GoT: a Growing Tree Model for Clustering Ensemble

Feijiang Li, Yuhua Qian, Jieting Wang

摘要

聚类集成技术将多个聚类结果集成在一起，可以提高最终聚类的准确性和健壮性。在许多聚类集成算法中，联合矩阵 (CA 矩阵) 起着重要的作用，它反映了任意两个样本被划分到同一聚类的频率。然而，通常情况下，CA 矩阵是高度稀疏的，值密度低，这可能会限制基于它的算法的性能。为了解决这些问题，本文提出了一个生长树模型 (GoT)。该模型首先利用最短路径技术对 CA 矩阵进行优化，以降低其稀疏性。然后，发现一组具有代表性的原型实例。最后，为了处理 CA 矩阵的值密度的问题，原型逐渐连接到它们的邻域，就像一组树木生长一样。通过理论分析和实验分析，说明了所发现的原型算例的合理性。综合数据集直观地展示了 GoT 的工作机理。对 8 个 UCI 数据集和 8 个图像数据集的实验分析表明，该算法优于 9 种典型的聚类集成算法。

关键词：聚类集成；CA 矩阵；RM 矩阵；生长树模型

1 引言

数据聚类是机器学习中一种有趣的无监督技术，其目的是根据样本之间的相似性将数据集划分为同质组或聚类。聚类集成技术因其能够提高聚类的有效性和健壮性而备受关注。聚类集成技术通过组合多个不同的聚类结果来发现数据的组结构，而不涉及原始数据集。由于处理过程灵活，聚类集成技术被广泛应用于许多具有挑战性的任务中，如高维数据聚类、大规模数据聚类、时态数据聚类、图像分割等。给定一组聚类结果，用两个样本出现在同一个聚类中的频率来衡量两个样本之间的关系。所有成对频率形成协关联矩阵（CA 矩阵）。CA 矩阵被大量的聚类集成技术所利用，然而，CA 矩阵的一些局限性可能会影响基于该矩阵的聚类集成方法的性能。限制主要来自两个方面: 高稀疏：CA 矩阵是高度稀疏的，这使得大部分样本之间的关系不能被反映出来。低值密度：协关联值越高，信息越可靠，而 CA 矩阵的大部分元素都是小值。在这篇论文中，我们专注于解决 CA 矩阵的局限性。为了处理 CA 矩阵的稀疏性，我们引入了最短路径技术来优化 CA 矩阵。针对聚类集成的低密度限制，提出了一种生长树模型。在生长树模型中，首先基于改进的 CA 矩阵发现一组原型样本，并将其视为树根。然后，这些树的根通过与数据集中的样本相连接而生长。不同于传统的将每个样本与其最近的原型连接起来的连接过程，我们逐渐将原型周围的样本连接起来，扩展原型集。首先，树根连接到与树根有较强关系的样本; 然后，指定的样本形成新的叶节点，并与与它们有较强关系的样本连接。重复这个过程，直到数据集中的所有样本都连接到树。

2 相关工作

协关联矩阵（CA 矩阵）的构建：通过多次聚类，根据每两个点之间处于同一个簇的概率，构建协关联矩阵，该矩阵应该是对称的。

3 本文方法

3.1 利用 CA 矩阵构造 DM 矩阵和 RM 矩阵

CA 矩阵的每一个元素值表示两个点处于同一个簇的频率，可以看作是两个点之间的“相似度”。而 CA 矩阵是高度稀疏矩阵，有非常多的零值。对于 CA 矩阵中的每一个元素，执行 1-CA[i][j]，可以得到另一个矩阵，称为 DM 矩阵。易得，CA 矩阵中元素值越小，即每两个点之间的“相似性”越小，对应的 DM 矩阵元素值越大，因此，可以把 DM 矩阵中的元素值看作是两个点之间的“距离”。再对 DM 矩阵利用 Dijkstra 算法得到每两个点之间的“最小距离”，得到 RM 矩阵。

3.2 发现原型样本（即每个簇的中心点

样本成为原型样本的趋势包含两个因素，即局部密度和代表性容量。在本文中，我们介绍了一种基于一组聚类结果来度量样本成为原型样本的趋势的方法。基于一组基本聚类结果，样本 xi 的密度为: density（i）= Σ|cb(xi)|/(l * n)

其中 |cb(xi)| 为聚类结果中包含 xi 的聚类样本个数，l 为聚类的次数，n 为样本点个数。xi 的代表性容量为：capacity (i) = min pij 其中，j：pj > pi

xi 成为原型样本的可能性为: tendency(i) = density（i）* capacity (i)

将有样本的 ri 计算出来之后，选取 ri 最大的 K 个样本作为原型样本（K 为簇的个数）

3.3 利用 GoT 模型将其他点加入到对应的簇中

大多数基于原型的算法在发现一组原型样本后，根据距离或相似度度量将其他样本分配给离它最近的原型。CA 矩阵的低值密度可能会影响分配过程的有效性。为了应对这一挑战，本文提出了一种利用可靠信息优势的生长树模型。在生长树模型中，每个原型样本都被视为根。然后，根逐渐生长到树干和叶子，这是数据样本。假设得到的原型样例集为 Z = z1, z2，…zk。我们首先建立了一个有 k 棵树的森林 F = t1, t2，…，tk，其中每棵树都有一个原型样本作为根。一开始，每棵树只有一个节点，即原型样本 ti = zi。然后，一棵树就会通过逐渐接近它附近的样本而生长。得到了一组树 F = t1, t2，…，tk，F 到达的样本应该具有较高的置信度才能被正确分配。为了量化该置信水平，引入样本 xi 的裕度，即其最亲密树与第二亲密树的差值: m(i) = ot(xi,tp) −max(xi,tp’)

该公式表示某个点离最近原型样本的距离-第二近原型样本的距离，该数值越大，证明这个点加入最近原型样本簇的置信度越高。通过设置边界，我们可以选择分配到树形结构 F 的样本，阈值为 th（th 值的设置在另一篇论文中有提及，在此次实验中，我们可以自行设置 th 值）。对于 m（xi）>th 的样本，将其加入到对应的簇中，其他的样本点则直接舍弃，直到所有的点都处理完成，聚类集成的结果也就完成了。

4 复现细节

4.1 与已有开源代码对比

本论文没有开源代码

4.2 实验环境搭建

直接利用 pycharm 集成开发环境即可, 利用 numpy，sklearn 等进行数据处理，根据论文的算法描述来编写函数，用与论文一样的数据集来进行实验，将结果与论文比较。

5 实验结果分析

选取四个数据集进行实验，与论文的实验结果进行比较（评价指标为 ARI 和 NMI，ARI 和 NMI 是聚类集成常用的算法评价指标）。实验结果见 Figure 文件夹中的 result.png。总体来看，实验结果与论文结果的 ARI 值和 NMI 值都非常接近，平均误差大概在 3

6 总结与展望

许多聚类集成算法都是基于 CA 矩阵的。然而，CA 矩阵是高度稀疏和低值密度的，这可能会影响这些算法的性能。在本文中，我们引入了最短路径技术来降低协关联矩阵的稀疏性。此外，我们还提出了一个生长树模型来整合多个聚类结果。从理论上和实验上说明了原型算例的合理性。通过多个数据集，直观地展示了生长树模型的工作机理。实验结果表明，在四个 UCI 数据集上，该模型有效地整合了多个聚类结果。然而，当数据集样本量较大时，程序的运行时间会呈指数增长，如何实现大规模数据集的聚类集成，一个成为今后的重点研究方向。