page quality based PageRank algorithm, QPR) 定义了相对质量 $Q(a)$，用于迭代过程协助分配 PR 值，动态评估每个节点的质量，其计算公式为

$$Q(a)=\frac{\mathrm{PR}(a)}{\left(\sum\limits_{b\in B(a)}\mathrm{PR}(b)\right)\Big/\mathrm{MPR}(a)}$$
$$\mathrm{MPR}(a)=\max_{b\in B(a)}\mathrm{PR}(b) \tag{1}$$

式中：$B(a)$ 表示节点 a 的入链节点集合；$\mathrm{MPR}(a)$ 表示 a 的入链节点集合中的最大 PR 值。相对质量 $Q(a)$ 将迭代过程中 a 的入链集合与其本身的 PR 值信息结合，对于具有相同 PR 值的节点，拥有较大入链 PR 值的节点将得到加权，从而在下

一轮迭代中具有更高 PR 值。随着多轮迭代, 节点各自的 PR 值出现差距, 高质量的节点能较快积累 PR 值, 获得较高排名。该算法计算 PageRank 的公式为

$$\begin{aligned}&\mathrm{PR}(a)=d\sum_{b\in \mathrm{B}(a)}\mathrm{PR}(b)\Pr(b\to a)+\frac{(1-d)}{n}\\&\Pr(b\to a)=\mathrm{Q}(a)/\sum_{c\in \mathrm{F}(b)}\mathrm{Q}(c)\end{aligned}\tag{2}$$

式中: $\Pr(b\to a)$ 表示节点 b 分配给节点 a 的 PR 值比重, $\mathrm{F}(b)$ 表示节点 b 的出链节点集合, 即 $\mathrm{F}(b)=\{a|(b,a)\in E\}$, d 是一个阻尼因子, 通常设置为 0.85, n 为节点个数。当所有节点满足式 (3) 时, 迭代过程停止:

$$\frac{\mathrm{PR}(a)_{\mathrm{iter}-1}-\mathrm{PR}(a)_{\mathrm{iter}}}{\mathrm{PR}(a)_{\mathrm{iter}}}\leqslant\beta\tag{3}$$

该式表示上一次迭代与当前迭代之间的归一化差值, 其中, β 是预先设置的停止阈值。当满足该式时, 停止迭代, 选择 PR 值最高的 p 节点作为地标点。

1.2　自适应谱聚类算法

相似度矩阵的构造是谱聚类算法的一个重要步骤, 而高斯核函数是构造相似度矩阵最常用的度量方法之一。给定数据集 $\boldsymbol{X}=[x_1,x_2,\cdots,x_n]\in R^{n\times d}$, $i=1,2,\cdots,n$, 拟将其划分为 K 簇。谱聚类算法将聚类问题视为无向图 $\boldsymbol{G}=(\boldsymbol{V},\boldsymbol{E})$ 的多路划分问题, 其中顶点集 $\boldsymbol{V}=\{x_1,x_2,\cdots,x_n\}$ 为所有样本集合, $\boldsymbol{E}=\{A_{ij}\}$ 表示顶点间边的权重集合, $\boldsymbol{A}=(A_{ij})\in R^{n\times n}$, $i,j=1,2,\cdots,n$即为所需构造的相似度矩阵, 其元素被定义为

$$A_{ij}=\begin{cases}\exp\left(\dfrac{-\rho^2(x_i,x_j)}{2\sigma^2}\right), & i\neq j\\ 0, & i=j\end{cases}$$

式中: $\rho(x_i,x_j)$ 指样本 x_i 与 x_j 之间的一种距离度量方法, 参数 σ 是手动给出的固定缩放参数, 并不具有自适应性, 难以体现数据的真实分布[20]。自适应谱聚类算法[21]针对全局尺度参数不能有效反映数据集真实分布信息, 提出了局部尺度参数的概念, 根据样本 x_i 的第 k 个近邻定义样本 x_i 的局部尺度参数 $\sigma_i=\rho(x_i,x_k)$, 该算法定义相似度矩阵元素为

$$A_{ij}=\begin{cases}\exp\left(\dfrac{-\rho^2(x_i,x_j)}{2\sigma_i\sigma_j}\right), & i\neq j\\ 0, & i=j\end{cases}$$

特定的缩放参数 σ_i 可根据样本 x_i 和 x_j 的邻域点进行自调。该算法虽然在一定程度上克服了自适应谱聚类算法的缺点, 但也有一定的局限性。其聚类结果可能受到异常值影响, 因为局部尺度参数 σ_i 可能扭曲了离群值。为避免该问题, 文献 [22] 提出一种局部标准差谱聚类算法, 定义局部标准差 $\sigma_{\mathrm{std}\_i}=\sqrt{\dfrac{1}{k-1}\sum_{t=1}^{k}\rho^2(x_i,x_t)}$, 其表示样本 x_i 的前 k 个近邻的局部标准差, 相似度矩阵的元素则如式 (4) 所示, 尽可能地反映数据集的原始分布:

$$A_{ij}=\begin{cases}\exp\left(\dfrac{-\rho^2(x_i,x_j)}{2\sigma_{\mathrm{std}\_i}\sigma_{\mathrm{std}\_j}}\right), & i\neq j\\ 0, & i=j\end{cases}\tag{4}$$

2　本文算法

2.1　结合度量融合与地标表示的相似度矩阵

由 1.2 节可以看出谱聚类算法中所构造的相似度矩阵维度为 $n\times n$, 对于大规模数据集无法实现高效的谱嵌入, 空间计算复杂度较高。对此文献 [8-9] 提出选取 $p\ll n$ 个地标点作为特征代表点, 将所有样本近似表示为这些地标点的线性组合, 有效利用稀疏表示矩阵近似获得图相似度矩阵。本文采用 1.1 节的基于 QPR 的地标选择方式, 选择 PR 值最高的 p 个节点作为地标点。对于任意给定样本 x_i, 其近似样本 $\hat{x}_i$ 可表示为

$$\hat{x}_i=\sum_{j=1}^{p}Z_{ji}y_j$$

式中: y_j 是地标点矩阵 $\boldsymbol{Y}\in R^{d\times p}$ 的列向量, Z_{ij} 为稀疏表示矩阵 $\boldsymbol{Z}\in R^{p\times n}$ 的第 i 行、第 j 列元素, 表示所选取的 p 个地标点与所有样本点之间的成对相似性。根据稀疏编码策略, x_i 与 y_j 越接近, Z_{ji} 越大, 而若 y_j 不在样本 x_i 的 k 近邻内, 则将 Z_{ji} 置为 0, 通过引入 k 近邻点来度量点对之间的局部亲和力。相应稀疏表示矩阵 $\boldsymbol{Z}$ 的元素为

$$Z_{ji}=\begin{cases}\dfrac{A_{ji}}{\sum\limits_{m\in \mathrm{KNN}(x_i)}A_{mi}}, & j\in \mathrm{KNN}(x_i)\\ 0, & j\notin \mathrm{KNN}(x_i)\end{cases}\tag{5}$$

式中: A_{ji} 表示样本 x_i 与地标点 y_j 之间的相似度, 采用式 (4) 计算方式, 尽可能反映数据集的原始分布。在计算点对相似性时往往采用欧氏距离传统度量方式, 一些邻域有用信息容易被忽略。Kendall Tau 距离是衡量两个等级列表之间两两不一致的数量, 即求两个排列之间的逆序数, 反映了两个排列的相似程度。对于两排列 $L_1=(L_{11},L_{21},\cdots,L_{n1})$ 及 $L_2=(L_{12},L_{22},\cdots,L_{n2})$, 它们之间的 Kendall Tau 距离定义为

$$\mathrm{KT}(L_1,L_2)=\left|\{(i,j):i<j,((L_{i1}<L_{j1})\wedge(L_{i2}<L_{j2}))\mathrm{or}((L_{i1}>L_{j1})\wedge(L_{i2}>L_{j2}))\}\right| \tag{6}$$

式中：L_{i1} 和 L_{i2} 表示 L_1 和 L_2 中第 i 个元素各自的序号，|·| 为集合中元素的个数。若两排列相同，则 $\mathrm{KT}(L_1,L_2)$ 为 0；若两排列互为逆序，$\mathrm{KT}(L_1,L_2)$ 为 $n(n-1)/2$。通过除以 $n(n-1)/2$ 进行规范化处理，使 $\mathrm{KT}(L_1,L_2)$ 位于区间 [0,1] 内。

若将欧氏距离及 Kendall Tau 距离两种度量方式融合，生成增强的相似度矩阵，能在反映点对之间地理相似性的同时考虑点之间的拓扑逻辑相似性。将点 x_i 和 x_j 之间的欧氏距离表示为 $\rho_1(x_i,x_j)$，Kendall Tau 距离表示为 $\rho_2(x_i,x_j)$。在基于欧氏距离对样本进行排序，然后利用 Kendall Tau 距离度量两个样本之间的全局关系，全面反映数据的底层结构信息。对于样本 x_i，基于欧氏距离计算与 p 个地标点之间的相似度，可得一有序序列：$\mathrm{List}_i=[\rho_1(x_1,x_i),\rho_1(x_2,x_i),\cdots,\rho_1(x_m,x_i),\cdots,\rho_1(x_p,x_i)]$，同样的对于样本 x_j 可得有序序列 List_j，那么结合式 (6)，样本 x_i 和 x_j 之间的 Kendall Tau 距离则为

$$\rho_2(x_i,x_j)=\mathrm{KT}(\mathrm{List}_i,\mathrm{List}_j)$$

从度量融合的角度来看，大多数主流的思想都是通过线性组合多个亲和图来获得互补的相似信息[23]，然而这个线性组合对分配给每个度量的权重很敏感，且来自多个亲和图的互补信息并非线性相关的。为探索两种距离度量方案相互融合的思想，综合考虑相邻点的几何分布和拓扑分布，受到联合训练算法的启发，采用基于消息传递理论和交叉扩散过程的融合方法。以完全相似矩阵作为初始状态矩阵，即对式 (5) 不考虑 k 近邻点而构建相似矩阵，该初始状态矩阵 $\boldsymbol{F}$ 携带关于每个地标点与其他所有样本点之间相似性的完整信息，对应元素为

$$F_{ji}=\frac{A_{ji}}{\sum_{m=1}^{p}A_{mi}} \tag{7}$$

式 (7) 分别采用欧氏距离、Kendall Tau 距离度量方法计算获得的完全相似矩阵表示为 $\boldsymbol{F}^{(1)}$、$\boldsymbol{F}^{(2)}$。稀疏表示矩阵实际上是完全相似矩阵的 KNN 图，在扩散过程中为提高计算效率，采用稀疏表示矩阵作为核矩阵进行交叉扩散。将稀疏表示矩阵 $\boldsymbol{Z}$ 采用欧氏距离、Kendall Tau 距离分别表示为 $\boldsymbol{Z}^{(1)}$ 和 $\boldsymbol{Z}^{(2)}$。首先将 $\boldsymbol{F}_{h=0}^{(1)}=\boldsymbol{F}^{(1)}$ 和 $\boldsymbol{F}_{h=0}^{(2)}=\boldsymbol{F}^{(2)}$ 视为迭代过程第一步中的初始两个状态矩阵 ($h=0$)，而度量融合的关键步骤是迭代更新对应于每个度量的稀疏表示矩阵

$$\begin{aligned}\boldsymbol{F}_{h+1}^{(1)}&=\boldsymbol{Z}^{(1)}\times\boldsymbol{F}_h^{(2)}\times(\boldsymbol{Z}^{(1)})^{\mathrm{T}}\\ \boldsymbol{F}_{h+1}^{(2)}&=\boldsymbol{Z}^{(2)}\times\boldsymbol{F}_h^{(1)}\times(\boldsymbol{Z}^{(2)})^{\mathrm{T}}\end{aligned} \tag{8}$$

式中：$\boldsymbol{F}_h^{(1)}$ 表示基于欧氏距离度量的稀疏表示矩阵在 h 步迭代更新后的状态矩阵，$\boldsymbol{F}_h^{(2)}$ 表示基于 Kendall Tau 距离度量的矩阵在 h 步迭代更新后的状态矩阵。此步骤每次更新状态矩阵时，生成两个并行交换扩散过程，在 h 步后，融合两种度量方式的稀疏表示矩阵 $\boldsymbol{Z}$：

$$\boldsymbol{Z}=\frac{\boldsymbol{F}_h^{(1)}+\boldsymbol{F}_h^{(2)}}{2} \tag{9}$$

从概率角度分析该交叉扩散过程，对于状态矩阵 $\boldsymbol{F}_h^{(1)}$，依据扩散映射[24]可定义进行 h 步迭代时的扩散距离 $M_h^{(1)}(i,j)=\left\|\boldsymbol{F}_h^{(1)}(i,:)-\boldsymbol{F}_h^{(1)}(j,:)\right\|$，扩散过程即是将数据空间映射到扩散空间 $\mathfrak{R}_h^{(1)}$ 的过程，本质上每个样本都可由自身与其他数据点间相似度表示。为实现核矩阵之间的融合，对于样本 $x_h^{(1)}\in\mathfrak{R}_h^{(1)}$，引入线性算子 $\boldsymbol{Z}^{(2)}$，经过 $x_{h+1}^{(2)}=\boldsymbol{Z}^{(2)}x_h^{(1)}+\varepsilon$ (ε 为白噪声) 线性运算后，由扩散映射性质可获得 $x_{h+1}^{(2)}$ 的边缘分布[23]，同理可得 $x_{h+1}^{(1)}$。该交叉扩散过程的实质并非在原始数据空间中进行线性投影，而是在扩散空间中进行迭代线性运算的，这样对样本的噪声和规模具有较强鲁棒性，且融合了整个样本集的相似流形的内在结构。

在由式 (9) 获得稀疏表示矩阵 $\boldsymbol{Z}$ 后，通过该矩阵来近似获得数据 $\boldsymbol{X}$ 的规范化图亲和矩阵，即：$\hat{\boldsymbol{G}}=\hat{\boldsymbol{Z}}^{\mathrm{T}}\hat{\boldsymbol{Z}}\in R^{n\times n}$，其中 $\hat{\boldsymbol{Z}}=\boldsymbol{D}^{-1/2}\boldsymbol{Z}$，$\boldsymbol{D}$ 为度矩阵，其元素一般为 $D_{ii}=\sum_j Z_{ij}$，为进一步降低计算复杂度，采用文献 [2] 计算度矩阵元素 $D_{ii}=\mathrm{diag}(\hat{\boldsymbol{Z}}_i^{\mathrm{T}}\hat{\boldsymbol{Z}}^s)$，其中 $\hat{\boldsymbol{Z}}^s=\sum_{j=1}^{n}\hat{Z}_j$，为 $p\times1$ 的向量，其第 k 个元素为 $\hat{\boldsymbol{Z}}$ 的第 k 行元素之和，该计算 $\boldsymbol{D}$ 的时间复杂度为 $O(np)$。结合度量融合与稀疏表示的拉普拉斯矩阵表示为

$$\boldsymbol{L}=\boldsymbol{D}^{-\frac{1}{2}}\hat{\boldsymbol{G}}\boldsymbol{D}^{-\frac{1}{2}}=\boldsymbol{D}^{-\frac{1}{2}}\hat{\boldsymbol{Z}}^{\mathrm{T}}\hat{\boldsymbol{Z}}\boldsymbol{D}^{-\frac{1}{2}} \tag{10}$$

然后将 $\boldsymbol{D}^{-\frac{1}{2}}\hat{\boldsymbol{Z}}^{\mathrm{T}}$ 作为栈式自编码器的输入，通过编码和解码重构学习数据潜在特征表示。

2.2 深度嵌入谱聚类

假定聚类任务为 N 个样本，$\boldsymbol{X}=[x_1,x_2,\cdots,x_n]$，$x_i\in R^{d_x},i=1,2,\cdots,n$ 划分为 K 簇，每个都由聚类中心 $\mu_j,j=1,2,\cdots,K$ 表示。为避免直接在样本空间 $\boldsymbol{X}$ 上进行聚类，首先使用非线性映射进行数据转换，嵌入函数 $\varphi_W:\boldsymbol{X}\to\boldsymbol{H}$，将原始数据映射到潜在特征空间 $\boldsymbol{H}=[h_1,h_2,\cdots,h_n],h_i\in R^{d_h}$，与输入样本相比具有较低维度 ($d_h\ll d_x$)，再经过非线性映射进行数据重构，解码器映射函数 $g_{W'}:\boldsymbol{H}\to\hat{\boldsymbol{X}}$，$\hat{\boldsymbol{X}}=[\hat{x}_1,\hat{x}_2,\cdots,\hat{x}_n]$，$\hat{x}_i$ 表示样本 x_i 的数据重构。本文旨在寻找良好的嵌入特征表示 $\{h_i\}_{i=1}^n$ 使其更适合于

聚类, 所构造的损失目标函数包含自编码器重构损失及聚类损失, 利用自编码器以无监督方式学习表示, 所习得特征能最大限度地保留数据的固有局部结构, 聚类损失用于分散嵌入点。与需要分层预训练以及非联合嵌入聚类的学习模型不同, 同时避免微调步骤覆盖预训练获得的参数, 采用联合聚类和重构损失函数的目标函数, 对所有网络层进行端到端优化训练, 迭代优化自编码器参数及聚类中心, 损失目标函数 J 定义为

$$J = J_r + \gamma J_c \tag{11}$$

其中 J_r 和 J_c 分别表示为重构损失和聚类损失, $\gamma(0<\gamma<1)$ 用来调节潜在特征空间的失真程度。基于 2.1 节, 将构造的相似度矩阵作为栈式自编码的输入, 在进行聚类之前, 对嵌入特征表示 $\{h_i\}_{i=1}^n$ 进行预训练, 执行 k-means 算法, 获得初始自编码器参数 $\{W,W'\}$ 及聚类中心 $\{\mu_j\}_{j=1}^K$。

对于训练所获得的嵌入特征表示, 在文献 [23] 引入了由 t-SNE 衡量的软标签分布 p_{ik}, 用来衡量潜在空间样本 h_i 与聚类中心 μ_j 之间的相似度:

$$p_{ik} = \frac{(1+\|h_i-m_j\|^2)^{-1}}{\sum_{j=1}^{K}(1+\|h_i-m_j\|^2)^{-1}}, \quad j=1,2,\cdots,K \tag{12}$$

式中 p_{ik} 表示潜在空间样本 h_i 属于 μ_j 所在簇的概率。结合最小化相关熵 (KL 散度) 来降低软标签分布 P 与辅助目标分布 Q 之间的差异, 将聚类损失目标函数 J_c 定义为

$$J_c = \mathrm{KL}(\boldsymbol{Q}\,\|\,\boldsymbol{P}) = \frac{1}{N}\sum_i\sum_k q_{ik}\lg\frac{q_{ik}}{p_{ik}} \tag{13}$$

其中, 辅助目标分布分量 q_{ik} 是由 p_{ik} 得到的, 通过计算 $\frac{\partial J_c}{\partial p_{ik}}$ 使其为零, 求取相应封闭解获得 q_{ik}:

$$q_{ik} = \frac{p_{ik}^2/\sum_{i'}p_{i'k}}{\sum_{k'}\left(p_{ik'}^2/\sum_{i'}p_{i'k'}\right)} \tag{14}$$

最小化聚类损失目标函数 J_c 可视为自训练过程, 此外, 重构误差函数 J_r 采用均值误差测量 (mean squared error, MSE):

$$J_r = \sum_{i=1}^{n}\|x_i - g_{W'}(h_i)\|_2^2 \tag{15}$$

关于自编码器参数 $\{W,W'\}$ 及聚类中心 $\{\mu_j\}_{j=1}^K$ 的更新, 采用 mini-batch 随机梯度算法反向传播逐层训练网络。J_c 关于嵌入特征表示 h_i 和聚类中心 μ_j 的梯度计算为

$$\frac{\partial J_c}{\partial h_i} = 2\sum_{j=1}^{K}(1+\|h_i-\mu_j\|^2)^{-1}(q_{ij}-p_{ij})(h_i-\mu_j)$$

$$\frac{\partial J_c}{\partial \mu_i} = 2\sum_{j=1}^{K}(1+\|h_i-\mu_j\|^2)^{-1}(p_{ij}-q_{ij})(h_i-\mu_j)$$

将 $\frac{\partial J_c}{\partial h_i}$ 反向传播给栈式自编码器, 利用下层神经元梯度由上层神经元残差导出的规律, 自上而下反向逐层计算出 $\frac{\partial J_c}{\partial W}$ 及 $\frac{\partial J_c}{\partial W'}$。给定小批量样本数目 m 及学习率 λ, 编码器参数 W、解码器参数 W' 及聚类中心 μ_j 更新公式为

$$W = W - \frac{\lambda}{m}\sum_{i=1}^{m}\left(\frac{\partial J_r}{\partial W}+\gamma\frac{\partial J_c}{\partial W}\right) \tag{16}$$

$$W' = W' - \frac{\lambda}{m}\sum_{i=1}^{m}\frac{\partial J_r}{\partial W'} \tag{17}$$

$$\mu_j = \mu_j - \frac{\lambda}{m}\sum_{j=1}^{k}\frac{\partial J_r}{\partial \mu_j} \tag{18}$$

2.3　算法步骤

算法　结合度量融合和地标表示的自编码谱聚类算法

输入　输入数据 $\boldsymbol{X}$, 聚类数目 K, 地标点个数 p, 停止阈值 β, 近邻点数目 k, 学习率 λ, 迭代次数 h, 最大迭代次数 maxiter, 停止收敛阈值 Th。

输出　自编码器参数 $\{W,W'\}$ 及聚类中心 $\{\mu_j\}_{j=1}^K$ 及样本类标签 Label。

1) 根据式 (1)、(2) 计算节点 PR 值, 当所有节点满足式 (3) 时, 停止迭代, PR 值降序排列, 择取前 p 个高 PR 值节点作为地标点;

2) 依据式 (4) 成对相似度计算, 引入 k 近邻点, 以欧氏距离、Kendall Tau 距离 2 种度量方式, 通过式 (5) 分别计算相应的稀疏表示矩阵 $\boldsymbol{Z}^{(1)}$ 和 $\boldsymbol{Z}^{(2)}$。同时, 通过式 (7) 计算对应于 2 种度量方式的完全相似矩阵 $\boldsymbol{F}^{(1)}$、$\boldsymbol{F}^{(2)}$;

3) 将 $\boldsymbol{F}^{(1)}$、$\boldsymbol{F}^{(2)}$ 作为初始状态矩阵, $\boldsymbol{Z}^{(1)}$、$\boldsymbol{Z}^{(2)}$ 作为核矩阵, 根据式 (8) 进行矩阵融合, 迭代更新对应于每个度量的稀疏表示矩阵, 在 h 步后结合式 (9) 获得融合 2 种度量方式的稀疏表示矩阵 $\boldsymbol{Z}$;

4) 通过式 (10) 构造拉普拉斯矩阵, 将 $\boldsymbol{D}^{-\frac{1}{2}}\hat{\boldsymbol{Z}}^{\mathrm{T}}$ 作为栈式自编码器的输入, 前向训练网络获得初始化数据潜在特征表示 $\{h_i\}_{i=1}^n$;

5) 对嵌入特征表示 $\{h_i\}_{i=1}^n$ 进行预训练, 执行 k-means 算法, 获得初始自编码器参数 $\{W,W'\}$ 及聚类中心 $\{\mu_j\}_{j=1}^K$;

6) 若迭代次数大于 maxiter, 转至 10), 否则重复执行 7) ~ 9);

7) 根据式 (12) 和式 (14) 计算 p_{ik} 和 q_{ik};

8) 根据式 (11)、(13) 和 (15) 计算损失目标函数;

9) 采用 mini-batch 随机梯度算法反向传播逐

层训练网络，根据式 (16)~(18) 更新整个网络参数 $\{W, W'\}$ 及聚类中心 $\{\mu_j\}_{j=1}^{K}$；

10) 根据软标签分布 P 来分配样本类标签：$\text{Label}(i) = \arg\max_{k\in 1,2,\cdots,K} p_{ik}$。

2.4　算法复杂度分析

首先，从 n 个数据样本中选择 p 个地标点，该步骤时间复杂度为 $O(tpn)$，t 为迭代次数。相似度矩阵构造部分涉及了 2 种度量方案的融合以及构造稀疏表示矩阵[17]，时间复杂度为 $O(pn\log_2 n)$。本文算法用栈式自编码器代替拉普拉斯矩阵特征分解部分，时间复杂度为 $O(nD^2+nvK)$，K 为聚类数目，D 为设置的隐藏层最大单元数，v 为数据潜在特征表示维度，一般 $K \leqslant v \leqslant D$，所提算法整体时间复杂度为 $O(tpn+pn\log_2 n+nD^2)$。基于地标表示的快速谱聚类算法[9] 时间复杂度为 $O(tpn+pn+p^3+p^2n)$，由于 $p \ll n$，时间复杂度可写为 $O(tpn+pn+p^2n)$。p 和 D 取值相当且远小于 n，因而本文算法复杂度略高于基于地标表示的快速谱聚类算法时间复杂度，但远远低于基于自编码器的谱聚类算法 DEC[25] 的时间复杂度 $O(n^2+nD^2)$、GraEn [16] 的时间复杂度 $O(n^2+nKD)$。

3　实验及分析结果

3.1　实验环境

为证明本文算法适用于大规模数据集，选取如下 5 个数据集：MNIST、USPS、COIL100、Cover-Type 和 CIFAR10。这 5 个数据集包含不同类型的图像，如手写数字、英文字母表、物体图像和森林植被类等，样本数较大，可以有效验证本文算法在大规模数据集上的聚类性能。表 1 为相关数据集的具体信息。仿真实验硬件环境为 Intel(R) Core(TM) i7-6 700 CPU @3.40 GHz; 16 GB RAM; 操作系统：Windows7; 编程语言：Python 3.5。

表 1　实验数据集描述

Table 1　The description of experimental datasets

数据集	样本数	类	维数
MNIST	70 000	10	784
LetterRec	20 000	26	16
COIL100	7 200	100	1 024
CoverType	581 012	7	54
CIFAR10	60 000	10	1 024

本文算法与文献 [8-9] 基于地标表示的快速谱聚类算法 LSC-R 和 LSC-K，未考虑地标点的基于自编码的谱聚类算法 DEC[25] 及 GraEn[16]、文献 [17] 中自编码器与地标点结合的快速谱聚类算法 SCAL-R 及 SCAL-K 进行对比。LSC-R 及 SCAL-R 是通过随机采样获取 p 个地标点，LSC-K 及 SCAL-K 是采用 k-means 算法获得 p 个地标点的。

为保证算法之间的公平对比，具体参数如下设置：文献 [8-9]、[17] 及本文算法涉及的地标点个数 p 统一设置为 1 000，稀疏表示矩阵构造涉及的近邻点数目 k 设置为 5，所有算法涉及 k-means 部分迭代次数为 500。文献 [16]、[17]、[25] 及本文算法涉及的自编码器部分，编码器维度为 $p-500-500-2\,000-10$，解码器则为 $10-2\,000-500-500-p$，非线性激活函数统一采用线性校正单元 (rectified linear units, ReLU)。本文算法中涉及的停止阈值 $\beta = 10^{-3}$，距离度量融合步骤迭代次数设置为 $h = 10$，参数优化更新部分采用 mini-batch 随机梯度算法，小批量样本数目 m 为 256，学习率 $\lambda = 0.1$；在预训练阶段，执行 k-means 算法 20 次选取最佳结果，初始化自编码器参数和聚类中心；收敛阈值设置为 Th = 0.001，最大迭代次数 maxiter 为 20 000，调节潜在特征空间的失真程度 $\gamma = 0.1$。

3.2　实验分析

为了比较这些算法性能，通过聚类结果和样本真实标签进行对比。采用聚类准确率 (ACC) 和标准化互信息 (NMI) 来作为度量指标进行评估比较。ACC 和 NMI 这 2 种度量标准在 [0,1] 之间取值，值越大聚类性能越好。各种算法在 5 个大规模数据集的实验结果如表 2、3 所示。

表 2　不同数据集的准确率 (ACC) 的比较

Table 2　Comparison of clustering accuracy (ACC) for different data sets

算法	MNIST	LetterRec	COIL100	CoverType	CIFAR10
LSC-R[9]	0.589 0	0.292 2	0.489 6	0.247 5	0.471 6
LSC-K[9]	0.727 0	0.303 3	0.545 6	0.255 0	0.504 0
SCAL-R[17]	0.836 1	0.439 4	0.790 5	0.260 2	0.561 9
SCAL-K[17]	0.889 8	0.447 0	0.812 2	0.293 3	0.580 2
DEC[25]	0.865 1	0.297 5	0.721 9	0.249 7	0.482 1
GraEn[16]	0.818 2	0.287 2	0.735 0	0.221 8	0.513 4
本文算法	**0.901 2**	**0.469 0**	**0.890 3**	**0.332 5**	**0.612 9**

表 3　不同数据集的标准化互信息 (NMI) 的比较
Table 3　Comparison of normalized mutual information (NMI) for different data sets

算法	MNIST	LetterRec	COIL100	CoverType	CIFAR10
LSC-R[9]	0.591 1	0.373 4	0.731 5	0.083 1	0.034 8
LSC-K[9]	0.722 2	0.396 3	0.763 2	0.090 2	0.054 2
SCAL-R[17]	0.829 7	0.391 5	0.827 4	0.091 2	0.076 5
SCAL-K[17]	0.875 4	0.405 5	0.841 6	0.102 5	0.084 4
DEC[25]	0.836 9	0.385 1	0.804 5	0.085 6	0.065 1
GraEn[16]	0.747 3	0.371 2	0.791 1	0.071 9	0.057 9
本文算法	**0.882 1**	**0.443 5**	**0.870 2**	**0.134 2**	**0.152 1**

通过表 2、3 两种指标下的各种算法性能度量对比, 可以看出本文提出的结合度量融合和地标表示的自编码谱聚类算法在 MNIST、LetterRec、COIL100、CoverType 及 CIFAR10 这 5 个大规模数据集上, 相对于基于地标表示的快速谱聚类算法 LSC-R 和 LSC-K, 未考虑地标点的基于自编码的谱聚类算法 DEC 及 GraEn, 自编码器与地标点结合的快速谱聚类算法 SCAL-R 及 SCAL-K, 在 ACC 和 NMI 两种指标下均表现出了更为理想的聚类性能。同时, 还可以得出如下结论:

1) 本文算法在稀疏表示矩阵构造部分由于引入网页质量进行地标点择取, 并进行度量融合计算点对相似度, 在算法复杂度方面, 相对于旨在提升谱聚类算法效率的 LSC-R 和 LSC-K 处于劣势。从运行时间上看, 在 MNIST 数据集上, LSC-R 及 LSC-K 算法仅需 10 s 左右, 本文算法则需 200~300 s 的运行时间, 但相对于 DEC 及 GraEn, 本文算法运行时间降低了 1 000~2 000 s, 有了很大的提升。且从表 2 中可以看出, 就 ACC, 本文算法与 LSC-K 算法相比, 在 MNIST、LetterRec、COIL100、CoverType 及 CIFAR10F 分别提升了 17.42%、16.57%、34.47%、7.75%、10.89%, 与随机抽样择取地标点的 LSC-R 算法相比, 优势更大。同样地, 从表 3 看出, 本文算法 NMI 值也有了明显的提升。而与 SCAL-R 及 SCAL-K 算法相比, 本文融合两种距离度量多角度描述数据集的结构信息, 使其在 ACC、NMI 两种指标上均处于优势。

2) 相比未考虑地标点的基于自编码的谱聚类算法 DEC 及 GraEn, 本文算法 ACC 和 NMI 也有了一定的提升, 如与 DEC 算法对比, 本文算法在 LetterRec 数据集上 ACC、NMI 分别提升了 17.15%、5.84%。说明本文构造的结合度量融合和地标表示的相似度矩阵更全面地反映了数据之间的结构信息, 且运行效率大大得到提升。

3) 此外, 间接可以看出, 以 LSC-R、LSC-K 为代表的引入地标点算法与 DEC、GraEn 为代表的基于深度自编码算法相比, 在 MNIST、COIL100 及 CIFAR10 这些维度相对较高的大规模数据集中, 聚类性能处于劣势, 但在 LetterRec 及 CoverType 维度较低的数据集上则处于优势, 可见基于自编码器的算法对于高维数据集较为友好, 地标点的择取有利于大规模数据集的聚类性能。

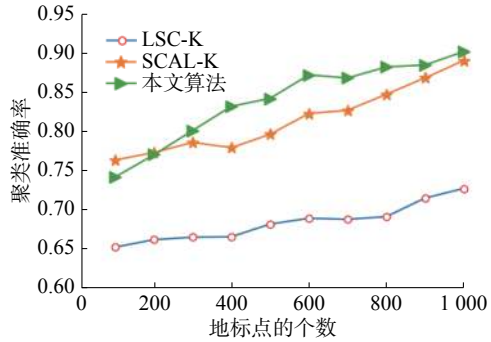
本文算法采用欧氏距离与 Kendall Tau 距离度量融合的方式, 挖掘数据的真实分布, 从上述结论 1) 可以看出算法性能会受到距离度量方式的影响。为深入分析, 将本文算法稀疏表示矩阵构造部分, 分别采用基于欧氏距离 (SCE) 与基于 Kendall Tau 距离 (SCK) 计算点对相似度, 采用 ACC 和 NMI 两个指标与本文算法融合度量方案进行性能对比。表 4 显示 3 种算法在 3 个数据集上的算法性能。从表 4 可以看出, 2 种距离度量方式的融合一定程度上能更全面地反映数据之间的结构信息, 从而获得更为理想的聚类性能。欧氏距离与 Kendall Tau 距离这两种度量方式的优劣, 并不能准确地从数据中评判出来, 但结合表 2、3 可以看出, 基于本文算法的 SCE 和 SCK 算法在 3 个数据集上与其他几种算法相比, 相对处于优势。特别是与 SCAL_R 及 SCAL_K 算法对应 ACC 和 NMI 对比可以得出, 本文通过基于网页质量的 PageRank 算法择取地标点总体上要优于随机选择与 k-means 算法的方法, 但也有低于这两种方法的情况出现, 为深入分析, 更改地标点个数 p 从 100~1 000, 在 3 个数据集上, 进行本文算法与 SCAL-R、SCAL-K 的 ACC 值的对比实验。图 1、2 分别显示了 SCAL-R、SCAL-K 及本文算法在选取的 3 个数据集上, 地标点个数从 100~1 000 之间变化, 所对应的 ACC 及 NMI 值。图中可直观看出, 本文算法在 3 个数据集上的 ACC、NMI 普遍比 LSC-K、SCAL-K 要高, 除去在 MNIST 数据集上, 地标点个数较少时, 本文算法与 SCAL-R 算法相比略处于劣势, 这是由于本文构造相似度矩阵采用的度量融合方案, 在地标点个数相对多时, 能有效地挖掘数据的结构信息, 地标点个数从 300 开始, 本文算法便展现出优势。同时从折线走势可以看出, 地标点个数的增加, 3 种算法的 ACC、NMI 值也随之提高, 且本文算法呈现稳步上升趋势, 这表明两个聚类性能指标受地标点个

数的选取影响很大。总体来说，在不同地标点的个数选取下，本文算法展现出优于其他算法的聚类性能。

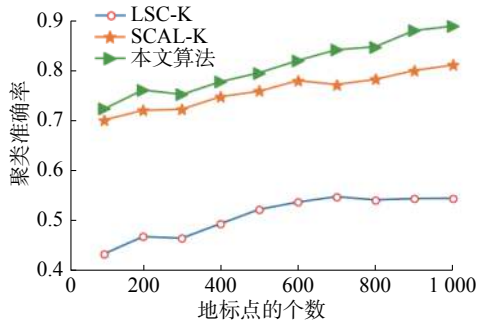
表 4　不同距离度量方案的算法性能对比

Table 4　Performance comparision of algorithms with different distance measurement schemes

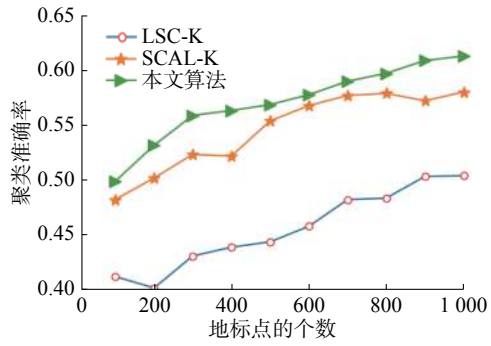
数据集	算法	SCE	SCK	本文算法
MNIST	ACC	0.873 3	0.862 4	0.901 2
	NMI	0.848 9	0.854 5	0.882 1
COIL100	ACC	0.839 1	0.821 4	0.890 3
	NMI	0.854 7	0.842 7	0.870 2
CIFAR10	ACC	0.597 5	0.589 9	0.612 9
	NMI	0.112 4	0.132 4	0.152 1



(a) MNIST 数据集



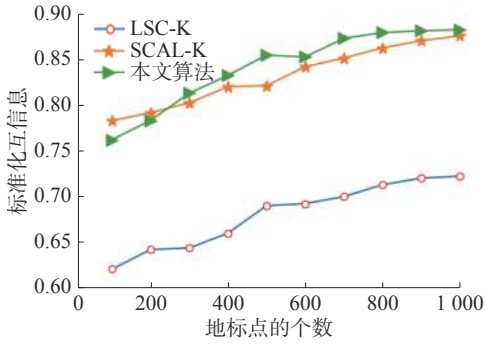
(b) COIL100 数据集



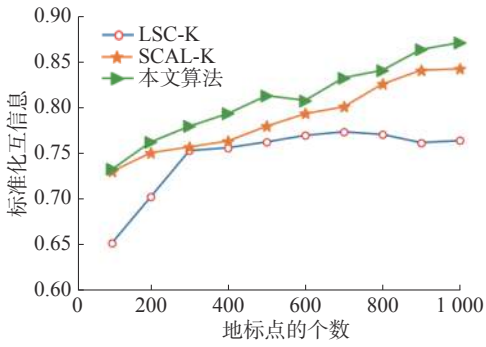
(c) CIFAR10 数据集

图 1　不同数据集上 3 种算法的 ACC 比较

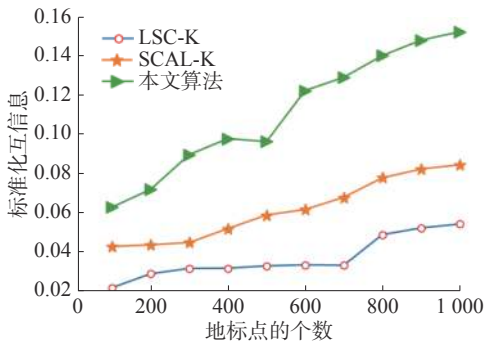
Fig. 1　Comparison of the ACCs of three algorithms on different datasets



(a) MNIST 数据集



(b) COIL100 数据集



(c) CIFAR10 数据集

图 2　不同数据集上 3 种算法的 NMI 比较

Fig. 2　Comparison of the NMIs of three algorithms on different dataset

4　结束语

随着数据规模的增大，结构信息的复杂度提高，在聚类过程中往往会耗费大量时间，存在相似度矩阵存储开销大及矩阵分解复杂度高的问题，且聚类精度也会受到影响。为此，本文提出一种结合度量融合和地标表示的自编码谱聚类算法，通过引入节点相对质量概念作为地标点选择的依据，以地标点与其他样本点之间相似度构造图相似度矩阵，以降低存储开销。同时，融合欧氏距离与 Kendall Tau 距离作为相似度度量方式，充分挖掘数据底层结构信息，且以栈式自编码器代替拉普拉斯矩阵分解步骤，通过联合学习框架进一步提高聚类精度。实验表明在几种大规模数据集上本文算法具有较好的聚类性能，但由于本

文算法聚类精度受地标点个数影响较大, 且选取方式略微增加了算法复杂度, 未来将致力于寻求更为有效的地标点选择方式, 在保证聚类精度的同时进一步降低算法复杂度。

参考文献:

[1] WANG Lijuan, DING Shifei, JIA Hongjie. An improvement of spectral clustering via message passing and density sensitive similarity[J]. IEEE access, 2019, 7: 101054–101062.

[2] LI Xinning, ZHAO Xiaoxiao, CHU Derun, et al. An autoencoder-based spectral clustering algorithm[J]. Soft computing, 2020, 24(3): 1661–1671.

[3] 王一宾, 李田力, 程玉胜. 结合谱聚类的标记分布学习 [J]. 智能系统学报, 2019, 14(5): 966–973.
WANG Yibin, LI Tianli, CHENG Yusheng. Label distribution learning based on spectral clustering[J]. CAAI transactions on intelligent systems, 2019, 14(5): 966–973.

[4] 赵晓晓, 周治平. 结合稀疏表示与约束传递的半监督谱聚类算法 [J]. 智能系统学报, 2018, 13(5): 855–863.
ZHAO Xiaoxiao, ZHOU Zhiping. A semi-supervised spectral clustering algorithm combined with sparse representation and constraint propagation[J]. CAAI transactions on intelligent systems, 2018, 13(5): 855–863.

[5] LANGONE R, SUYKENS J A K. Fast kernel spectral clustering[J]. Neurocomputing, 2017, 268: 27–33.

[6] ZHAN Qiang, MAO Yu. Improved spectral clustering based on Nyström method[J]. Multimedia tools and applications, 2017, 76(19): 20149–20165.

[7] YANG Xiaojun, YU Weizhong, WANG Rong, et al. Fast spectral clustering learning with hierarchical bipartite graph for large-scale data[J]. Pattern recognition letters, 2020, 130(2): 345–352.

[8] CHEN Xinlei, CAI Deng. Large scale spectral clustering with landmark-based representation[C]//Proceedings of the 24th AAAI Conference on Artificial Intelligence. San Francisco, USA, 2011: 313−318.

[9] CAI Deng, CHEN Xinlei. Large scale spectral clustering via landmark-based sparse representation[J]. IEEE trans cybern, 2015, 45(8): 1669–1680.

[10] 叶茂, 刘文芬. 基于快速地标采样的大规模谱聚类算法 [J]. 电子与信息学报, 2017, 39(2): 278–284.
YE Mao, LIU Wenfen. Large scale spectral clustering based on fast landmark sampling[J]. Journal of electronics and information technology, 2017, 39(2): 278–284.

[11] ZHANG Xianchao, ZONG Linlin, YOU Quanzeng, et al. Sampling for Nyström extension-based spectral clustering: incremental perspective and novel analysis[J]. ACM transactions on knowledge discovery from data, 2016, 11(1): 1–25.

[12] 邓思宇, 刘福伦, 黄雨婷, 等. 基于 PageRank 的主动学习算法 [J]. 智能系统学报, 2019, 14(3): 551–559.
DENG Siyu, LIU Fulun, HUANG Yuting, et al. Active learning through PageRank[J]. CAAI transactions on intelligent systems, 2019, 14(3): 551–559.

[13] RAFAILID D, CONSTANTINOU E, MANOLOPOULOS Y. Landmark selection for spectral clustering based on weighted PageRank[J]. Future generation computer systems, 2017, 68: 465–472.

[14] LIU Li, SUN Letian, CHEN Shiping, et al. K-PRSCAN: A clustering method based on PageRank[J]. Neurocomputing, 2016, 175: 65–80.

[15] JIA Hongjie, DING Shifei, DU Mingjing, et al. Approximate normalized cuts without eigen-decomposition[J]. Information sciences, 2016, 374: 135–150.

[16] TIAN Fei, GAO Bin, CUI Qing, et al. Learning deep representations for graph clustering[C]//Proceedings of the 28th AAAI Conference on Artificial Intelligence. Québec, Canada, 2014: 1293−1299.

[17] BANIJAMALI E, GHODSI A. Fast spectral clustering using autoencoders and landmarks[C]//Proceedings of International Conference Image Analysis and Recognition. Montreal, Canada, 2017: 380−388.

[18] 光俊叶, 邵伟, 孙亮, 等. 基于融合欧氏距离与 Kendall Tau 距离度量的谱聚类算法 [J]. 控制理论与应用, 2017, 34(6): 783–789.
GUANG Junye, SHAO Wei, SUN Liang, et al. Spectral clustering with mixed Euclidean and Kendall Tau metrics[J]. Control theory & applications, 2017, 34(6): 783–789.

[19] WEI Kai, TIAN Pingfang, GU Jingguang, et al. RDF data assessment based on metrics and improved PageRank algorithm[C]//Proceedings of International Conference on Database Systems for Advanced Applications. Suzhou, China, 2017: 204−212.

[20] 谢娟英, 丁丽娟. 完全自适应的谱聚类算法 [J]. 电子学报, 2019, 47(5): 1000–1008.
XIE Juanying, DING Lijuan. The true self-adaptive spectral clustering algorithms[J]. Acta electronica sinica, 2019, 47(5): 1000–1008.

[21] NG A Y, JORDAN M I, WEISS Y. On spectral clustering: analysis and an algorithm[C]//Proceedings of Neural Information Processing Systems 14, NIPS 2001. Vancouver, British Columbia, Canada, 2002: 849−856.

[22] XIE Juanying, ZHOU Ying, DING Lijuan. Local stand-

ard deviation spectral clustering[C]// Proceedings of the 2018 IEEE International Conference on Big Data and Smart Computing (BigComp). Shanghai, China, 2018: 242−250.

[23] WANG Bo, JIANG Jiayan, WANG Wei, et al. Unsupervised metric fusion by cross diffusion[C]//Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition. Providence, Rhode Island, 2012: 2997−3004.

[24] COIFMAN R R, LAFON S. Diffusion maps[J]. Applied and computational harmonic analysis, 2006, 21(1): 5–30.

[25] XIE Junyuan, GIRSHICK R B, FARHADI A. Unsupervised deep embedding for clustering analysis[C]//Proceedings of the 33nd International Conference on Machine Learning. New York, USA, 2016: 478−487.

作者简介：

张敏，硕士研究生，主要研究方向为数据挖掘。

周治平，教授，博士，主要研究方向为智能检测、网络安全。发表学术论文 80 余篇。

第五届认知系统和信息处理国际会议

认知系统和信息处理国际会议（ICCSIP）每两年举办一次，已成为认知科学、智能系统、机器人等领域学者与企业的交流桥梁，为促进海内外学者的交流提供了全球化的平台，目前已举办 4 届，录用的论文在 Springer 出版。当前正是认知科学与人工智能的飞速发展期，二者的结合与交融有利于触发瞬间灵感，推动创新步伐。因此，第五届认知系统和信息处理国际会议 (ICCSIP2020) 主题为”面向人工智能的认知计算”，并于 2020 年 12 月 18-20 日在中国珠海横琴岛召开，希望推动认知、心理、智能、机器人等领域的融通交汇。此外，还将特别设立科技抗疫专题，欢迎各界人士依托此平台为全球科技抗疫贡献力量。同时国际会议现场还举办中国人工智能学会认知系统与信息处理专委会的年会。

1.专辑录用的优秀论文将会推荐至下列期刊专辑发表：

1）Cognitive Computation and Systems；2）InternationalJournal of Control, Automation, and Systems；3）Robotics and Autonomous Systems；4）Science China: Information Sciences；5）Tsinghua Science and Technology (English Version)；6）IEEE Transactions on Cognitive and Development Systems；7）IEEE Transactions on Fuzzy Systems。

2.主题（包含但不限于）：

认知系统：认知科学与技术,视觉认知处理,听觉认知处理,新颖认知计算模型,认知指标,认知心理学,认知机器人学,认知无线电,认知雷达。

信息处理：信息表示与度量,多模态信息交互与融合,大数据和智能信息处理,神经认知计算与学习,视觉信息处理,脑机接口,生物信息学及应用,灵巧操作的多模态认知机制,极限学习及应用。

3.重要日期：

投稿截止：2020.10.10

录用通知：2020.10.30

会议注册：2020.11.20

4.联系我们：

E-mail:csip2020-2020@163.com, 手机号：15952525480

更多信息请详见：http://iccsip2020.caai.cn/